GOC User Manual

Document No.: ED-2002-114 Version: 1.10

Revision:

Version	Date	Description
1.0	January 2015	First Release
1.1	April 2015	Added details of GOC variants GC35MH-32MT-DSS, GC35MH-16MT-DSS and GC-8ET-ESS
1.2	June 2015	Updated specifications of GC-8ET-ESS, Precaution to be taken in Installation and Wiring and COM Extension Unit
1.3	Oct 2015	Added details of GOC I/O Extension GC-4HSXPTY
1.4	Dec 2015	Added details of GOC I/O Extension GC-4UAD-10E
1.5	Feb 2016	Updated with microSD card support and Data Logging support
1.6	April 2016	Added details of frequency and pulse on time measurement support in GOC
1.7	August 2016	Added details of GOC I/O Extension GC-4UAD-16
1.8	March 2017	Added note related to maximum no. of user defined screens (up to 64) for HMI
1.9	April 2018	Added specifications for CC-Link IEF Basic Master & Slave
1.10	May 2018	General specifications are updated. Added parameter “Approvals” of CE certification.

Thanks for choosing Graphical Operation Controller (GOC), the Micro Programmable Logic Controller with built in display keyboard and illuminated keys offering high flexibility in I/Os, features and functionality. Before using this controller, please read this manual carefully for safety precautions, specifications, dimensional details, installation guidelines and procedures, wiring guidelines and procedures.

Safety Recommendations

- Read and understand the manual carefully before product use, to avoid damages to persons, property & environments. Ensure safe & proper usage of this product.
- The qualified personnel should only install & operate the product. The personnel should be aware of all safety of automated products & completely familiar with all associated documentation for the said product.
- Manual should be located at the easily retrievable location for reference. Also, share this manual with the end user of this product.
- This controller to be considered as Industrial waste.

Protect the controller from conductive dust, corrosive gases, wire debris, flammable gases, rain and fluid from entering into the controller through ventilation slits; this may cause malfunction, damage, fire, electrical shock and deterioration to the controller.

The controller should not be exposed to direct sunlight, high explosive risk, excessive magnetic interference & inflammable substances.

Do not modify, dismantle, re construct and repair the controller. For repair, contact the nearest authorized Sales / Repair agency.

Do not paint the controller.

Provide external interlock circuit like emergency stop or protective circuit to keep the systems safe, in case, there is problem in the controller.

For output, place fail safe protective circuit external to the controller to ensure machine operates safely, in case of, the controller fault condition.

Also, it is recommended to build external output monitoring circuit to identify the output to which the problem has occurred.

During installation and wiring or maintenance activities of the controller and extensions, switch OFF the controller supply as well as all the phases of power supply for control panel. Failure to do so may cause personnel injury or controller malfunction.

Do not touch any terminal while the controller power is ON. It may cause electric shocks and malfunction due to static electricity.

Refer this manual before and during usage of either hardware or software key feature/s of controller.

Use controller within the range of general and technical specifications.

This page is kept blank intentionally

Content

1 Overview

Graphic Operation Controller (GOC) is a micro range of controller which consists of PLC function, HMI function and embedded illuminated keys. It is designed to cater the automation requirements of any small size, standalone machine.

1.1 Highlighting Features

PLC Function	
Flexible Hardware Configuration	▪ Various options of main units and extension units depending upon number and type of I/Os. ▪ Functionality can be enhanced with addition of up to 2 I/O extension units and up to 2 COM extension units ▪ Offers 16 digital I/Os minimum to 48 digital I/Os maximum. Can provide up to 8 analog I/Os. ▪ Up to 2 serial ports by adding COM extension units ▪ Programming via Ethernet and RS232 interface possible
Configurable Special Functionality to Digital Inputs in Main Unit	▪ Up to 2 single phase counters with software direction and start/stop control ▪ 2 counters (inputs I00 and I03) up to 20 KHz input frequency ▪ Up to 2 Quadrature ABZ encoder interfaces (inputs I00, I01, I02 and I03, I04, I05) with 10 KHz input frequency ▪ 2 inputs (inputs I02 and I05) can be individually configured for hardware input interrupt (rising/falling edge) Above functionality with enhancement is supported for extension unit GC-4HSXPTY when fixed in IO1 slot on the back side of Main unit ▪ Additionally, 2 Inputs (I02 and I05) can be individually configured for pulse catch or frequency measurement or pulse ON time measurement function. This functionality is supported for inputs on Main unit only.
Supports Interrupt Events	▪ Provides fast and predictable response to external events from the field such as. Digital input interrupt, counter count match, etc ▪ One periodic interrupt event with 0.5/ 1 ms resolution. Periodic interrupt can be set from 0.5/ 1ms to 1250/ 2500 ms. ▪ Up to 2 high speed counter compare interrupts ▪ Up to 2 hardware input rising edge and falling edge interrupts
General Features	▪ Built-in Real Time Clock ▪ Controller input power is 24 VDC ▪ Front panel mount; IP65 protection from front, IP20 protection from rare
Common Programming Platform	▪ Windows based IEC 61131-3 compatible programming software CoDeSys. ▪ Single software for programming PLC and HMI functionality.
Integrated Tool with User Friendly GUI *Hardware Configuration Tool*	▪ Configuration and Parameterization of Main unit and Extension units ▪ Special functionality configuration: - Counter, Encoder, Counter compare interrupt, Hardware input interrupt, Pulse catch, Frequency measurement and Pulse On time measurement. ▪ Serial port parameterization and protocol configuration ▪ Analog I/O Configuration and Handling

Highlighting Features…

HMI Function	
Built-in LCD Display	▪ 128 x 64 pixel graphics LCD with backlit, View size: 70.7 x 38.8 mm ▪ White characters with blue backlit ▪ 10 keys for display navigation, data entry and user settings - 5 user configurable function keys with local/ global operation ▪ 8 illuminated keys; keys with dual colour (Green, Red) LEDs
User Defined LCD Screens	▪ Up to 64 user defined screens ▪ Programmable screen actions (Entry, Display, Exit) and screen chaining ▪ 5 fonts, single size, double size, normal/bold, normal/ Inverse selection, blinking option ▪ Support of Static Text and Dynamic (ASCII) Text. Text with inclusion of special characters ▪ Screen objects like Static Text, ASCII, Numeric, Bit, Time, Date, Time of Day, RTC ▪ Monitor/ modify PLC data with different data types and formats ▪ Alphanumeric data entry ▪ Direct access of PLC variables with symbolic name
Advanced HMI Features	▪ Graphical symbols (32 x 32 pixel size and 48 x 48 pixel size) ▪ Multi-language support ▪ Custom image import to show logo, special symbol on LCD ▪ Screen level password ▪ Data logging in SD memory card
Function Keys	▪ 5 HMI keys F1 to F5 ▪ User configurable operation Inch/ Toggle/ GoTo Screen ▪ Local/ Global operation ▪ Key legend on LCD for F2, F3, F4
Built-in Status and Diagnostics	▪ One Key I/O Monitor ▪ System Menu for - Monitoring system status - Monitoring input and output status - System diagnostics - System settings - Log in, log out and modify password
Integrated Tool with User Friendly GUI *HMI Configuration Tool*	▪ Fast screen definition with drag n drop objects and properties ▪ Assign screen actions (Entry, Display, Exit) ▪ Define screen chaining ▪ Configurable operation for function key configuration with local/global scope ▪ Configurable operation for illuminated keys and control of dual-colour LEDs ▪ Multi-language configuration ▪ Custom image import ▪ Configuration of data logging using SD memory card

Highlighting Features…

Illuminated Keys	
Illuminated Keys	▪ 8 Illuminated keys i.e. keys with Dual colour bright LEDs ▪ User configurable operation Inch/ Toggle/ GoTo Screen ▪ LED control Red/Green/Yellow ON/ OFF/ Blink Slow/ Blink Fast ▪ Multi-fold usage combined with slide-in label - PB Station, Alarm Annunciation. Process/ Machine Mimic
Easy Customization of Front Looks	▪ Insertable slide-in label over illuminated keys ▪ Customizable for OEM branding.

MicroSD card support	
Data Logging	▪ Up to 32 log variables with 1 group ▪ Logging interval 500 ms/ 1 Sec minimum depending on record size ▪ Computer friendly .CSV file format of log file ▪ User controlled log trigger and file switching ▪ PC based SD card explorer tool to retrieve files

Ethernet protocol support	
Ethernet protocols	▪ Configuration via function blocks ▪ Modbus TCP server ▪ CC-Link IEF Basic master or slave

1.2 Nomenclature

GOC consists of Main unit with built-in I/Os, display, keypad and illuminated keys. User can attach I/O extension units (up to 2) and COM extension units (up to 2) to add I/Os and enhance functionality.

This section provides nomenclature details as below.

- Main unit
- I/O extension unit
- COM extension unit
- Main unit with extension units

1.2.1 Main Unit

The figure below shows all the views of bare Main unit with part names. All dimensions are in mm.

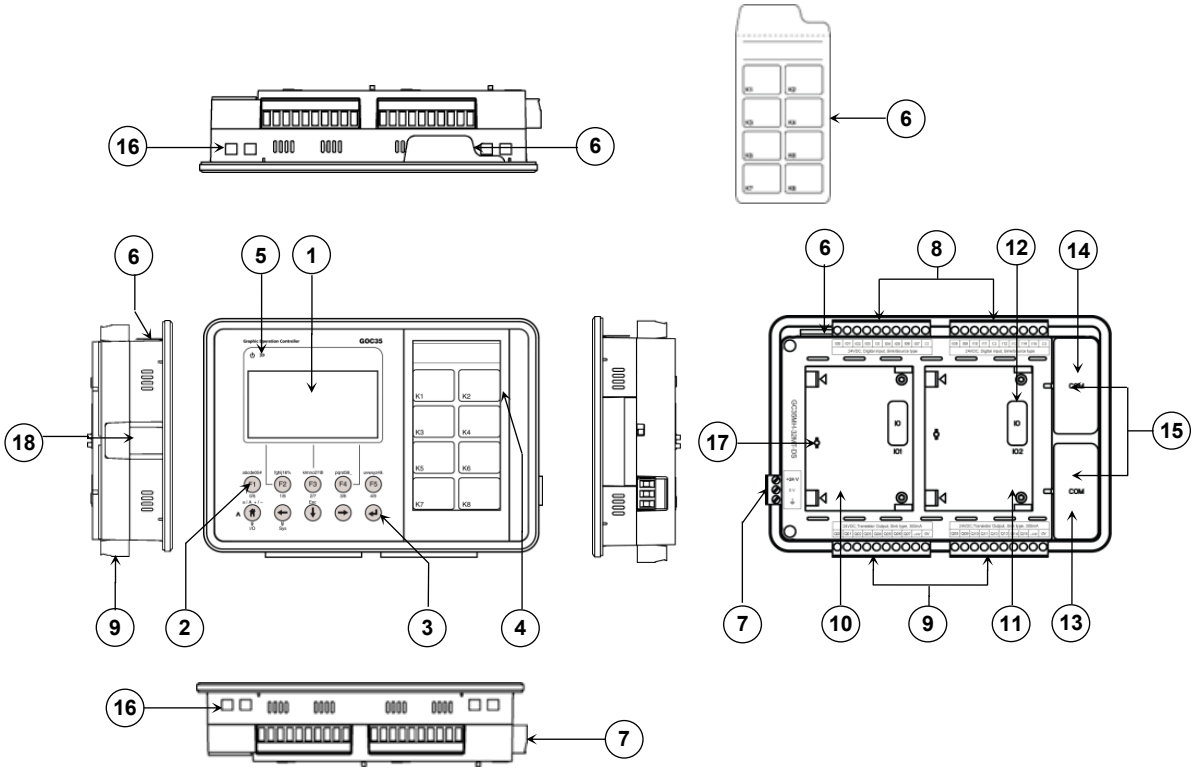


Figure 1: Main unit nomenclature

1. 128 x 64 pixels LCD with backlit
2. Function keys [F1 to F5]
3. HMI keys for display navigation and settings
4. 8 illuminated keys
5. LED indications [POWER, RUN]
6. Slide-in label
7. 24 VDC supply terminal block [+24VDC, 0V, Earth]
8. 2 nos., 10-pins input terminal block
9. 2 nos., 10-pins output terminal block
10. IO1 slot
11. IO2 slot
12. IO slot cover
13. COM1 slot
14. COM2 slot
15. COM slot cover
16. Slots for mounting clamp
17. Slot projection
18. microSD Card slot

MicroSD card slot is available only on GOC Main units dispatched from June 2016.

1.2.2 I/O Extension Unit

User can attach up to 2 I/O extension units on the back side of Main unit. The figure below shows all the views of I/O extension unit with part names.

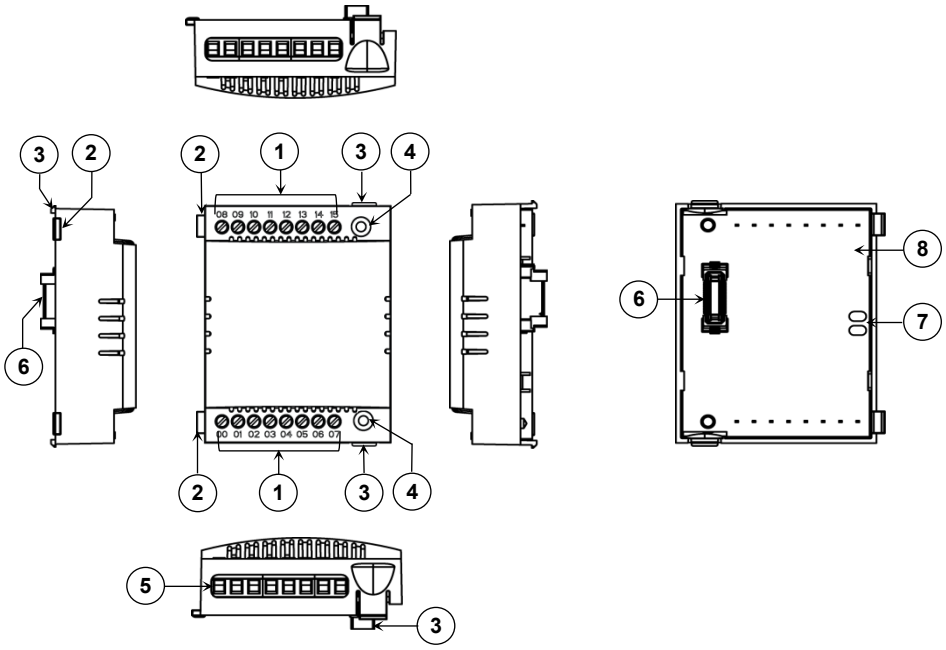


Figure 2: I/O extension unit nomenclature

Parts Description
1. I/O terminals
2. Latch
3. Clip
4. Unit fixing screw hole
5. 8-pin fixed I/O terminal block
6. Interface connector
7. Slot position holes
8. Printed circuit board (PCB)

Back side of I/O extension unit is open. Do not touch PCB and interface connector. It may cause damage to electronic hardware due to electrostatic discharge.

1.2.3 COM Extension Unit

User can attach up to 2 COM extension units on the back side of Main unit. The figure below shows all the views of COM extension unit with part names.

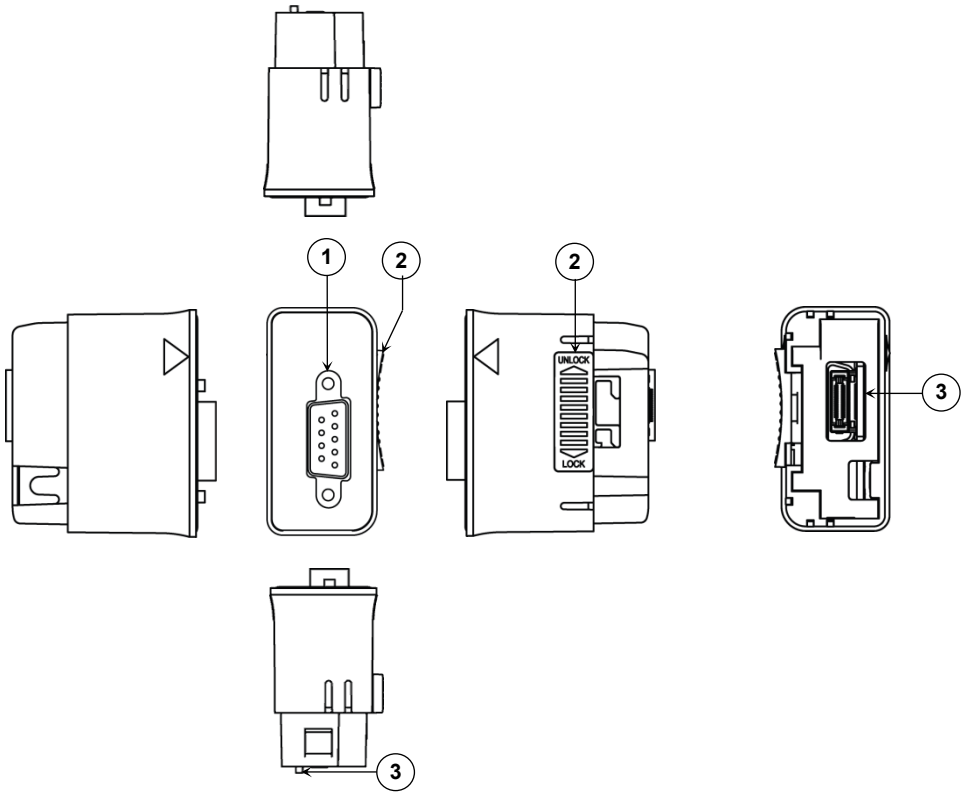


Figure 3: COM extension unit nomenclature

Parts Description

1. Connector for external communication interface
   - 9-pin D male connector for GC-232-COM
   - 5-pin removable terminal block for GC-422-COM
   - RJ45 connector for GC-ENET-COM
2. Locking clip
3. Interface connector

Nomenclature details in Figure 3 shows GC-RS232-COM extension unit. Similar plastic enclosure is used for other COM extension units like GC-RS422-COM and GC-ENET-COM but with different interface connectors for external communication interface.

1.2.4 Main Unit with Extension Units

User can attach up to 2 I/O extension units and up to 2 COM extension units on the back side of Main unit. The figure below shows all the views of Main unit attached with 2 I/O extension units and 2 COM extension units with part names.

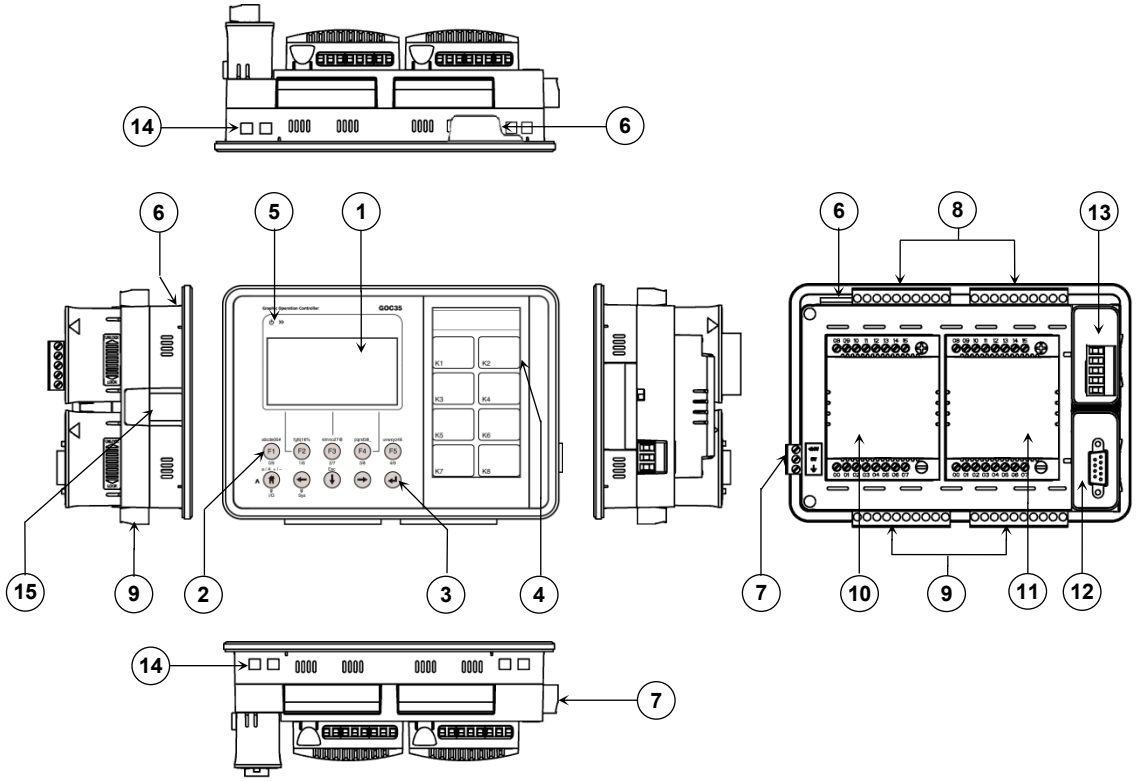


Figure 4: Main unit nomenclature with extension units

1. 128 x 64 pixels LCD with backlit
2. Function keys [F1 to F5]
3. HMI keys for display navigation and data entry
4. 8 illuminated keys
5. LED indications [POWER, RUN]
6. Slide-in label
7. 24 VDC supply terminal block [+, -, Earth]

8. 2 nos., 10-pins input terminal block
9. 2 nos., 10-pins output terminal block
10. IO1 extension unit
11. IO2 extension unit
12. COM1 extension unit
13. COM2 extension unit
14. Slot for mounting clamp
15. microSD card slot

Ensure that slot covers are fixed for unused slots of I/O extension/s as well as COM extension/s.

microSD card slot is available only on GOC main units manufactured from June 2016.

Month and year of manufacturing can be decoded from 11 digit product serial number.

In serial number of GOC, third and fourth digit designate year of manufacturing (16 for year 2016) and fifth digit is month of manufacturing as 1 for Jan, 2 for Feb….9 for September and ‘A’ as October, B as November and C as December.

1.3 Ordering Information

Ordering Code	Ordering Description	Details
Main Units		
GC35MH-32MT-DS	GOC-H MAIN, MIX 16DCIP+16DCOP, SINK, 300mA	16 Pt. 24 VDC Digital Input, sink/source + 16 Pt. 24 VDC Transistor Output, Sink type, 300mA per output Horizontal model
GC35MH-32MR-D	GOC-H MAIN, MIX 16DCIP+16RL OP, 500mA	16 Pt. 24 VDC Digital Input, sink/source + 16 Pt. Relay Output, 500mA per output, 220 VAC/30 VDC Horizontal model
GC35MH-32MT-DSS	GOC-H MAIN, MIX 16DCIP+16DCOP, SOURCE, 300mA	16 Pt. 24 VDC Digital Input, sink/source + 16 Pt. 24 VDC Transistor Output, Source type, 300mA per output Horizontal model
GC35MH-16MT-DS	GOC-H MAIN, MIX 8DCIP+8DCOP, SINK, 300mA	8 Pt. 24 VDC Digital Input, sink/source + 8 Pt. 24 VDC Transistor Output, Sink type, 300mA per output Horizontal model
GC35MH-16MR-D	GOC-H MAIN, MIX 8DCIP+8RLOP, 500mA	8 Pt. 24 VDC Digital Input, sink/source + 8 Pt. Relay Output, 500mA per output, 220 VAC/30 VDC Horizontal model
GC35MH-16MT-DSS	GOC-H MAIN, MIX 8DCIP+8DCOP, SOURCE, 300mA	8 Pt. 24 VDC Digital Input, sink/source + 8 Pt. 24 VDC Transistor Output, Source type, 300mA per output Horizontal model
COM Extension Units		
GC-RS232-COM	GOC COM EXT PORT RS232 SERIAL	1 Port RS232 Serial
GC-RS422-COM	GOC COM EXT PORT RS422/485 SERIAL	1 Port RS422/485 Serial
GC-ENET-COM	GOC COM EXT TCP/IP, 10 BASE T/ 100BASE-TX	10 Base T/100 Base –TX Ethernet TCP/IP for Programming

Ordering Information…

Ordering Code	Ordering Description	Details
IO Extension Units		
GC-8EX-ES	GOC EXT DI 8DC IP, 24VDC	8 Pt. 24 VDC Digital Input, sink/source
GC-6EYR-ES	GOC EXT DO 6RL OP, 500mA , 220VAC/ 30VDC	6 Pt. Relay Output, 500mA per output, 220 VAC/30 VDC
GC-4AD-12	GOC EXT AI 4CH AIP, V/I ,12BITS	4 Ch. Analog Voltage/Current Input, 12-bit
GC-4DA-12	GOC EXT AO 4CH AOP, V/I ,12BITS	4 Ch. Analog Voltage/Current Output, 12-bit
GC-4UAD-10	GOC EXT AI 4CH AIP, V/I,10BITS, PT100	4 Ch. Universal Analog Voltage/Current/ PT100 (-50 to 150^{0}C) Input, 10-bit
GC-4UAD-10E	GOC EXT AI 4CH AIP, V/ I/ PT, 10BITS	4 Ch. Universal Analog Voltage/Current/ PT100 (-50 to 450^{0}C) Input, 10-bit
GC-4UAD-16	GOC EXT AI 4CH AIP, V/ I/ Tc/ PT, 16BITS	4 Ch. Universal Analog Voltage/Current/ Thermocouple/ PT100/ PT1000 Input, 16-bit
GC-8ET-ESS	GOC EXT 4DC IP, 4DC OP SOURCE, 1.5A	4 Pt. 24 VDC Digital Input, sink/source + 4 Pt. 24 VDC Transistor Output, Source type, 1.5A per output
GC-4HSXPTY	GOC 2CH, HSC/ENCODER, 100/50KHZ, 2CH, PTO 100	2 Ch. HSC/Encoder Input, 24VDC Sink/Source+ 2 Ch. PTO, 5VDC Differential
Miscellaneous Items		
GC-10TB	TERMINAL BLOCK 10PIN I/O, FEMALE	10-pin female I/O Terminal Block
GC-3TB	TERMINAL BLOCK 3 PIN, PSU, FEMALE	3-pin female PSU Connector
GC-5TB	TERMINAL BLOCK 5 PIN, RS422/ RS485, FEMALE	5-pin female RS422/485 Connector
GC-ENET-03-CAB	CABLE PROGRAMMING ETHERNET, L3M	Programming cable Ethernet, 3 meter
GC -TOOLKIT	CD FOR GOC PROGRAMMING SOFTWARE	Programming Software CD

1.4 General Specifications

Item		Description	
Power supply	Input voltage	24 VDC (18 to 30 VDC ripple included), 400 mA, 9.6 Watt.	
	Inrush current	15 Amps maximu**m** for 6 ms duration	
	Fuse protection	Subminiature fuse T630 mA, 250V, Type 372, Littelfuse make Fixed internally.	
	Reverse polarity	Protected by series diode up to 40 V	
Operating temperature		**Operating:** 0 to 55 °C	**Storage:** -20 to 75 °C
Humidity		**Operating:** 10 to 90 % RH, No condensation	**Storage:** 10 to 90 % RH, No condensation
Altitude		2000 m or less	
Pollution level		2 maximum. (only non-conductive pollution)	
Operating atmosphere		Corrosive gas must not be present	
EMC – Immunity		as required by IEC 61131-2, IEC 61000-6-2	
Electro Static Discharge (ESD) (IEC 61000-4-2)		±8 KV Air discharge, ±4KV contact discharge	
Electrical Fast Transient (EFT) (IEC 61000-4-4)		**Power line:** ±2 KV, **Digital I/O:** ±1 KV, **Analog and communication I/O:** ±1 KV	
Radiation Susceptibility (RS) (IEC 61000-4-3)		80 MHz ~ 2.7 GHz, 10 V/m to 1 V/m, 80% AM at 1 KHz	
Conducted by Radio Frequency (CRF) (IEC 61000-4-6)		0.15 MHz~80 MHz,10V/m, 80% AM at 1 KHz	
Surge (IEC 61000-4-5)		**Power line:** ±0.5 KV, Digital I/O : ±1 KV, **Analog and communication I/O:** ±1 KV	
Power Frequency Magnetic Field (IEC 61000-4-8)		30 A/m, 50 /60 Hz	

General specifications…

Item		Description
EMC – Emission		as required by IEC 61131-2, IEC 61000-6-4
Radiated Emission (CISPR 16-2-3):		30 MHz ~ 1000 MHz
Over voltage category		II (IEC 60664-1) The surge voltage withstand level for up to the rated voltage of 30V is ±500 V
Vibration, Shock as required by EN- 61131-2		IEC 60068 -2-6 (test Fc), IEC 60068-2-27 test Ea
Class of equipment		Front panel mount
IP protection		IP65 from front. IP20 from rare
Dimensions (W x H x D) in mm		**Main unit: Front panel:** 177.0 (W) x 123.0 (H) x 4 (D) **Rear side:** 164.5 (W) x 106.5 (H) x 38.3 (D)
		I/O extension unit : 61.5 (W) x 75 (H) x 24.5 (D)
		COM extension unit : 26.0 (W) x 51.0 (H) x 42.2 (D)
Recommended I/O wires		0.5 to 1 mm² copper, stranded (flexible) or solid wire
I/O wiring	Termination lugs	For 0.5 to 1 mm² wire, insertion length 6 mm
	Suggested tool	Flat blade screwdriver 3 mm wide, 0.4 mm thick
Approvals		CE

1.5 Technical Specifications

This section provides CPU specifications covering system specifications.

Item	Description
Execution time	**BOOL:** 0.3 µsec **BYTE/ WORD/ DWORD/ REAL/ Move:** 0.3 / 0.3 / 0.1 / 0.1 µs
Number of I/O points	**Main unit:** up to 32 digital I/Os. Can be extended up to 48 digital I/Os. Digital I/O status indication on LCD.
Extensions units	Up to 2 I/O extension units and Up to 2 COM extension units
Marker memory	4 Kbytes
Data memory	24 Kbytes
Retain memory	1 Kbytes 700 bytes of data memory and 300 bytes of marker memory (%MB3700 to %MB3999) Stored in FRAM type of memory. **Battery free operation.**
Application program memory	192 Kbytes Flash (for PLC application as well as HMI application) 384 Program Organization Units (POUs) maximum POU Max size allowed is 8192 bytes
App data memory	64000 bytes Flash memory. *1 Storage of HMI specific features like Multilingual data and Data logging configuration.
External memory support	Yes *2 microSD card

*1 Data logging configuration consumes up to 1KB memory of app data memory. So remaining memory is available for multilingual data.

*2 microSD card slot is available only on GOC Main units dispatched from June 2016. Refer section 'microSD memory' for more details.

Technical specifications…

Item		Description
Source code memory		1.5 Mbytes Flash Stores project file with symbolic names, comments, visualization screens , libraries (optionally)
Application program security		Password protection supported for 1. Project file/ POU read or write 2. Source code upload
Timers		Number of instances (TON, TOFF, TP) can be called. Limited by available data memory only. *3
Counter		Number of instances (CTU, CTD, CTUD) can be called. (Limited by available data memory only) *3
Real Time Clock		Onboard
		Super capacitor backup: 2 weeks duration nominal at 25°C ambient
		Max error: ± 2 Secs max per day
Interrupt events	Periodic	1, 0.5/ 1ms resolution
		Settable from 0.5/ 1 ms to 1250/ 2500 ms.
	Counter compare	Up to 2 for counter/encoder configuration for inputs I00 and I03
	Hardware input	Up to 2 Rising edge and falling edge for configuration of input I02 and I05
		Pulse ON time: 100 µsec minimum **Pulse OFF time:** 2 msec minimum
Special functionality for digital inputs on Main unit (User configurable)		**Single phase counter:** 20 KHz – Up to 2 Counter0: I00 Counter3: I03
		Encoder (A, B, Z) interfaces: 10 KHz – Up to 2 Encoder1: I00 (A), I01(B), I02(Z) Encoder3: I03 (A), I04(B), I05(Z)
		Pulse catch: Up to 2, for input02 and input05 **Minimum pulse width**: 500 µSec.
		Frequency measurement: Up to 2, for input I02 and input I05 **Input Frequency:** 0.1 Hz to 20 KHz **Pulse ON / OFF time:** 20 µsec minimum **Accuracy:** 0.5 % of full scale
		Pulse ON time measurement: Up to 2, for input I02 and input I05 **Pulse duration:** 200 µsec to 10 Sec **Accuracy:** 2.5 % of full scale
Operating Modes		RUN, STOP and Bootstrap (for new firmware download)
HMI *4	Display	128 x 64 pixel graphics LCD with backlit LCD with white characters and blue backlit **View size:** 70.7 x 38.8 in mm
	Keypad	10 keys for display navigation and data entry (User configurable functionality for 5 keys; F1 to F5)
	Illuminated Keys	8 illuminated keys 4 x 2 key matrix with dual colored LED (Red, Green)
	Slide-in label	Insertable label over illuminated keys

*3 FB instance can be declared retentive and entire instance data is retained. (limited by available retentive memory)

*4 Even though, maximum 64 user defined screens are allowed, it is limited by application program memory (192 Kbytes) and POU size (8192 bytes) because display related logic i.e. HMI screen definition and key configuration is a part of application program and is available in POU HMI_PRG.

Technical specifications…

Item	Description
Programming	
Port	Through - Ethernet COM extension unit (GC-ENET-COM) or - RS232 serial COM extension unit (GC-RS232-COM) fixed in COM1 slot.
Remote operation	Remote programming and online monitoring via Ethernet port
Standard	IEC 61131-3
Software	Windows based 'CoDeSys' V2.3
Languages	IL, LD, FBD, SFC, ST, CFC
Debugging and Online Monitoring	Visualization, Forcing, Writing, Watch and Receipt for PLC variables.
Online change	Not supported
Offline simulation	Not supported
Integtated configuration tools (with programming software CoDeSys V2.3)	Hardware Configuration Tool for ▪ Configuration and parameterization of Main Unit, I/O extension, COM extension with user friendly GUI. ▪ Special functionality configuration: - Counter, encoder, counter compare interrupt, hardware input interrupt, pulse catch, frequency measurement, pulse on time measurement ▪ Serial port parameterization and protocol configuration ▪ Analog I/O configuration and handling
	HMI Configuration Tool for ▪ Fast screen development with drag n drop objects and properties ▪ Configurable operation for function keys ▪ Configurable operation for illuminated keys and control of dual-colour LEDs ▪ Screen actions and chaining ▪ Multi-language configuration ▪ Custom image import ▪ Configuration of data logging using SD memory card
Firmware update	Possible on field by MEI authorized personnel. Via PC based Bootstrap utility through programming communication port

User can have HSC functionality and Hardware input rising/ falling edge interrupt functionality either from Main unit (channel numbers 0/3) or I/O Extension unit (channel numbers 10/13) GC-4HSXPTY at a time.

When any special input functionality like encoder with Z input etc. (excluding functionality like pulse catch/ frequency measurement/ pulse ON time measurement) is configured on Main unit, user should not fix I/O extension unit GC-4HSXPTY in IO1 slot. User is informed and care is taken during configuration using ***Hardware Configuration Tool***.

2 Installation and Wiring

This chapter discusses about installation and wiring of Main unit, IO extension unit and COM extension unit. It also explains installation and removal of microSD card in the Main unit.

2.1 Dimensional Details

This section provides dimensional details of various parts of GOC such as Main unit, I/O extension unit and COM extension unit. These details helps user during mounting of Main unit and extension units in the control panel.

2.1.1 Main Unit

The figure below shows all the views of bare Main unit with dimensional details. All dimensions are in mm.

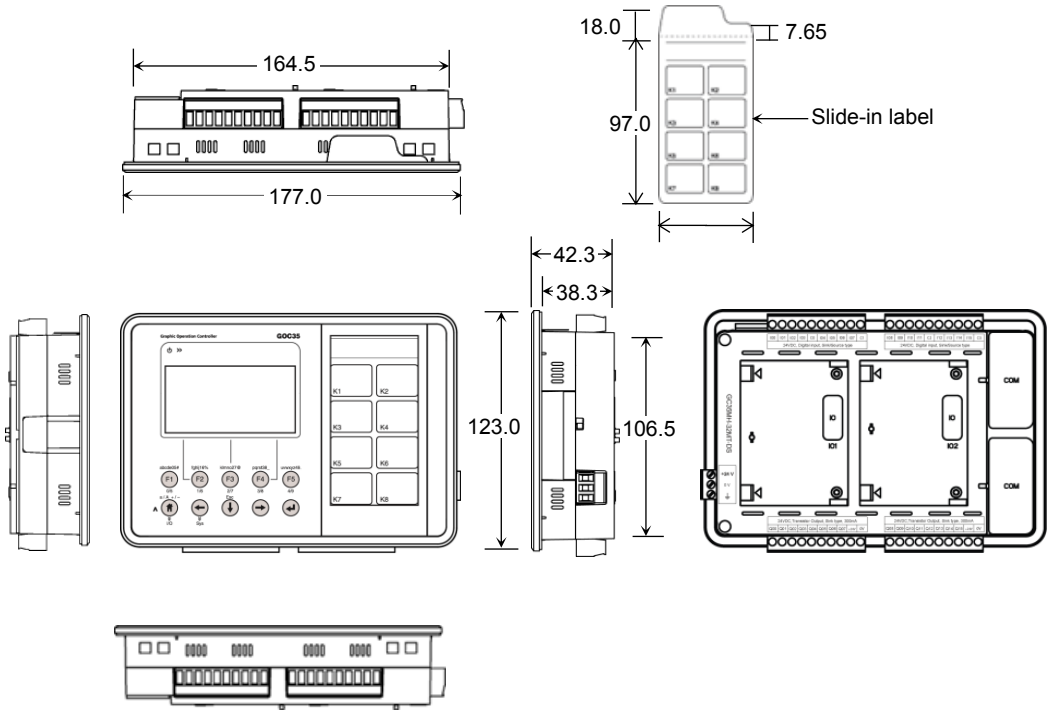


Figure 5: Main unit dimensions

2.1.2 I/O Extension Unit

User can attach up to 2 I/O extension units on the back side of Main unit. The figure below shows all the views of I/O extension unit with dimensional details. All the dimensions are in mm.

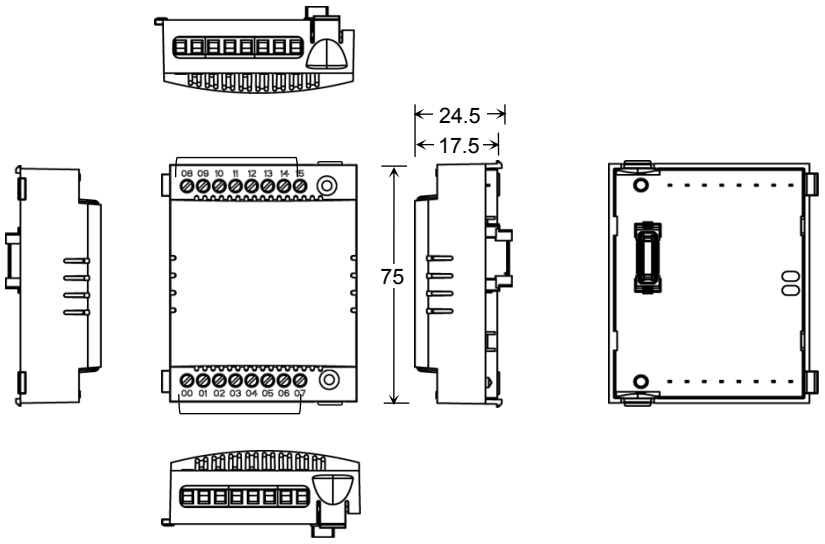


Figure 6: I/O extension unit dimensions

2.1.3 COM Extension Unit

User can attach up to 2 COM extension units on the back side of Main unit. The figure below shows all the views of COM extension unit with dimensional details. All the dimensions are in mm.

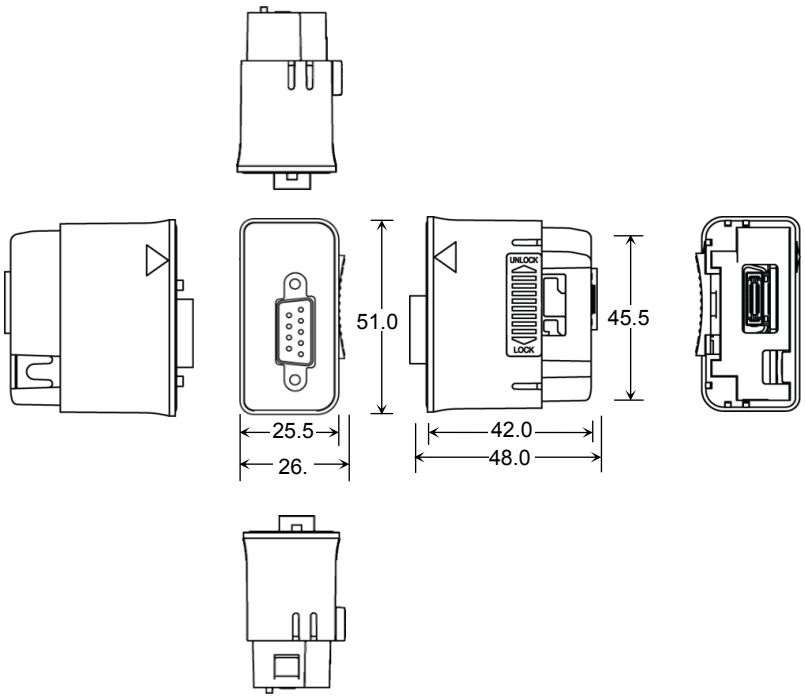


Figure 7: COM extension unit dimensions

Dimensional details in Figure 7 shows GC-RS232-COM extension unit. Similar plastic enclosure is used for other COM extension units like GC-RS422-COM and GC-ENET-COM but with different interface connectors. So there is small change in dimensions due to connector used for external communication interface.

2.1.4 Main Unit with Extension Units

User can attach up to 2 I/O extension units and up to 2 COM extension units on the back side of Main unit. The figure below shows all the views of Main unit attached with 2 I/O extension units and 2 COM extension units with dimensional details. All the dimensions are in mm.

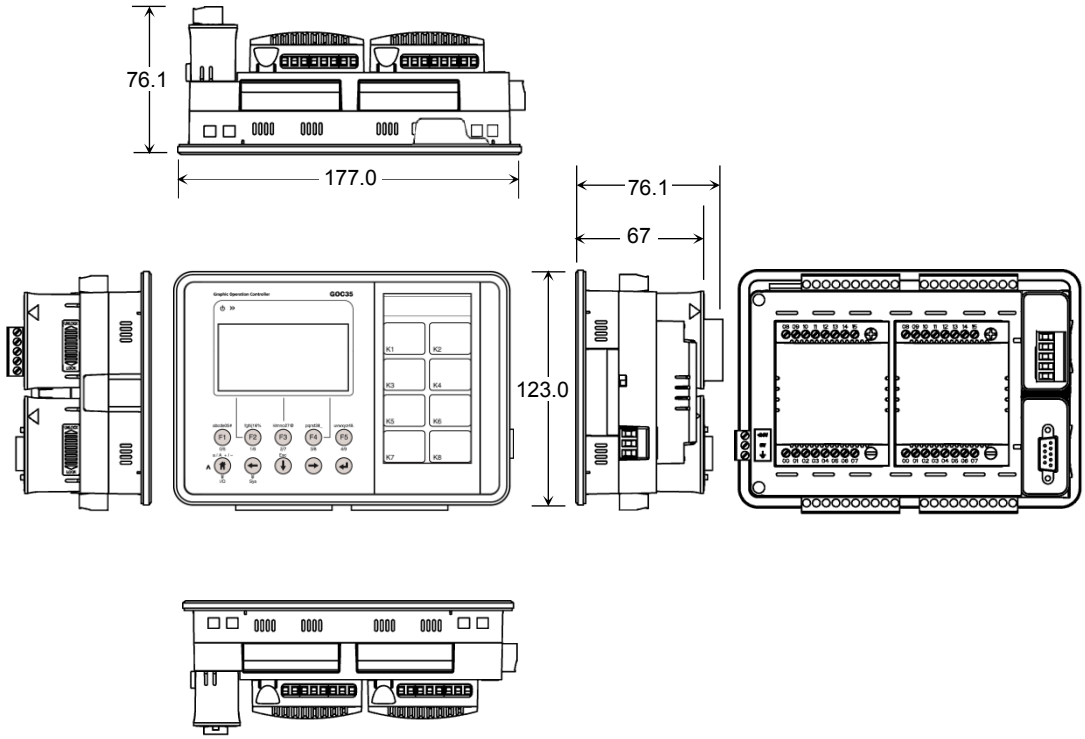


Figure 8: Main unit dimensions with extension units

2.2 Installation

This section provides recommendations and precautions to be observed during installation of various units of GOC.

2.2.1 Installation Recommendations

GOC is a front panel mount controller. Install the controller in an environment conforming to the general specifications and installation recommendations and precautions.

The recommendations are as below.

1. Mount controller on a firm, plane and conducting surface. Installation in orientation other than recommended one (as shown in adjuscent figure), may cause overheating, damage and malfunctioning of the controller.

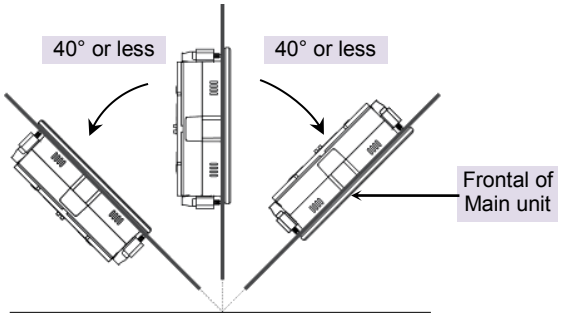


Figure 9: Mounting orientation

2. Mount controller on non-vibrating surfaces and should be protected by rubber pads so that the shock is not felt.
3. Mounting plate thickness should not exceed 4 mm.
4. Programming of the controller is possible by attaching COM extension (GC-ENET-COM or GC-RS232-COM) unit in COM1 slot. This unit is optional and should be purchased separately. Installation should take care of keeping free space considering depth of controller with COM extension unit installed on it i.e. 90 mm inclusive of additionl space required for communication cable routing.

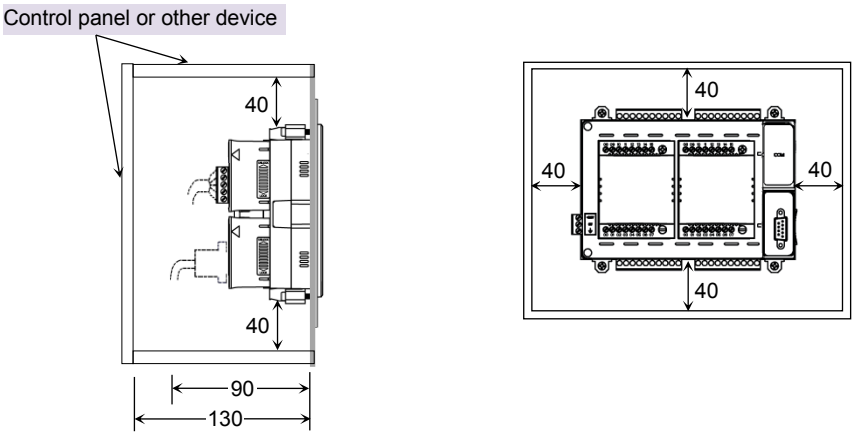


Figure 10: Mounting recommendations

5. Ensure the gap of 40 mm between controller and cabinet walls, other equipments and wiring duct.
6. Leave a minimum space of 40 mm around the Main unit to facilitate air circulation for heat transfer by natural convection and easy fixing and removal of unit.

SD card slot is provided on right side when viewed from the backside i.e. on COM extension slot side. Keep sufficient free space (minimum 10-15 cm) on right side for insertion and removal of microSD card, if microSD is in use.

2.2.2 Precautions to be taken

This section lists out general precautions to be observed during installation.

1. Make sure to cut off all the phases of the power supply externally before attempting installation or wiring work. Failure to do so may cause electric shock or damage to the product.

   Back side of I/O extension unit is open. Do not remove I/O extension unit especially relay output extension unit with AC power in power on condition. It may cause electric shock.

2. Maintain proper thermal distances between equipments producing heat (like heaters, transformers, etc) inside the control panel. Do not install controller above such equipments.
3. Exposure to humid environment for a long time can reduce component life. It may cause corrosion of electrical and electronic components, or may lead to shorts or malfunctions. Do not expose controller to humid atmosphere for an extended period.
4. Avoid controller exposure to excessive or continuous vibrations or shocks. Failure to do so may cause disenguagement of PCB components, sockets, on-board soldered components etc. from their counter positions.
5. Cover unused slots (IO and COM) to protect them against dust, moisture and ESD (Electric Static Discharge).
6. Use controller within the range of general and technical specifications.

2.2.3 Main Unit Installation

This section provides steps to mount Main unit on front panel as well as removes it.

Product packaging consists of Main unit (with all the terminal blocks attached), installation manual, mounting template and 4 mounting clamps.

Before installation and removal, refer sections **Installation Recommendations** and **Precautions to be observed** from *Installation and Wiring*. Failure to follow recommendations and precautions to be observed may cause electric shock or damage to the product.

Mounting Main Unit

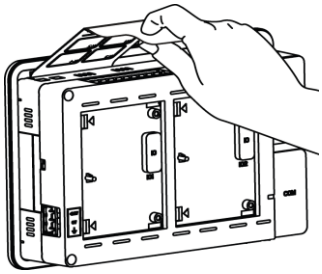
Follow the steps below to mount main unit on front panel.

1. Prepare Main unit for mounting

   Detach all the terminal blocks (10-pin I/O terminal blocks and 3-pin power supply terminal block) from Main unit. Make sure that gasket on outer periphery of front panel backside is in place.

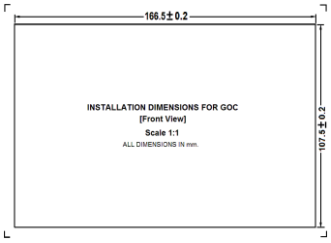
2. Insert slide-in label

   Normally, Main unit is provided with default slide-in label inserted. But user can remove it and insert customized label. Slit is provided to insert slide-in label. It is located at left top on the backside of Main unit.
   Insert label from side opposite to the fold such that fold line is aligned with slit edge. Shape of folded part is designed such that it does not obstruct mounding clamp.



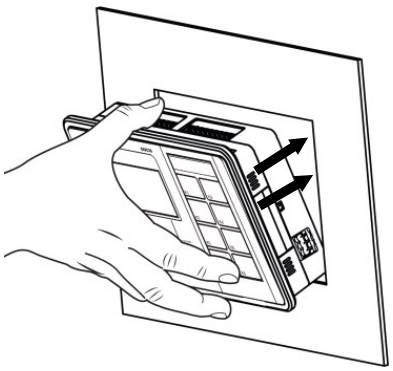
3. Make cut out

   Remove adhesive tapes provided at corners of backside of mounting template and stick the mounting template on front panel where Main unit is to be mounted**.** Mark 4 corners of the rectangular cut-out and make a cut out. Dimensions of cut out are 166.5 X 107.5 mm.



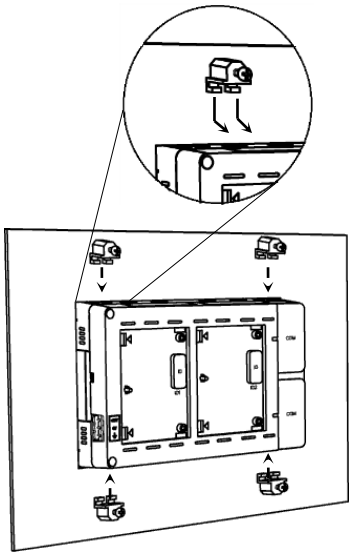
4. Insert Main unit through cut-out

   Insert Main unit from outside through cut out on panel. Make sure that folded part of slide-in label is passed through the cut out. Hold Main unit by hand from outer side of the panel so that it will not fall during fitment of mounting clamps.

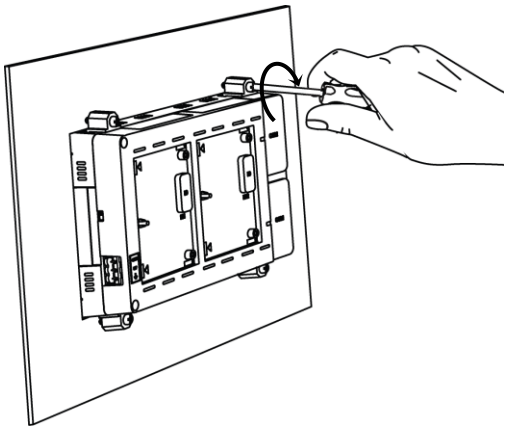


5. Tighten mounting clamp

   2 rectangular openings are provided near each corner on the extended part on back side. Insert 'L' shaped projections of mounting clamp into matching slots and pull it away from panel to engage it into the slots.

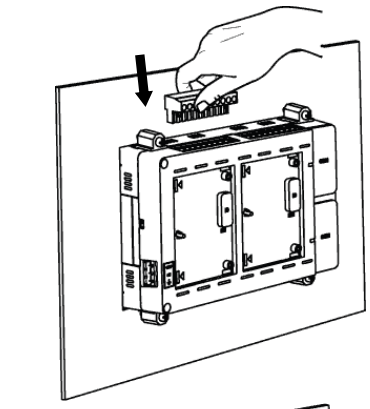


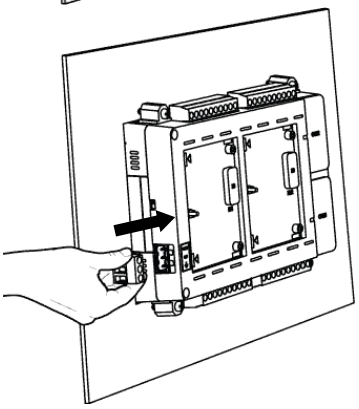
   Turn mounting clamp screw in clockwise direction till tip of screw slightly touches inner surface of panel. Rotate screw an additional ¼ - ½ turn in clockwise direction and ensure controller is firmly mounted in the panel. Fix all the 4 mounting clamps by tightening screws one by one progressively.



6. Insert terminal blocks

   Insert 10-pin input terminal blocks/s at upper side.
   Insert 10-pin output terminal block/s at lower side.
   Insert 3-pin power supply terminal block.





Removing Main Unit

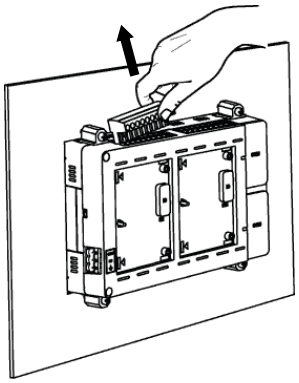
Follow the steps below to remove Main unit from front panel.

7. Preparation for removal

   Cut off all the phases of the power supply to the control panel.

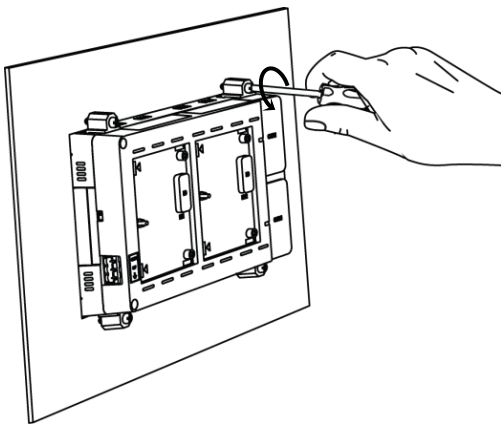
   Remove 3 pin power supply terminal block.

   Remove all the I/O terminal blocks. For removal, pull terminal block from one side first. Once this part is out, pull remaining part easily.



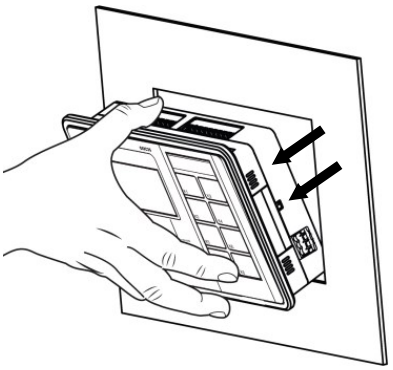
8. Removing mounting clamps

   Turn mounting clamp screws in anti-clockwise direction to loosen it.
   Push body of clamp towards panel to disengage it from matching slots on the Main unit.
   Pull body of clamps off the Main unit. Hold Main unit with one hand while undoing last of the clamps.



9. Removal of Main unit

   After removing all mounting clamps, pull out unit from outside, to remove it from cut out.



2.2.4 I/O Extension Unit Installation

User can attach up to 2 I/O extension units on the back side of Main unit and extend no. of I/Os as per application requirement. This unit is optional and should be procured separately.This section explains mounting and removal of I/O extension unit.

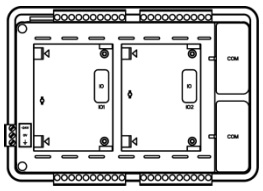
Product packaging consists of I/O extension unit, installation manual and 2 self-tapping screws (M3 x 10 mm) for fixing I/O extension unit on Main unit.

Before installation and removal, refer sections **Installation Recommendations** and **Precautions to be observed** from *Installation and Wiring*. Failure to follow recommendations and precautions to be observed may cause electric shock or damage to the product.

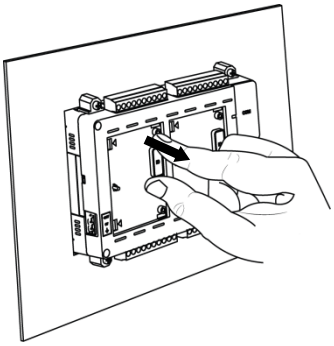
Fixing of I/O Extension Unit

1. Prepare Main unit for fixing I/O extension unit

   Main unit is provided with covers fixed on IO interface connectors and COM slots.

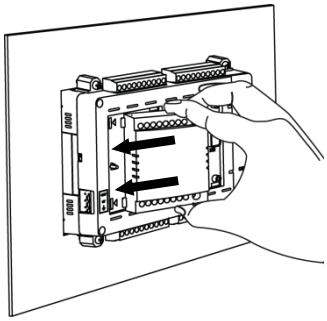


Remove interface connector cover on IO slot interface connector on Main unit.

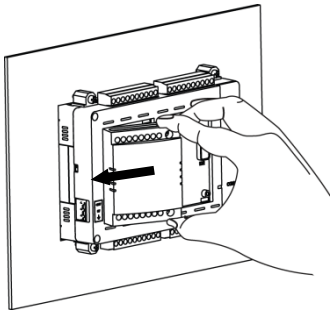


2. Fixing I/O extension unit

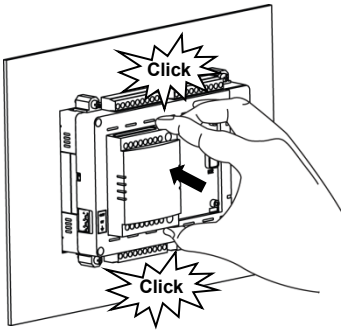
Hold I/O extension unit between thumb and pointing finger with latches on left side. Ensure that unit left side part is tilted towards Main unit by 30 degrees approximately. Otherwise, unit backside will obstruct projection provided at left side of slot area.



Insert both latches in respective openings on left side of desired IO slot (IO1/IO2) on Main unit and slide unit to left to insert latches completely inside openings.

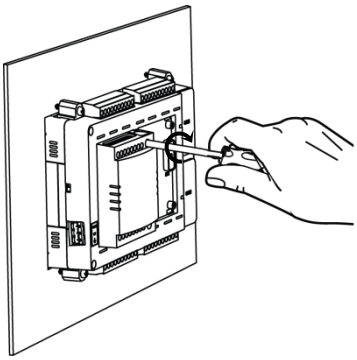


Push right side part of I/O extension unit towards Main unit till unit interface connector gets engaged with its male counterpart on Main unit.
Projection provided on slot area on Main unit is accommodated through the oval shaped hole on PCB.
Then push right side further gently till both the unit clips are clicked.



3. Tighten self-tapping screws

Insert self-tapping screws (M3 x 10 mm, dispatched along with I/O extension unit) in unit fixing screw holes and tighten it to prevent effect of vibrations. Main unit mounted on slanted front panel may require fitting with screws.

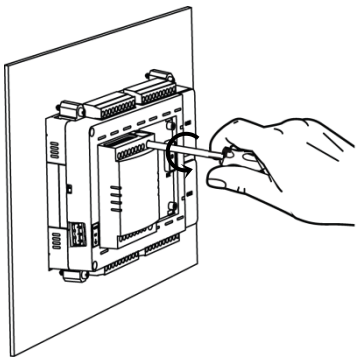


Do not use any other screw of different size to fix the I/O extension unit on Main unit. Incorrect handling and installation of I/O extension unit may cause malfunctioning and/or damage to the hardware.

Removal of I/O Extension Unit

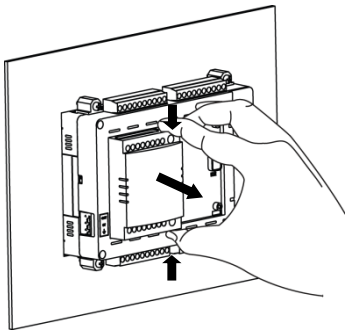
Firstly, remove I/O wiring from I/O terminal blocks.

4. Un-tighten self-tapping screws. Untighten both mounting screws fully if already fitted. Do not try to pull out extension unit forcefully with tightened fixing screws. It may cause damage to the hardware/ plastic enclosure.

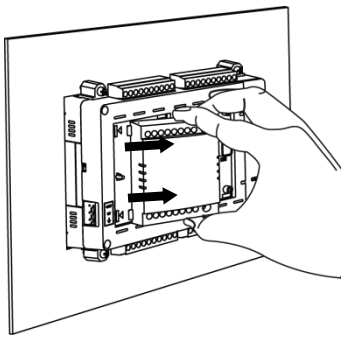


5. Removal of I/O extension unit

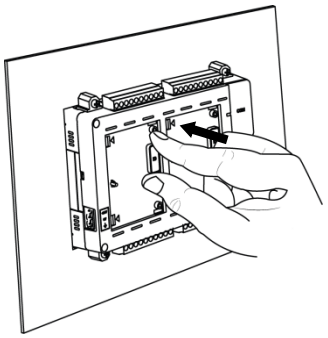
   Keep thumb on bottom clip and pointing finger on top clip. Push both I/O extension unit clips inside so that they are unlocked from respective slot openings. Pull right side of I/O extension unit away from Main unit so that interface connector gets disengaged.



   Slide I/O extension unit towards right side such that both latches on left side come out of respective openings on left side of slot area. Lift I/O extension unit away from Main unit to remove it from IO slot.



6. Ensure that cover is fitted on interface connector of unused IO slot.



2.2.5 COM Extension Unit Installation

User can attach up to 2 COM extension units on the back side of Main unit and interface third party devices. Programming of the controller is possible by attaching COM extension unit (GC-ENET-COM or GC-RS232-COM) unit in COM 1 slot. It can be fixed during programming and commissioning and can be removed later on. This unit is optional and should be procured separately.This section explains mounting and removal of COM extension unit.

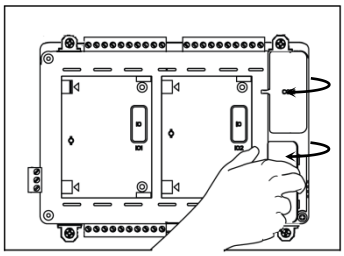
Product packaging consists of COM extension unit and installation manual. For GC-RS422-COM unit, 5-pin terminal block is attached to the unit.

> ⚠ Before installation and removal, refer sections **Installation Recommendations** and **Precautions to be observed** from *Installation and Wiring*. Failure to follow recommendations and precautions to be observed may cause electric shock or damage to the product.

Fixing of COM Extension Unit

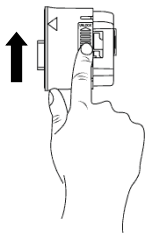
1. Prepare Main unit for fixing COM extension unit

   Remove slot cover from Main unit. Cover remains attached on Main unit due to hinge on left side.

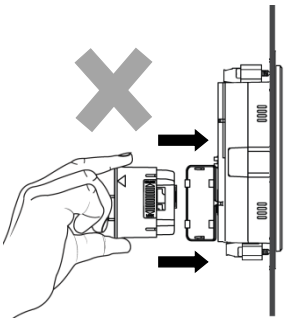


2. Prepare I/O extension unit for fixing

   Make sure that locking clip on right side of COM extension unit is pushed upward completely before fixing it in the slot on Main unit.

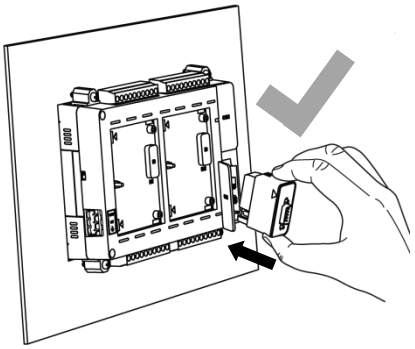


3. COM extension unit cannot be inserted in the COM slot with straight orientation.

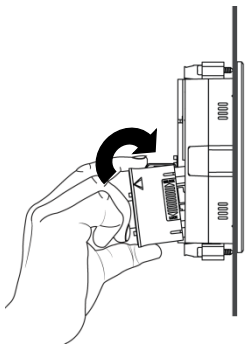


4. Insert COM extension unit in Main unit COM slot

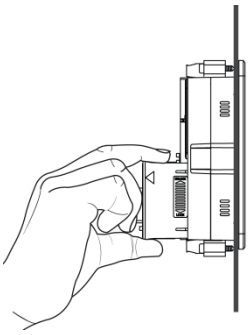
   Hold COM extension unit with thumb on bottom front edge and pointing finger on upper front edge with unit locking clip on right side. Hold it in tilted position such that bottom side gets inserted first.



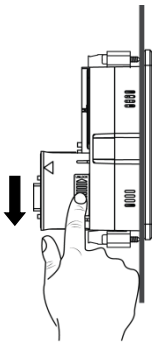
   Firstly, insert bottom part of COM extension unit through slot



   Then push upper part gently so that interface connector gets engaged with its male counterpart on Main unit.



   Push unit locking clip downward fully so that COM extension unit is locked firmly on the Main unit. Failing to do so will cause loose connection and sudden removal of COM extension unit from the slot.



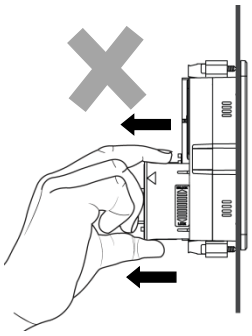
⚠	Incorrect handling and installation of COM extension unit may cause malfunctioning and/or damage to the hardware/plastic enclosure.

Removal of COM Extension Unit

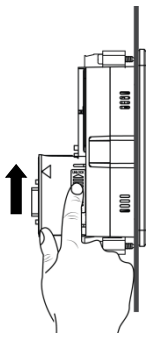
5. Prepare COM extension unit for removal

   Remove communication cable connected to COM extension unit.

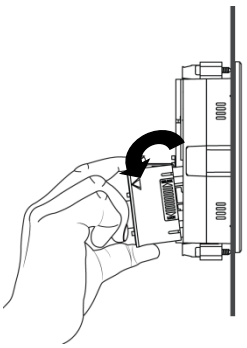
6. Do not try to pull COM extension unit with unit locking clip in downward position. It may cause damage to COM extension unit as well as Main unit



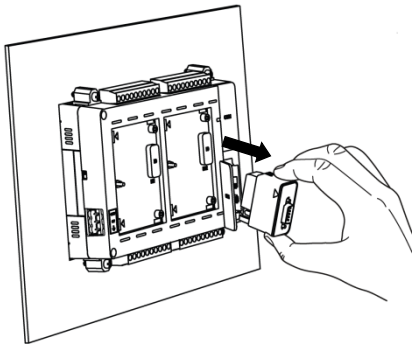
7. Removal of COM extension unit
   Push unit locking clip upward fully first.



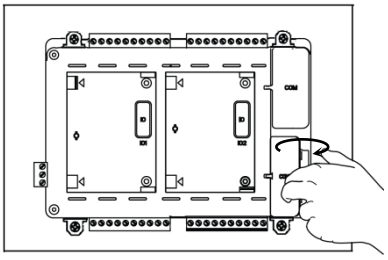
   Keep thumb on bottom edge of front and pointing finger on top edge of front. Pull upper part of unit, so that its interface connector gets disengaged from its counterpart on Main unit.



   Then take out unit out of the COM slot.



8. Ensure that cover is fitted on interface connector of unused COM slot.



2.2.6 microSD card Installation

All the Main units are equipped with a SD card slot on its right side wall (i.e. on the side of COM extension slot) when looking from the backside.

It is recommended to leave sufficient space (approximately 10 to 15 cm) on right side of Main unit to facilitate easy insertion and removal of microSD card if used.

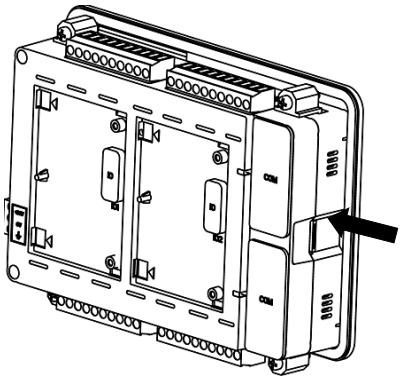
User can insert commercially available microSD card in this slot. Specifications of compatible microSD cards are mentioned in section 'microSD Memory'

> ⚠ microSD card slot is available only on GOC main units manufactured from June 2016.
>
> Month and year of manufacturing can be decoded from 11 digit product serial number.
>
> In serial number of GOC, third and fourth digit designate year of manufacturing (16 for year 2016) and fifth digit is month of manufacturing as 1 for Jan, 2 for Feb….9 for September and 'A' as October, B as November and C as December.

Insertion of microSD card

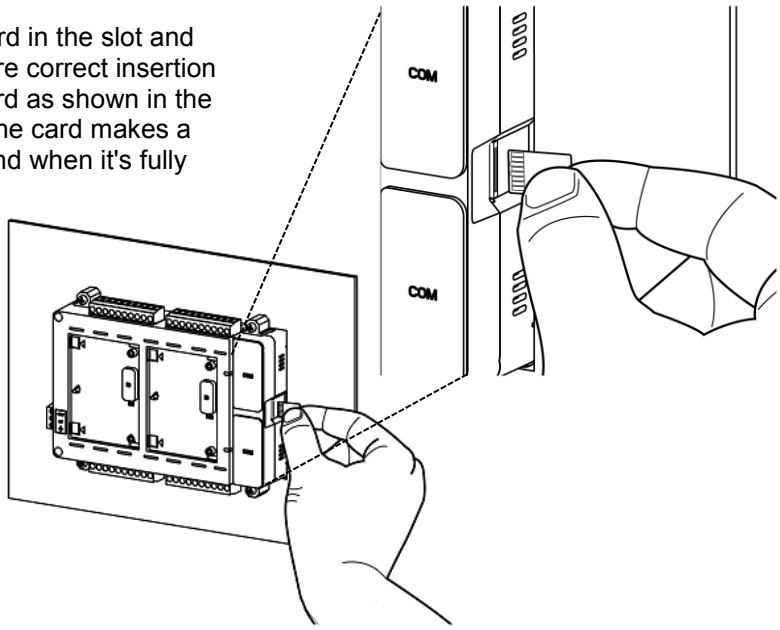
1. Locate microSD card slot on Main unit

   microSD card slot is provided on right side wall of the Main unit when looking from the back side.



2. Inserting microSD card

   Insert microSD card in the slot and push inside. Ensure correct insertion direction of the card as shown in the expanded view. The card makes a slight clicking sound when it's fully inserted.



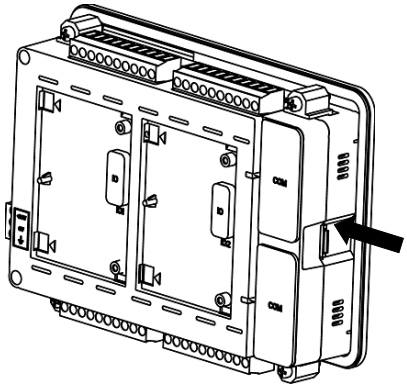
3. After correct insertion, SD card is detected and mounted automatically. Status can be monitored in system menu. Refer chapter 'System menu'.

> ⚠ Incorrect insertion of microSD card may cause malfunctioning and/or damage to the Main unit hardware or card itself.

Removal of microSD card

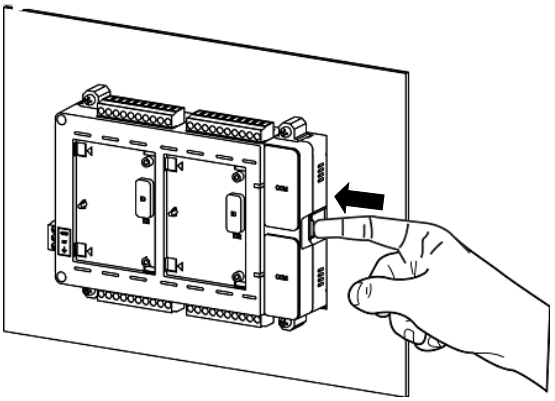
1. Unmount card

   If you are removing SD card when Main unit is powered up, make sure that SD card is un-mounted first. Refer chapter 'System Menu' for un-mounting procedure.

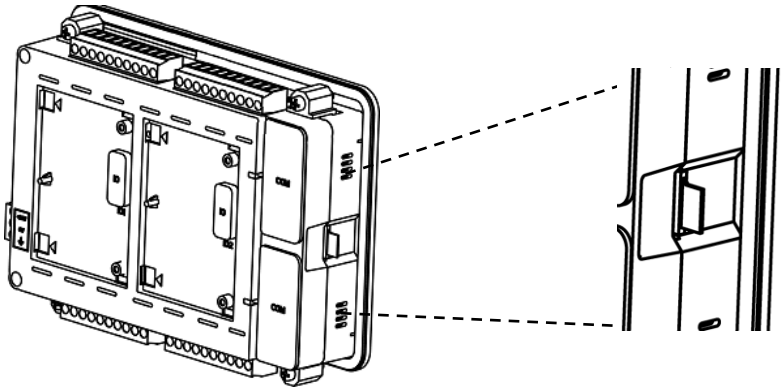


2. Locate SD card slot on side wall of Main unit.
   microSD card slot is provided on right side wall of the Main unit when looking from the back side.

3. Gently push SD card inside till it makes clicking sound of unlock.
   Release finger after clicking sound.



4. microSD card comes out from card holder**.**
   Now microSD card can be pulled out easily.



Accidental removal of SD card before un-mounting may cause malfunctioning, loss of log data and/or damage to microSD card.

2.3 Wiring

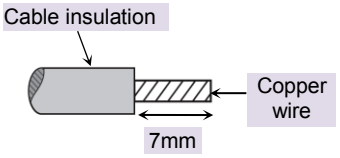
This chapter provides recommendations and precautions to be observed during wiring of entire controller. GOC consists of Main unit, IO extension unit and COM extension unit. For wiring of individual unit, refer subsequent chapters specific to individual unit type and model.

2.3.1 Recommendations

Cable

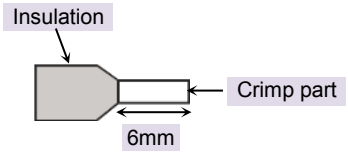
Terminal block pitch size is 5.08 mm.

Use stranded (flexible) or solid wire of size 0.5 to 1 mm² (AWG 22 to 18). Strip insulation of stranded wire and twist the strands to prevent it from spreading and crimp the lug.



Lug

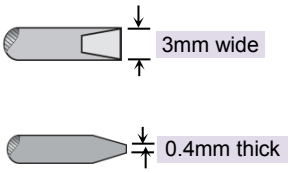
The adjacent figure shows recommended size of lug.



Screw driver

Terminal block screw size is M3. For tightening terminal, use flat blade screw driver. The figure shows desired size of screwdriver blade.

The tightening torque should not exceed 0.50 Nm.



2.3.2 Precautions to be taken

1. Make sure to cut off all phases of the power supply externally before attempting installation or wiring work. Failure to do so may cause electric shock or damage to the product.
2. Do not use wire without lug. Do not solder-plate the wire ends.It may cause loose connection. Ensure that only one lug is connected to one terminal.
3. Ensure that size of wire and lug used are as per the specifications. Use screw driver with specified size of tip. Tightening torque should be as per the specifications.
4. Ensure the gap of 40 mm between controller and cabinet walls, other equipments and wiring duct This will help in natural cooling of controller and also easy mounting or unmounting of hardware.
5. Separate wiring by signal types. Bundle wiring with similar electrical characteristics together.
6. Differnetiate wiring with different electrical characteristics by coloured insulations e.g. AC wiring and DC wiring
7. Make sure that there is a separate bundle and routing for input and output wires. Fixup the wire bundle with support on panel so that there is no stress on wires and subsequently on unit. Ensure that bunch is routed properly and wires are not kept hanging.
8. Do not bundle 24 VDC I/O wires with main control panel wiring.
9. Do not bundle cable carrying low level signals like communication and analog signals with input output wiring and control panel wiring.

10. 50 to 100 meter long wiring for input/output will not cause any problems of noise but, generally, the wiring length should not exceed 30 meters to ensure the safety. For longer distance, route the input and output signal lines separately.
11. Ensure that length of wire that connects 24 VDC power supply to I/O unit is less than 3 meters. Locate 24 VDC power supply near to the controller.

2.3.3 Guidelines for Earthing

1. Connect EARTH (Symbol) terminal directly to clean earth in the control panel avoiding ground loops.
2. Ensure Class D grounding. (Grounding resistance: 100 Ω or less)
3. Ground the controller independently. If it cannot be grounded independently, ground it jointly as shown below.

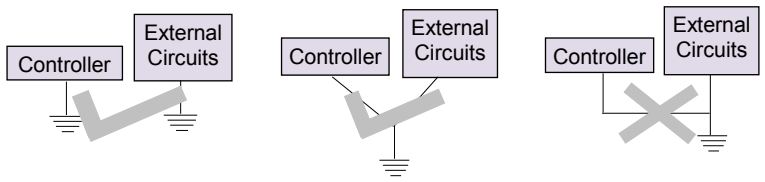


Figure 11: Unit Earthing

4. Ensure that EARTH cable is thick and short as far as possible to provide low impedance path.
5. If EARTH is not connected, it may result in electric shock or erroneous operation.

It is recommended to twist power supply cable to minimize adverse effects of noise.

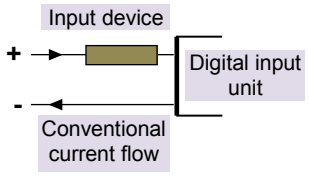
2.3.4 Digital input sink/source operation

The term sourcing and sinking applicable to digital input section refer to the manner in which input device is wired to input section.

Sink type of input connection

For this, typically +24 Vdc is connected to input device. When this device is active, +24 Vdc is available at input terminal on unit.

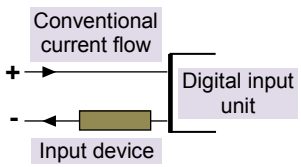
The ground of 24 Vdc supply is connected to common point on input section. So input device in active state supplies current to input circuit. As input circuit is receiving current, it is said to be sinking. Normally, PNP type of devices (e.g. proximity switches) are connected in this fashion.



Source type of input connection

For this, typically ground of 24 Vdc power supply is connected to input device. +24 Vdc is connected to common point on input section.

When this device is active, current flows through input circuit and passes through input device. As input circuit is supplying current, it is said to be sourcing. Normally, NPN type of devices (e.g. proximity switches) are connected in this fashion.



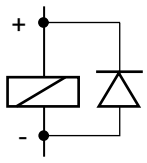
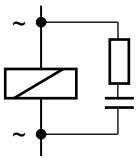
2.3.5 Guidelines for using digital outputs

Noise suppressors

It is extremely important to connect noise suppressors directly across all inductive load (relays, contactors, solenoid valves, etc.) irrespective of whether it is actuated by PLC output section or actuated externally. The inductive loads generate strong electrical noise that may affect PLC operation.

The noise suppressor should be mounted close to the load, as a rule, should not be away more than 0.5 meters. This helps in attenuating noise at the source it self.

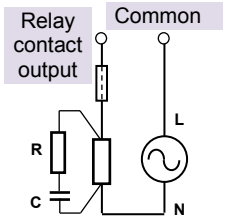
In case of resistive loads (incandescent lamps, LED lamps, heating resistors, etc.), It is not necessary to use noise suppressors.

Noise suppressor for DC devices	**Noise suppressor for AC devices**
DC operated devices are protected by diode connected a shown below 	AC operated devices are protected by RC snubber as shown below 

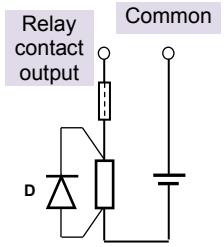
2.3.6 Guidelines for using relay outputs

- External fuse links or fused terminals are recommended for relay output wiring to avoid any burnout of internal copper tracks due to excessive current flow due to external short circuit, overload or inductive surges.
- The life of relay contacts can be enhanced by the use of RC snubber (spark quenchers) across the AC load. A suggested combination for of the R and C could be R=220Ω/ Half watt and C=0.1 µF/1000 Volts.
- For DC loads a free-wheeling diode such as 1N4007 should be used in reverse polarity to avoid effects of back EMFs generated by inductive load.
- The diode and the snubber should be positioned and wired up as near as possible to the external load for maximum effect.

RC Snubber for AC load



Diode for DC load



GOC Main unit model GC35MH-32MR-D provides 2 output terminal blocks.
Model GC35MH-16MR-D provides 1 output terminal block.

3 Main Unit

6 models of Main unit are available depending upon number of I/Os and type of outputs.

Main unit model	No. of inputs	No. of outputs	Type of output
GC35MH-32MT-DS	16	16	Transistor (sink)
GC35MH-32MT-DSS	16	16	Transistor (source)
GC35MH-32MR-D	16	16	Relay
GC35MH-16MT-DS	8	8	Transistor (sink)
GC35MH-16MT-DSS	8	8	Transistor (source)
GC35MH-16MR-D	8	8	Relay

3.1 I/O Specifications

This section provides specifications of digital inputs and outputs built in Main unit.

3.1.1 Digital Input Specifications

Item	Description	
Number of inputs	16 for GC35MH-32MT-DS, GC35MH-32MT-DSS, GC35MH-32MR-D 8 for GC35MH-16MT-DS, GC35MH-16MT-DSS, GC35MH-16MR-D	
Voltage rating	24 VDC (18 to 30 V including ripple)	
Type	Sink or Source in group of 4, with one common per group	
ON voltage level	18 VDC minimum	
OFF voltage level	5 VDC maximum	
ON/ OFF Current	**ON current**: 6 mA at 24 VDC	**OFF current**: 3.8 mA at 24 VDC
Input impedance	5.2 KΩ	
Transition delay	10 ms (filter time)	
Isolation	Optical 1.5 KV between input and internal circuit 1.5 KV between groups. Nil between individual input points in a group.	
I/O terminal blocks	Two 10-pin, screw type removable terminal blocks for GC35MH-32MT-DS, GC35MH-32MT-DSS and GC35MH-32MR-D One 10-pin, screw type removable terminal block for GC35MH-16MT-DS, GC35MH-16MT-DSS and GC35MH-16MR-D	

Digital input specifications…

Special functions of digital inputs (User configurable)				
Single phase counters (up to 2 nos.)	**Counter**	**Input**		
	Counter0	input I00		
	Counter3	input I03		
	Input frequency: 20 KHz maximum			
	Pulse ON/ OFF time: 20 µsec minimum			
Quadrature encoder (Up to 2 nos.)	**Encoder**	**A phase**	**B phase**	**Z marker**
	Encoder0	input I00	input I01	input I02
	Encoder3	input I03	input I04	input I05
	Input frequency: 10 KHz maximum (for individual phase)			
	Pulse ON / OFF time for A and B phase: 20 µsec minimum. **Pulse ON / OFF time for Z marker pulse:** 50 µsec minimum.			
Interrupt events	**Counter compare** EVENT_HSC0_CMP for counter0 or encoder0 EVENT_HSC3_CMP for counter3 or encoder3			
	Rising edge and falling edge EVENT_IX2_R_TRIG and EVENT_IX2_F_TRIG for digital input I02, EVENT_IX5_R_TRIG and EVENT_IX5_F_TRIG for digital input I05			
Pulse Catch	Up to 2, User configurable for input02 and input05			
	Minimum pulse width: 500 µsec			
Frequency Measurement	Up to 2, User configurable for input I02 and input I05, **Input Frequency**: 0.1 Hz to 20 KHz **Pulse ON / OFF time**: 20 µsec minimum **Accuracy**: 0.5 % of full scale			
Pulse ON Time Measurement	Up to 2, User configurable for input I02 and input I05, **Pulse duration**: 200 µsec to 10 secs **Accuracy:** 2.5% of full scale			

3.1.2 Transistor Output (Sink) Specifications

Item	Description	
Number of outputs	16 for GC35MH-32MT-DS 8 for GC35MH-16MT-DS	
Type of output	Transistor sink type	
Voltage rating	24 VDC (18 to 30 VDC including ripple)	
Current rating	300 mA per point 1 common per group of 8 outputs	
On voltage drop	0.6 VDC maximum	
Off state leakage current	5 µA maximum	
Response time	**OFF to ON**	82 µs
	ON to OFF	92 µs
Isolation	Optical 1.5 KV between input and internal circuit	
Protection	Output short circuit protection	
	Fast demagnetization for inductive loads	
Load supply	24 VDC (18 to 30 VDC including ripple)	
	Reverse polarity protection	
I/O terminal blocks	For **GC35MH-32MT-DS**, 2 nos., 10-pin terminal block, removable screw type, located at lower side of unit. For **GC35MH-16MT-DS**, 1 no., 10-pin terminal block, removable screw type, located at left lower side of unit.	

3.1.3 Transistor Output (Source) Specifications

Item	Description	
Number of outputs	16 for GC35MH-32MT-DSS 8 for GC35MH-16MT-DSS	
Type of output	Transistor source type	
Voltage rating	24VDC (18 to 30 VDC including ripple)	
Current rating	300 mA per point 1 common per group of 8 outputs Paralleling of outputs is possible	
On voltage drop	0.6 VDC maximum	
Off state leakage current	10 µA maximum	
Response time	**OFF to ON**	250 µs
	ON to OFF	300 µs
Isolation	Optical 1.5 KV between input and internal circuit	
Protection	Output short circuit protection	
	Fast demagnetization for inductive loads	
Load supply	24 VDC (18 to 30 VDC including ripple)	
	Reverse polarity protection	
I/O terminal blocks	For **GC35MH-32MT-DSS**, 2 nos., 10-pin terminal block, removable screw type, located at lower side of unit. For **GC35MH-16MT-DSS**, 1 no., 10-pin terminal block, removable screw type, located at left lower side of unit.	

3.1.4 Relay Output Specifications

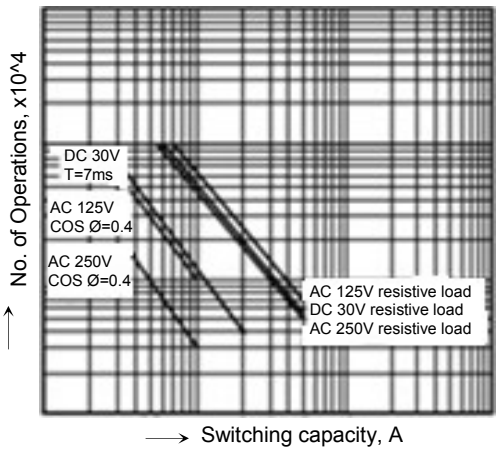
Item	Description	
Number of outputs	16 for GC35MH-32MR-D 8 for GC35MH-16MR-D	
Type of output	Non latching normally open (NO) contact Electro-mechanical relay	
Voltage rating	24 to 250 VAC, 47-63 Hz, 5-30 VDC	
Current rating	500 mA at 250 VAC and 30 VDC per point 1.5 A per common	
Minimum load	1 mA	
Contact life	Electrical life	Refer table and Life curve *
	Mechanical life	min 20, 000, 000 (180 cpm)
Response time	**OFF to ON**	10 ms
	ON to OFF	5 ms
Isolation	Galvanic between output and internal circuit	
Dielectric strength	3 KVrms (between coil and contact circuit) 0.75 KVrms (for open contact)	
I/O terminal blocks	Two 10-pin, screw type removable terminal blocks for GC35MH-32MR-D One 10-pin, screw type removable terminal block for GC35MH-16MR-D	

Electrical life of relay

Voltage	Current	Type of load	Electrical life (20 cpm)
250 VAC	1 A	Resistive	500,000
		Inductive	30,000
	500 mA	Resistive	10,00,000
		Inductive	80,000
30 VDC	1 A	Resistive	600,000
		Inductive	150,000
	500 mA	Resistive	10,00,000
		Inductive	3,80,000

The graph shown is provided by relay manufacturer specification sheet. There may be some degree of variation in relay characteristics depending on ambient and type of load.
So this data should be used only for reference purpose.

*** Life curve of Relay:**



3.2 Wiring

Wiring of Main unit comprises of 24 VDC input supply wiring and I/O wiring.

3.2.1 Wiring of Input Power Supply (24 VDC)

For connecting power supply, 3-pin removable terminal block is provided to connect 24 VDC input supply to the controller Main unit. Connect 24 VDC supply between first 2 terminals. Connect last terminal to clean Earth directly as per the guidelines provided in above section.

Ensure that EARTH cable is thick and short as far as possible to provide low impedance path.

If EARTH is not connected, it may result in electric shock or erroneous operation.

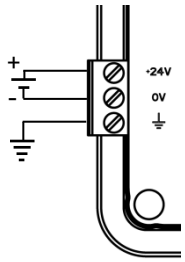


Figure 12: Main unit power supply wiring

Marked as ⏻, Power ON LED glows when proper 24 VDC input power supply is connected and internal 5 VDC supply is generated. It is OFF if internal 5 VDC is not generated or input power 24 VDC not connected or incoming fuse blown off.

Subminiature fuse protection (630 mA) is provided onboard to protect incoming 24 VDC supply. This fuse is soldered on PCB internally and should not be replaced on the field. It is recommended to connect a Miniature Circuit Breaker (MCB) of proper rating in series with mains as additional protection and to serve as a manual isolator.

3.2.2 Wiring Digital Inputs

Main unit provides 16/8 points of 24 VDC digital inputs (sink/source type) as shown in the table below.

Main unit model	No. of inputs	No. of 10-pin terminal blocks for input connection
GC35MH-32MT-DS	16	2
GC35MH-32MT-DSS	16	2
GC35MH-32MR-D	16	2
GC35MH-16MT-DS	8	1
GC35MH-16MT-DSS	8	1
GC35MH-16MR-D	8	1

For Main unit, 8 input points are connected to one 10-pin input terminal block. Input terminal block/s are provided at upper side.

Unit provides; 1 common each for a group of 4 inputs. Any group can be wired for sink or source operation independently.

Refer section *Digital input sink/source operation* to understand sink/ source operation,

The wiring diagram below shows how to connect field input devices like potential free push buttons and limit switches for sink and source operation. The diagram also shows connection of PNP type of switch connected for sink type of operation and NPN type of switch connected for source type of operation.

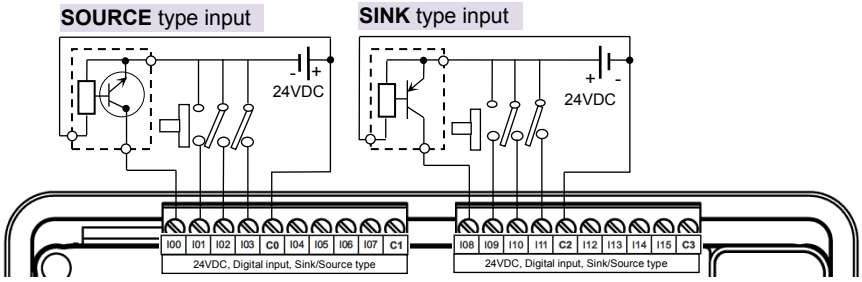


Figure 13: Main unit input connections

Here, input group **I00** to **I03** connected for source type of operation and input group **I08** to **I11** connected for sink type of operation.

GOC Main unit models GC35MH-32MT-DS, GC35MH-32MT-DSS and GC35MH-32MR-D provide 2 input terminal blocks. Models GC35MH-16MT-DS, GC35MH-16MT-DSS and GC35MH-16MR-D provide 1 input terminal block.

Some of the input devices like proximity switches may malfunction due to inherent off state leakage current. Ensure that proper bleeder resistor is connected as a load considering maximum OFF current specified.

3.2.3 Wiring of Transistor Sink Outputs

Main unit provides 16/8 points of 24 VDC transistor output (sink type) as shown in the table below.

Main unit model	No. of transistor outputs (sink)	No. of 10-pin terminal blocks for output connection
GC35MH-32MT-DS	16	2
GC35MH-16MT-DS	8	1

For Main unit, 8 output points are connected to one 10-pin input terminal block. Output terminal block/s are provided at lower side. As transistor output is of sink type, connect one end of output device to output point on terminal block and connect other end of output device to 24 VDC load supply.

It is mandatory to connect 24 VDC output load supply as shown below.

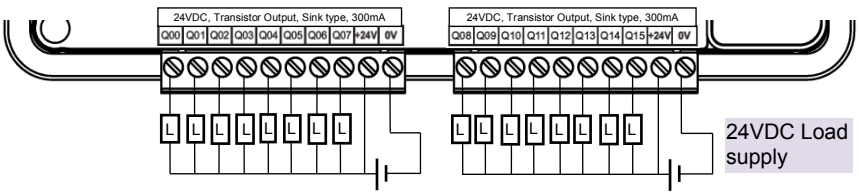


Figure 14: Main unit transistor sink output connections

The ON voltage of the output transistor is approximately 0.6V. When driving a semiconductor element, check the voltage characteristics of the applied element.

GOC Main unit model GC35MH-32MT-DS provides 2 output terminal blocks. Model GC35MH-16MT-DS provides 1 output terminal block.

3.2.4 Wiring of Transistor Source Outputs

Main unit provides 16/8 points of 24 VDC transistor output (source type) as shown in the table below.

Main unit model	No. of transistor outputs (source)	No. of 10-pin terminal blocks for output connection
GC35MH-32MT-DSS	16	2
GC35MH-16MT-DSS	8	1

For Main unit, 8 output points are connected to one 10-pin input terminal block. Output terminal block/s are provided at lower side. As transistor output is of source type, connect one end of output device to output point on terminal block and connect other end of output device to GND terminal of 24 VDC load supply.

It is mandatory to connect 24 VDC output load supply as shown below.

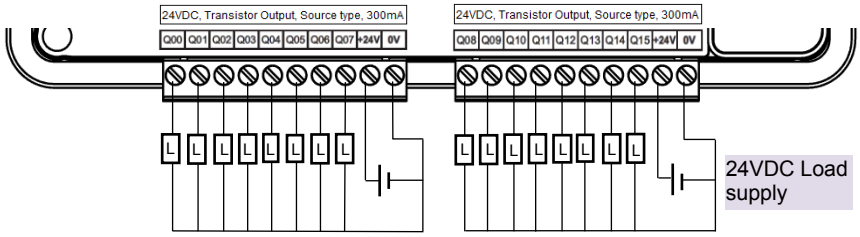


Figure 15: Main unit transistor output connections

The ON voltage across the output transistor is approximately 0.6V maximum. When driving a semiconductor element, check the voltage characteristics of the applied element.

GOC Main unit model GC35MH-32MT-DSS provides 2 output terminal blocks. Model GC35MH-16MT-DSS provides 1 output terminal block.

3.2.5 Wiring of Relay Outputs

Main unit provides 16/8 points of relay outputs as shown in the table below.

Main unit model	No. of relay outputs	No. of 10-pin terminal blocks for outputs connection
GC35MH-32MR-D	16	2
GC35MH-16MR-D	8	1

Before wiring relay outputs, refer section *Guidelines for using relay output*.

For Main unit, 8 outpoint points are connected to one 10-pin input terminal block. Output terminal block/s are provided at lower side. 2 common points are provided per 10-pin output terminal block i.e. 1 common point for 4 outputs.

The figure below shows how to connect output devices to terminal block.

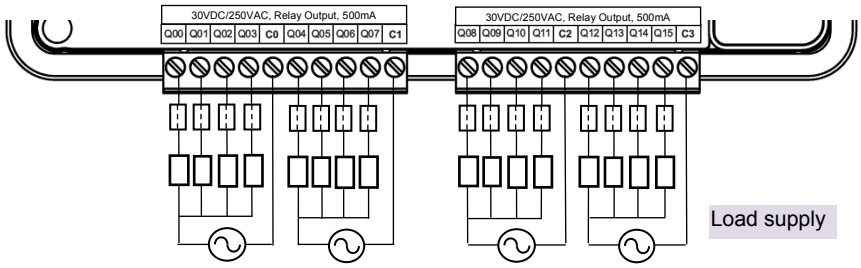


Figure 166: Main unit relay output connections

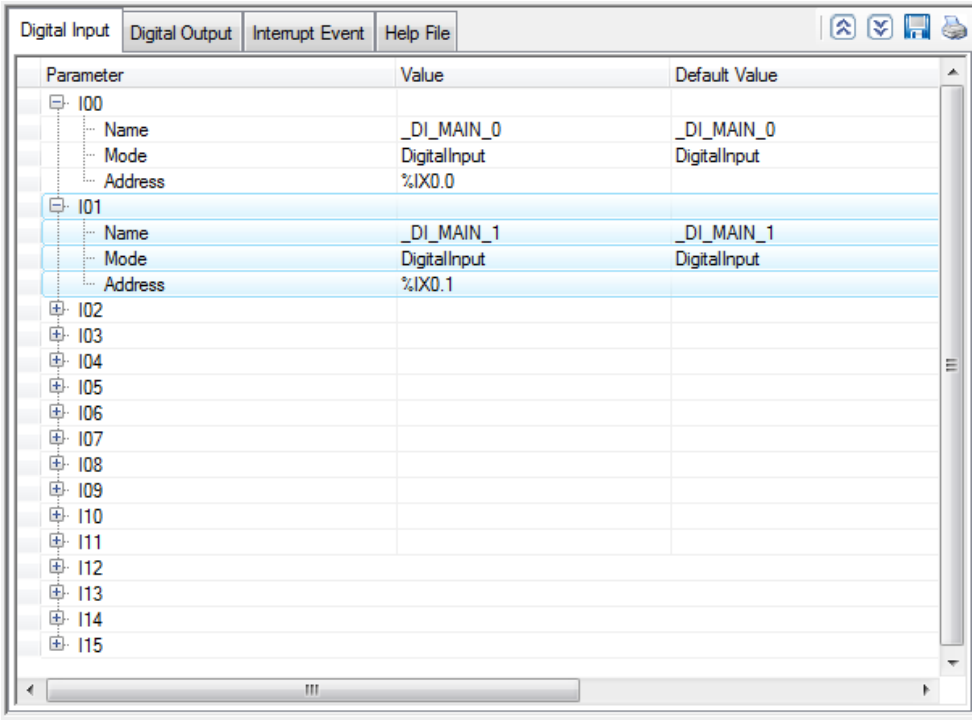
GOC Main unit model GC35MH-32MR-D provides 2 output terminal blocks. Model GC35MH-16MR-D provides 1 output terminal block.

3.3 Configuration and Programming

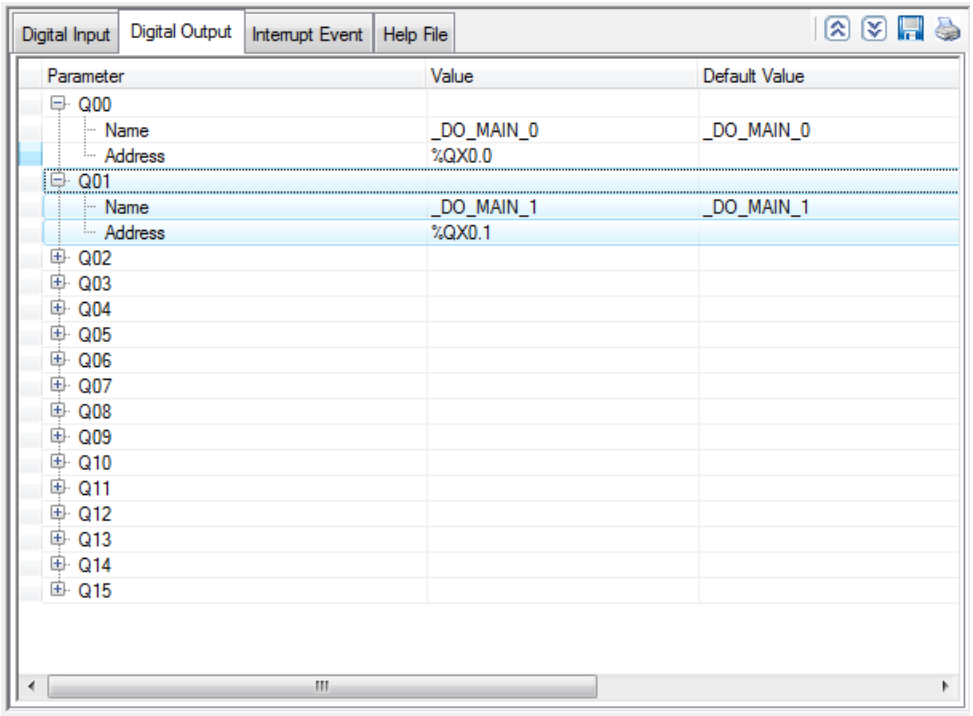
Integrated tool " Hardware Configuration <R> provides input output configuration as well as interrupt event configuration. It also allows configuring special functionality to some of the digital inputs e.g. high speed counter.

3.3.1 Digital I/O Configuration

For GOC, I/O memory map is fixed. Main unit consumes input memory %IB0, %IB1 and output memory %QB0, %QB1. Screen shot below shows Digital Input configuration.



Screen shot below shows Digital Output configuration.

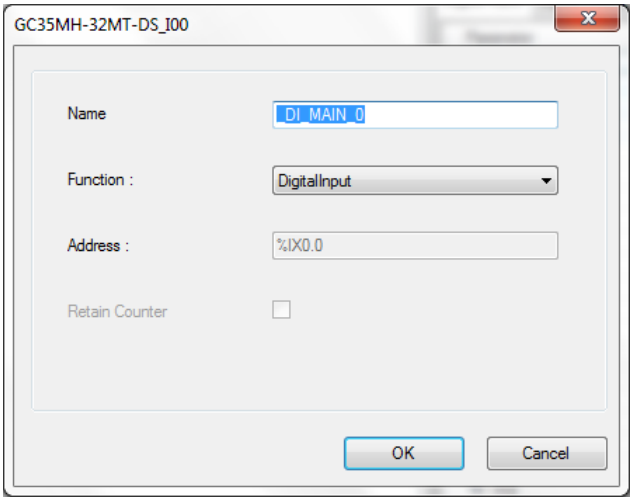


Default configuration provides predefined symbolic naming for each input and output. There is no other configuration required if user requires functionality as normal digital input and output.

For input I00, symbolic name is `_DI_MAIN_0` and address is %IX0.0.

Prefix is `_DI_` and text `MAIN_0` indicates that it is input I00 of Main unit.

User can change the name with fixed prefix as `_DI_` e.g. `_DI_CycleStart` after selecting input and clicking on respective highlight.



Similarly, symbolic names of any input and output can be changed as per application requirement. Based on this configuration, " Hardware Configuration <R> generates a Global Variable List " ImplicitIOList <R> which provides declaration of all the I/O points for configured units as well as predefined variables as per the feature configured depending of type of unit configured. Screen shot below shows I/O list for Main unit model GC35MH-32MT-DS which provides 16 Pt. 24 VDC Digital Input, sink/source + 16 Pt. 24 VDC Transistor Output, Sink type, 300mA.

```
(*                                  MAIN : GC35MH-32MT-DS                                                    *)
(*************************************************************************************************************)
(*--------Digital Input--------*)
    _DI_CycleStart      AT    %IX0.0    :    BOOL;          (* DI 00, Main *)
    _DI_MAIN_1          AT    %IX0.1    :    BOOL;          (* DI 01, Main *)
    _DI_MAIN_2          AT    %IX0.2    :    BOOL;          (* DI 02, Main *)
    _DI_MAIN_3          AT    %IX0.3    :    BOOL;          (* DI 03, Main *)
    _DI_MAIN_4          AT    %IX0.4    :    BOOL;          (* DI 04, Main *)
    _DI_MAIN_5          AT    %IX0.5    :    BOOL;          (* DI 05, Main *)
    _DI_MAIN_6          AT    %IX0.6    :    BOOL;          (* DI 06, Main *)
    _DI_MAIN_7          AT    %IX0.7    :    BOOL;          (* DI 07, Main *)
    _DI_MAIN_8          AT    %IX1.0    :    BOOL;          (* DI 08, Main *)
    _DI_MAIN_9          AT    %IX1.1    :    BOOL;          (* DI 09, Main *)
    _DI_MAIN_10         AT    %IX1.2    :    BOOL;          (* DI 10, Main *)
    _DI_MAIN_11         AT    %IX1.3    :    BOOL;          (* DI 11, Main *)
    _DI_MAIN_12         AT    %IX1.4    :    BOOL;          (* DI 12, Main *)
    _DI_MAIN_13         AT    %IX1.5    :    BOOL;          (* DI 13, Main *)
    _DI_MAIN_14         AT    %IX1.6    :    BOOL;          (* DI 14, Main *)
    _DI_MAIN_15         AT    %IX1.7    :    BOOL;          (* DI 15, Main *)

(*--------Digital Output--------*)
    _DO_SV1             AT    %QX0.0    :    BOOL;          (* DO 00, Main *)
    _DO_MAIN_1          AT    %QX0.1    :    BOOL;          (* DO 01, Main *)
    _DO_MAIN_2          AT    %QX0.2    :    BOOL;          (* DO 02, Main *)
    _DO_MAIN_3          AT    %QX0.3    :    BOOL;          (* DO 03, Main *)
    _DO_MAIN_4          AT    %QX0.4    :    BOOL;          (* DO 04, Main *)
    _DO_MAIN_5          AT    %QX0.5    :    BOOL;          (* DO 05, Main *)
```

The table below provides the details of I/O bits related to GC35H-32MT-DS.

I/O Variables	Address	Description
Digital Inputs		
_DI_MAIN_0	%IX0.0	Holds ON/OFF status of Main unit input I00
_DI_MAIN_1	%IX0.1	Holds ON/OFF status of Main unit input I01
⋮	⋮	⋮
⋮	⋮	⋮
_DI_MAIN_14	%IX1.6	Holds ON/OFF status of Main unit input I14
_DI_MAIN_15	%IX1.7	Holds ON/OFF status of Main unit input I15
Digital Outputs		
_DO_MAIN_0	%QX0.0	Holds ON/OFF status of Main unit output Q00
_DO_MAIN_1	%QX0.1	Holds ON/OFF status of Main unit output Q01
⋮	⋮	⋮
⋮	⋮	⋮
_DO_MAIN_14	%QX1.6	Holds ON/OFF status of Main unit output Q14
_DO_MAIN_15	%QX1.7	Holds ON/OFF status of Main unit output Q15

Tool " Hardware Configuration <R> generates I/O list as per Main unit model selected. It generates list of 16 digital inputs and 16 digital/relay outputs for models GC35H-32MT-DS and GC35H-32MR-D. It generates list of 8 digital inputs and 8 digital/relay outputs for models GC35H-16MT-DS and GC35H-16MR-D.

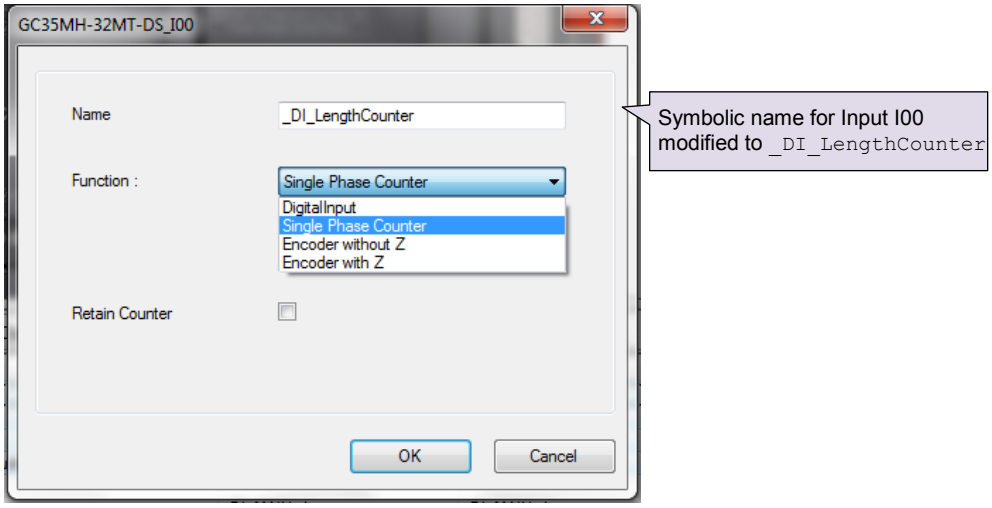
3.3.2 High Speed Input Configuration

Main unit offers special functionality for some of the digital inputs as shown in the table below.

Description \ Inputs	I00	I01	I02	I03	I04	I05
Single phase counter with software direction control	HSC0	-	-	HSC3	-	-
	20 KHz	-	-	20 KHz	-	-
Quadrature encoder A phase, B phase	ENCAB0		-	ENCAB3		-
	10 KHz		-	10 KHz		-
Quadrature encoder A phase, B phase with Z pulse	ENCABZ0			ENCABZ3		
	10 KHz			10 KHz		

Refer *High Speed Counter* for more detail.

User can configure counter functionality for digital input I00 and I03 by selecting respective input from digital input configuration tab and clicking on respective highlight.



Based on this configuration, " Hardware Configuration <R> generates a Global Variable List " ImplicitIOList <R> which provides declaration of predefined global variables. The list below shows declaration of predefined variables for configuration of single phase counter.

```
(*-------- High Speed Counter 0 '_DI_LengthCounter' Parameters--------*)
    _HSC0_En        :    BOOL;     (* Enable Counting HSC0 *)
    _HSC0_Dir       :    BOOL;     (* Counting Direction HSC0 *)
    _HSC0_Reset     :    BOOL;     (* Reset HSC0 count *)
    _HSC0_Load      :    BOOL;     (* Load Preset value HSC0 *)
    _HSC0_PV        :    DINT;     (* Preset value HSC0 *)
    _HSC0_CV        :    DINT;     (* Current value HSC0 *)
```

The list below shows declaration of predefined variables for configuration of encoder at digital inputs I00, I01 without Z marker input.

```
(*--------ENCODER 0 '_DI_LengthCounter' Parameters--------*)
    _ENC0_En                :       BOOL;           (* Enable Counting ENCODER0 *)
    _ENC0_Reset             :       BOOL;           (* Reset ENCODER0 count *)
    _ENC0_Load              :       BOOL;           (* Load Preset value ENCODER0 *)
    _ENC0_PV                :       DINT;           (* Preset value ENCODER0 *)
    _ENC0_CV                :       DINT;           (* Current Value ENCODER0 *)
```

The list below shows declaration of predefined variables for configuration of encoder with Z marker at digital inputs I00, I01, I02.

```
(*--------ENCODER 0 '_DI_LengthCounter' Parameters--------*)
    _ENC0_En                :       BOOL;           (* Enable Counting ENCODER0 *)
    _ENC0_Reset             :       BOOL;           (* Reset ENCODER0 count *)
    _ENC0_Load              :       BOOL;           (* Load Preset value ENCODER0 *)
    _ENC0_PV                :       DINT;           (* Preset value ENCODER0 *)
    _ENC0_Z_En              :       BOOL;           (* Enable Z Signal of ENCODER0 *)
    _ENC0_CV                :       DINT;           (* Current Value ENCODER0 *)
```

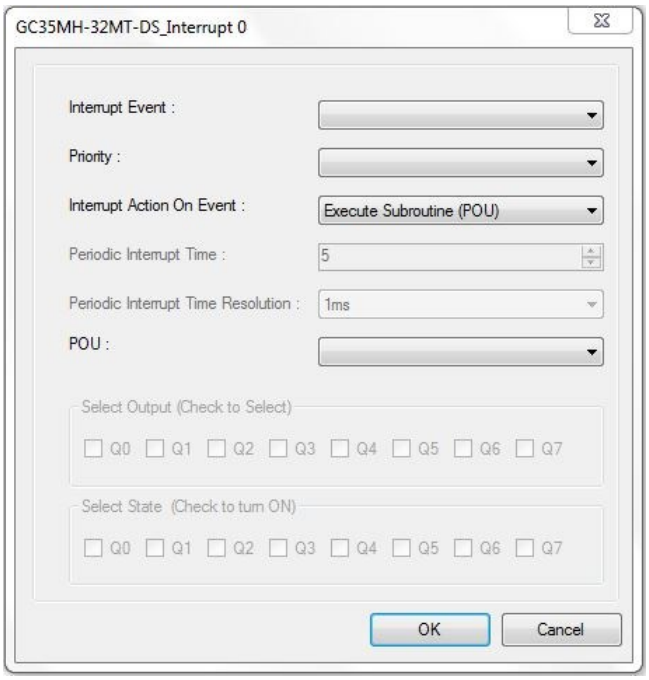
The table below provides the details of predefined global variables related to high speed counter/ encoder interface.

☐=0 for counter configuration for input I00 and encoder configuration at input I00, I01 and I02
☐=3 for counter configuration for input I03 and encoder configuration at input I03, I04 and I05

Single Phase Counter	Encoder Without Z	Encoder With Z	Data Type	Description
_HSC☐_En	_ENC☐_En	_ENC☐_En	BOOL	If TRUE, counter starts counting incoming pulses and updates counter value CV. If FALSE, counter does not count pulses and CV holds last counter value.
_HSC☐_Dir	Not applicable	Not applicable	BOOL	If TRUE, counter counts in upward direction. If FALSE, counter counts in downward direction.
_HSC☐_Reset	_ENC☐_Reset	_ENC☐_Reset	BOOL	If TRUE, CV is reset to 0.
_HSC☐_Load	_ENC☐_Load	_ENC☐_Load	BOOL	If TRUE, counter value gets modified to a value defined by PV and CV shows modified counter value.
_HSC☐_PV	_ENC☐_PV	_ENC☐_PV	DINT	If Load is TRUE, counter value gets modified to a value defined by PV.
Not applicable	Not applicable	_ENC☐_Z_En	BOOL	If FALSE, Z marker pulse is ignored If TRUE, counter is reset to 0 on occurrence of Z marker pulse.
_HSC☐_CV	_ENC☐_CV	_ENC☐_CV	DINT	Returns counter current value if En is TRUE. CV holds last value as long as RUN is FALSE.

3.3.3 Interrupt Configuration

User can configure up to 4 interrupt events. To configure any interrupt event, click on  in Interrupt Event tab. It pops up a dialogue box as shown below.



The table below explains various options available for different parameters.

Parameter	Options	Description
Interrupt Event	**1.** EVENT_PERIODIC	Interrupt event occurs periodically after specified time period.
	2. EVENT_IX2_R_TRIG	Interrupt event occurs on rising and/or falling edge of digital input I02
	3. EVENT_IX2_F_TRIG	
	4. EVENT_IX5_R_TRIG	Interrupt event occurs on rising and/or falling edge of digital input I05
	5. EVENT_IX5_F_TRIG	
	6. EVENT_HSC0_CMP	Interrupt event occurs when high speed counter value reaches target value.
	7. EVENT_HSC3_CMP	
	8. EVENT_WARM	Interrupt event occurs on particular CPU initialization
	9. EVENT_COLD	
	10.EVENT_BEFORE_INPUTSCAN	Interrupt event occurs at particular time slot in PLC scan.
	11.EVENT_AFTER_INPUTSCAN	
	12.EVENT_AFTER_LOGICSCAN	
	13.EVENT_BEFORE_OUTPUTSCAN	
	14.EVENT_AFTER_OUTPUTSCAN	
Priority	0 to 15	Select priority of execution in case of occurrence of multiple interrupt events. Priority 0 has higher precedence. Priority should be unique for each event.

Interrupt event configuration…

Parameter	Options	Description
Interrupt Action On Event	Execute Subroutine (POU)	Decide action to be performed on occurrence of interrupt event. User can select predefined PROGRAM type POU to be executed on occurrence of interrupt event.
	Set Digital Output	Decide action to be performed on occurrence of interrupt event. User can select switching On/Off output/s Q00 to Q07 on Main unit. User can select output/s and desired state as On/Off. It is recommended to use this option for transistor outputs for better performance.
Periodic Interrupt Time	0.5/ 1 to 1250/ 2500 ms	User can set interval of periodic interrupt from 1 to 2500 ms for occurrence of periodic interrupt event if resolution selected is 1 ms. User can set interval of periodic interrupt from 0.5 to 1250 ms for occurrence of periodic interrupt event if resolution selected is 0.5 ms.
Periodic Interrupt Time Resolution	1 ms or 0.5 ms	Select resolution for time interval for periodic interrupt event. Minimum and maximum interval for occurrence of periodic interrupt event depends upon this factor.
POU	Select already programmed PROGRAM type POU from the list	This option is available if 'Execute Subroutine (POU)' is selected for 'Interrupt Action On Event' It is necessary to program PROGRAM type POU before configuring interrupt event.
Select Output	Checkout particular output to select	This option is available if 'Set Digital Output' is selected for 'Interrupt Action On Event'. Select output/s from Q00 to Q07 from Main unit which is to be switched to selected State (On/Off) on occurrence of selected interrupt event. It is recommended to use this option for transistor outputs for better performance.
Select State	Checkout to turn On selected output.	

Based on this configuration, " Hardware Configuration <R> generates a Global Variable List " ImplicitIOList <R> which provides declaration of predefined global variables. The list below shows declaration of predefined variables for configuration of periodic interrupt event.

```
(*--------Interrupt Parameters--------*)
   _ClearInterruptQueue          :    BOOL;          (* Clears queue of pending interrupt events *)
   _HoldAllInterrupts            :    BOOL;          (* Holds processing of all attached interrupt events  *)

   _EVENT_PERIODIC_Attach        :    BOOL;          (* Attach Interrupt Event*)
```

The table below provides the details of global variables related to interrupt event configuration.

□□□□□□= Name of the interrupt event configured,

Global Variables	Data Type	Description
_ClearInterruptQueue	BOOL	Clears queue of pending interrupt events.
_HoldAllInterrupts	BOOL	Holds processing of all attached interrupt events.
□□□□□□_Attach	BOOL	If TRUE, enables specified interrupt and configured interrupt action will be executed on occurrence of event. If FALSE, ignores interrupt event even if it is configured.
□□□□□□_ATCH_Done	BOOL	TRUE : Interrupt attached successfully FALSE : Interrupt not attached
_HSC#_SetCompareValue	BOOL	Applicable only for interrupt event EVENT_HSC#_CMP. If TRUE, respective high speed counter compare value is set to value defined by _HSC#_CompareValue.
_HSC#_CompareValue	DINT	Applicable only for interrupt event EVENT_HSC#_CMP. Holds value for high speed counter comparison and this value is set when boolean variable _HSC#_SetCompareValue is TRUE.

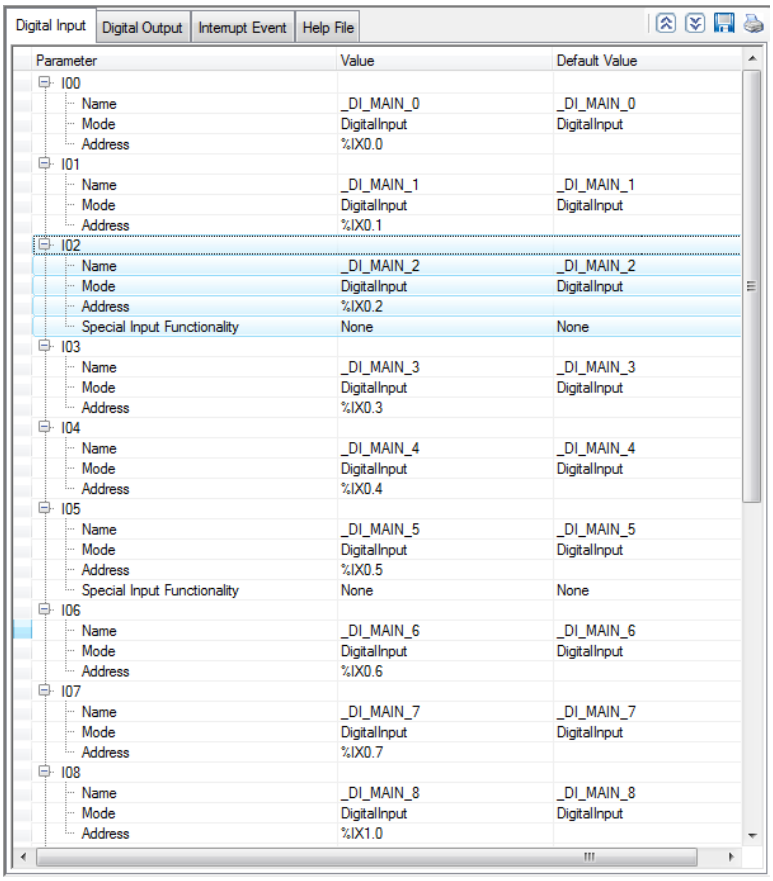
3.3.4 Special Input Functionality

User can configure special functionality of pulse catch or frequency measurement or pulse ON time measurement to inputs I02 and I05 on Main unit. The subsequent sections explain configuration of each functionality is detail

3.3.4.1 Pulse Catch Configuration

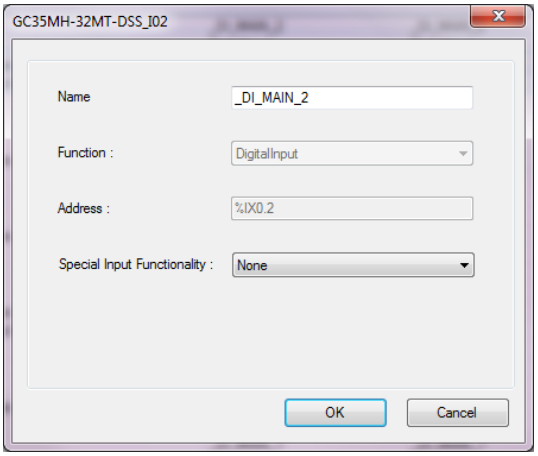
The pulse catch feature allows user to capture change in input state that last for short duration and which may not be captured when the PLC reads the digital inputs in regular input scan.

To configure pulse catch, select respective input from 'Digital Input' configuration tab. Configuration dialogue is displayed as shown below.

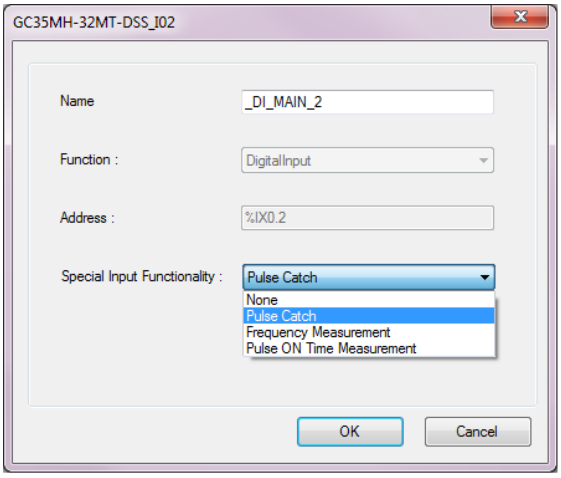


User can configure Pulse Catch functionality for digital input I02 and I05 by selecting respective input from "Digital Input" configuration tab and clicking on respective highlight.

Dialogue “GC35MH-32MT-DSS_I02” pops up showing holds special input functionality as “None”.

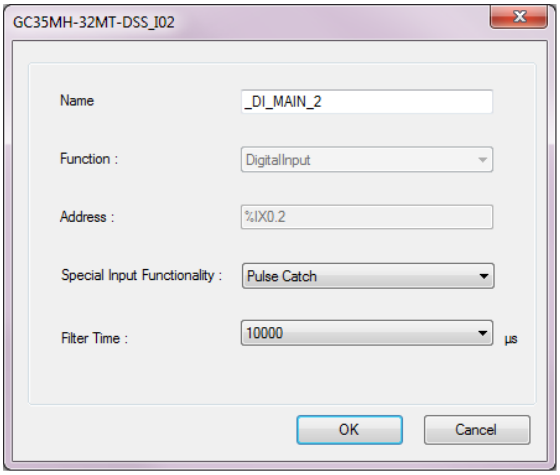


User can select special input functionality as “Pulse Catch” and can set parameter “Filter Time” in Hz with default value as 10000 µs using drop-down list.



Filter time defines minimum width of pulse to be captured. Pulses with duration less than filter time are not captured.

Filter time options available for selection are 200 µs, 400 µs, 800 µs, 1000 µs, 2000 µs, 5000 µs and 10000 µs. Default selection is 10000 µs.



Based on this configuration, " Hardware Configuration <R> generates a Global Variable List " ImplicitIOList <R> which provides declaration of predefined global variables.

The list below shows declaration of predefined variables for configuration of pulse catch functionality at digital input I02.

```
(*-------- Pulse catch at DI 02 '_DI_MAIN_2' Parameters--------*)
    _PulseCatch2_En                 :       BOOL;                   (*Enable Pulse Catch, DI 02, Main *)
```

The table below provides the details of global variables related to pulse catch functionality.

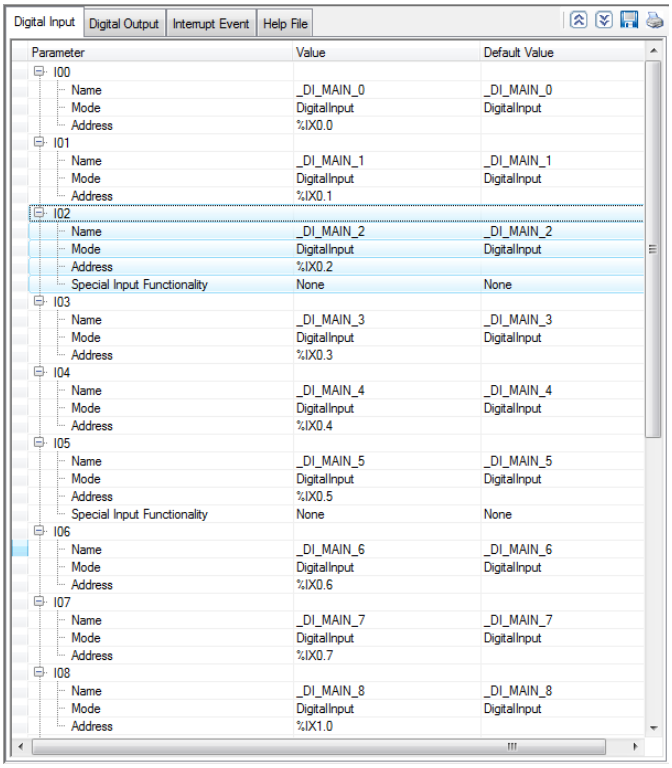
☐=2 for pulse catch configuration for input I02,
☐=5 for pulse catch configuration for input I05

Global Variable	Data Type	Description
_PulseCatch☐_En	BOOL	If **TRUE**, enables pulse catch function for digital input. If **FALSE**, input acts as normal digital input with 10 ms filter time.

3.3.4.2 Frequency Measurement Input Configuration

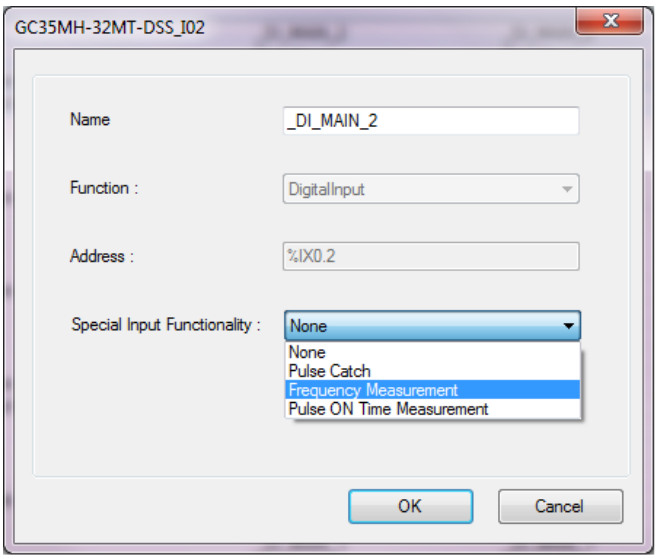
This feature measures frequency of input pulses at inputs I02 and I05 on Main unit.

Select respective input from 'Digital Input' configuration tab to configure special input functionality as shown below.

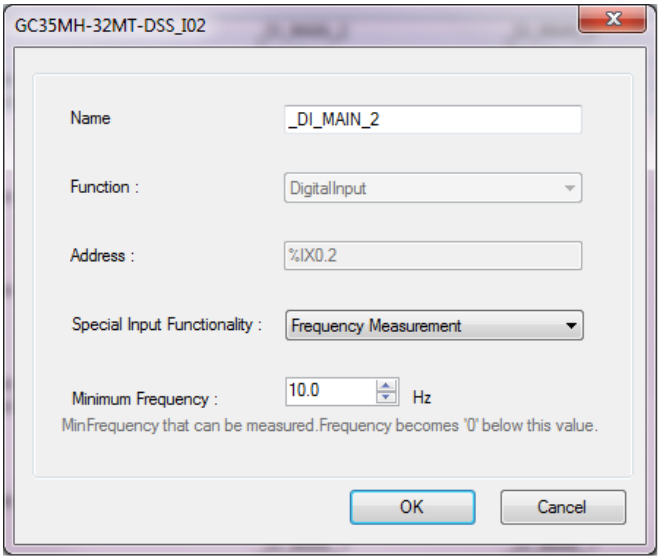


Select respective input from "Digital Input" configuration tab and click on respective highlight.

Dialogue “GC35MH-32MT-DSS_I02” pops up with special input functionality as “None”.



User can select “Frequency Measurement” and then set parameter “Minimum Frequency” in Hz from 0.1 to 50 Hz. Default value as 10.0 Hz.



After saving and closing application of “Hardware configuration tool”, based on configuration,
Hardware Configuration <R>generates Global Variable List ImplicitIOList <R> which provides declaration of predefined global variables. The list below shows declaration of predefined variables for configuration of frequency measurement.

```
(*-------- Frequency Measurement at DI 02 '_DI_MAIN_2' --------*)
    _Frequency2                         :       REAL;       (*Measured Frequency, DI 02, Main *)
```

The table below provides the details of global variables related to frequency measurement functionality configuration.

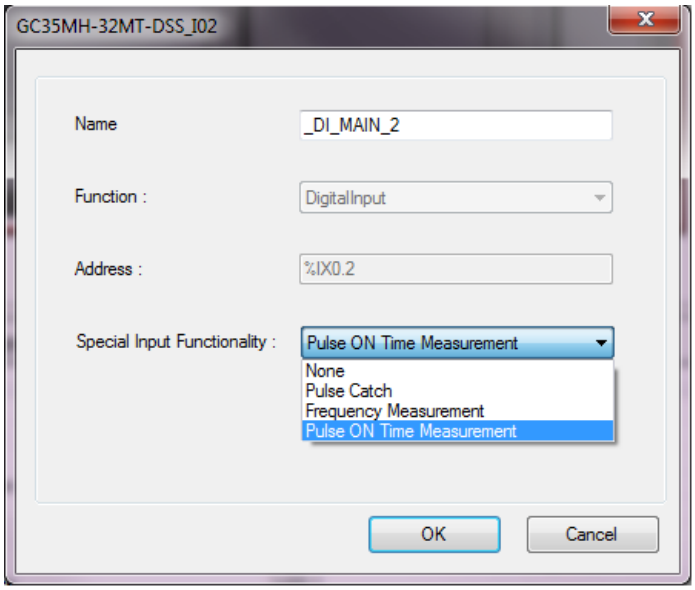
☐=2 for frequency measurement configuration for input I02,
☐=5 for frequency measurement configuration for input I05

Global Variables	Data Type	Description
_Frequency☐	REAL	This variable holds actual frequency of input pulses.

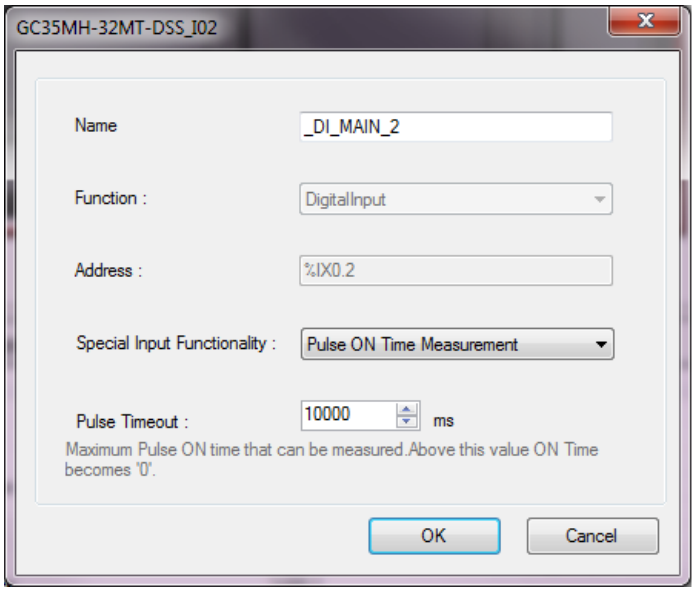
3.3.4.3 Pulse ON Time Measurement Configuration

User can configure Pulse ON Time measurement functionality for digital input **I02** and **I05** by selecting respective input from “Digital Input” configuration tab and clicking on respective highlight.

In dialogue “GC35MH-32MT-DSS_I02”, select “Pulse ON Time Measurement” from drop-down list.



User can then set parameter “Pulse Timeout” in ms from 10 to 10000 ms. Default value as 10000 ms.



After saving and closing application of “Hardware configuration tool”, based on configuration,

Hardware Configuration <R> generates a Global Variable List ImplicitIOList <R> which provides declaration of predefined global variables. The list below shows declaration of predefined variables for configuration of frequency measurement.

```
(*-------- Pulse ON Time Measurement at DI 02 '_DI_MAIN_2' --------*)
   _PulseONTime2                :        DWORD;                (*Measured Pulse ON Time, DI 02, Main *)
```

☐=2 for Pulse ON Time measurement configuration for input I02,
☐=5 for Pulse ON Time measurement configuration for input I05

Global Variables	Data Type	Description
_PulseONTime☐	DWORD	This variable holds the latest value of time duration with resolution 1 µsec

Note: Section explains configuration and variable declaration for input **I02**. User can configure input **I05** in similar manner with dialogue “GC35MH-32MT-DSS_I05”.

4 I/O Extension Unit GC-8EX-ES

This is 8 Pt. 24 VDC Digital Input extension unit. User can attach up to 2 I/O extension units on the back side of Main unit.

4.1 Specifications

Item	Description
Number of Inputs	8, 2 groups of 4 inputs each
Mode of operation	Sink or source in group of 4 inputs; depends on connections
Voltage rating	24 VDC (18 to 30 V including ripple)
ON / OFF voltage	**ON voltage:** 18 VDC minimum **OFF voltage:** 5 VDC maximum
Current rating	6 mA at 24 VDC
OFF Current	3.8 mA max
Input impedance	5.2 KΩ
Transition delay	10 ms (Digital filter)
Isolation	Optical 1.5 KV between input and internal bus, 1.5 KV between groups, Nil between input points in a group.
Method of termination	2 nos. 8-pin terminal blocks, fixed, screw type
Status indication	On LCD screen on Main unit
Dimensions (in mm)	61.5 (W) x 75 (H) x 24.5 (D)
Weight (in grams)	60

4.2 Wiring

I/O extension unit provides two 8-pin fixed terminal blocks for wiring I/O devices. One is located at lower side of unit and another is located at upper side of unit.

I/O extension unit provides 2 commons; 1 common each for a group of 4 inputs. Either group can be wired for sink or source operation independently. For an example, the wiring diagram shows input group I00 to I03 connected for sink type of operation and input group I04 to I07 connected for source type of operation.

Refer section *Digital input sink/source operation* to understand sink/ source operation.

Refer section *Wiring*, before wiring digital inputs to I/O extension unit.

The wiring diagram shows how to connect field input devices like potential free push buttons and limit switches for sink and source operation. The diagram also shows connection of typical proximity switch. PNP switch is connected for source type of operation and NPN switch is connected for sink type of operation.

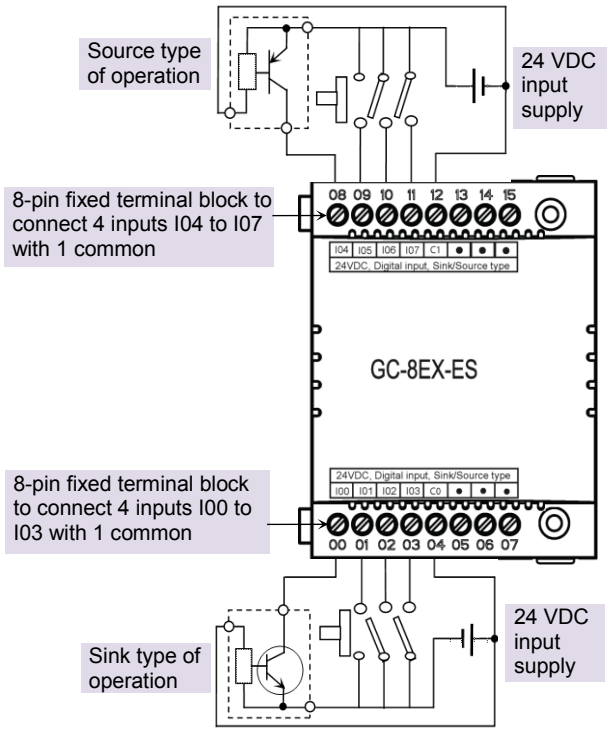


Figure 17: Wiring GC-8EX-ES

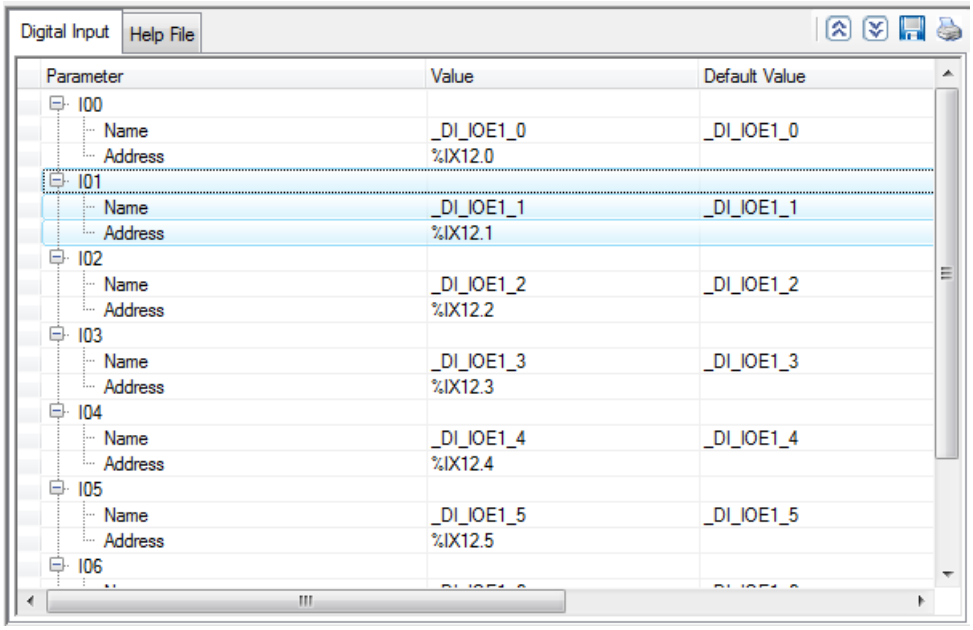
> Some of the input devices like proximity switches may malfunction due to inherent off state leakage current. Ensure that proper bleeder resistor is connected as a load considering maximum OFF current specified.

4.3 Configuration and Programming

For GOC I/O Extension units, I/O memory map is fixed.

GC-8EX-ES consumes %IB12 when fixed in IO1 slot and %IB14 when fixed in IO2 slot.

Integrated tool " Hardware Configuration <R> provides Digital Input configuration as shown below.

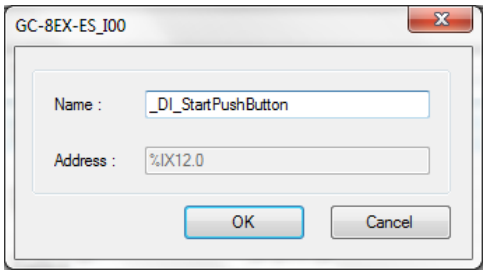


Default configuration provides predefined symbolic naming for each input. There is no other configuration required.

For input I00, symbolic name is `_DI_IOE1_0` and address is %IX12.0.

Prefix is `_DI_`. Text `IOE1_0` indicates that unit is fixed in IO1 slot and input is I00.

User can change the name with fixed prefix as `_DI_` e.g. `_DI_StartPushButton` after selecting input and clicking on respective highlight.



Based on this configuration, " Hardware Configuration <R> generates a Global Variable List " ImplicitIOList <R> which provides declaration of all the I/O points for configured units as well as predefined variables as per the feature configured depending of type of unit configured.

Screen shot below shows Input list for GC-8EX-ES unit fixed in IO1 slot.

```
(*************************************************************************************************************************)
    _IOE1UnitState                          :       BYTE;
(*--------Digital Input--------*)
    _DI_StartPushButton         AT      %IX12.0 :       BOOL;           (* DI 00, IOE1 *)
    _DI_IOE1_1          AT      %IX12.1 :       BOOL;           (* DI 01, IOE1 *)
    _DI_IOE1_2          AT      %IX12.2 :       BOOL;           (* DI 02, IOE1 *)
    _DI_IOE1_3          AT      %IX12.3 :       BOOL;           (* DI 03, IOE1 *)
    _DI_IOE1_4          AT      %IX12.4 :       BOOL;           (* DI 04, IOE1 *)
    _DI_IOE1_5          AT      %IX12.5 :       BOOL;           (* DI 05, IOE1 *)
    _DI_IOE1_6          AT      %IX12.6 :       BOOL;           (* DI 06, IOE1 *)
    _DI_IOE1_7          AT      %IX12.7 :       BOOL;           (* DI 07, IOE1 *)
```

The table below provides the details of global variables related to GC-8EX-ES.

☐=1 for unit fixed in IO1 slot, ☐=2 for unit fixed in IO2 slot,

Global Variables	Data Type	Description
_IOE☐UnitState	BYTE	Holds current state IO Extension unit as **0**: No unit is fixed in IO slot or No unit is detected in IO slot by CPU **1**: Mismatch between configured unit and attached unit in IO slot. **2**: Configured unit is detected and it is in configuration state **100**: Configured unit is detected, configured successfully and is in running condition.

The table below provides the details of I/O bits related to GC-8EX-ES.

☐=1 for unit fixed in IO1 slot, ☐=2 for unit fixed in IO2 slot,

I/O Variables	Address		Description
	IO1 Slot	IO2 Slot	
_DI_IOE☐_0	%IX12.0	%IX14.0	Holds ON/OFF status of extension unit input I00
_DI_IOE☐_1	%IX12.1	%IX14.1	Holds ON/OFF status of extension unit input I01
_DI_IOE☐_2	%IX12.2	%IX14.2	Holds ON/OFF status of extension unit input I02
_DI_IOE☐_3	%IX12.3	%IX14.3	Holds ON/OFF status of extension unit input I03
_DI_IOE☐_4	%IX12.4	%IX14.4	Holds ON/OFF status of extension unit input I04
_DI_IOE☐_5	%IX12.5	%IX14.5	Holds ON/OFF status of extension unit input I05
_DI_IOE☐_6	%IX12.6	%IX14.6	Holds ON/OFF status of extension unit input I06
_DI_IOE☐_7	%IX12.7	%IX14.7	Holds ON/OFF status of extension unit input I07

For GC-8EX-ES, input byte %IB13 and %IB15 as well as output bytes %QB12 to %QB15 are not used.

5 I/O Extension Unit GC-6EYR-ES

This is 6 Pt. Relay output extension unit. User can attach up to 2 I/O extension units on the back side of Main unit.

5.1 Specifications

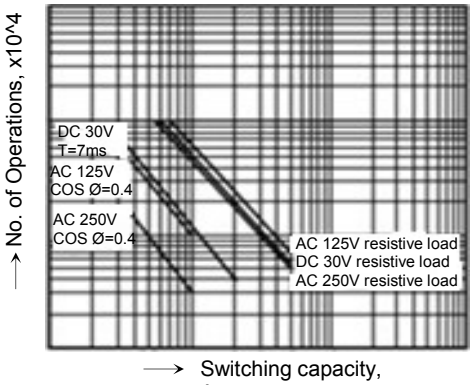
Item	Description	
Number of Outputs	6, 2 groups of 3 outputs each	
Type of output	Non latching normally open (NO) contact Electro-mechanical relay	
Voltage rating	24 to 250 VAC, 47-63 Hz, 5-30 VDC	
Current rating	500 mA at 250 VAC and 30 VDC per point 1.5 A per common	
Minimum load	1 mA	
Contact life	Electrical life	Refer table and Life curve*
	Mechanical life	min 20, 000, 000 (180 cpm)
Response time	OFF to ON	10 ms
	ON to OFF	5 ms
Isolation	Galvanic between output and internal circuit	
Dielectric strength	3 KVrms (between coil and contact circuit) 0.75 KVrms (for open contact)	
Method of termination	2 nos. 8-pin terminal blocks, fixed, screw type	
Status indication	On LCD screen on Main unit	
Dimensions (in mm)	61.5 (W) x 75 (H) x 24.5 (D)	
Weight (in grams)	60	

Electrical life of relay

Voltage	Current	Type of load	Electrical life (20 cpm)
250 VAC	1 A	Resistive	500,000
		Inductive	30,000
	500 mA	Resistive	10,00,000
		Inductive	80,000
30 VDC	1 A	Resistive	600,000
		Inductive	150,000
	500 mA	Resistive	10,00,000
		Inductive	3,80,000

The graph shown is provided by relay manufacturer specification sheet. There may be some degree of variation in relay characteristics depending on ambient and type of load.
So this data should be used only for reference purpose.

***Life curve of Relay:**



5.2 Wiring

I/O extension unit provides two 8-pin fixed terminal blocks for wiring I/O devices. One is located at lower side of unit and another is located at upper side of unit.

Refer section *Guidelines for using relay output*, before wiring relay outputs,

Refer section *Wiring*, before wiring output devices to I/O extension unit.

I/O extension unit provides 2 commons; 1 common each for a group of 3 relay outputs. Unit requires external 24 VDC supply for relay coil operation. The wiring diagram shows how to connect field output devices to the unit.

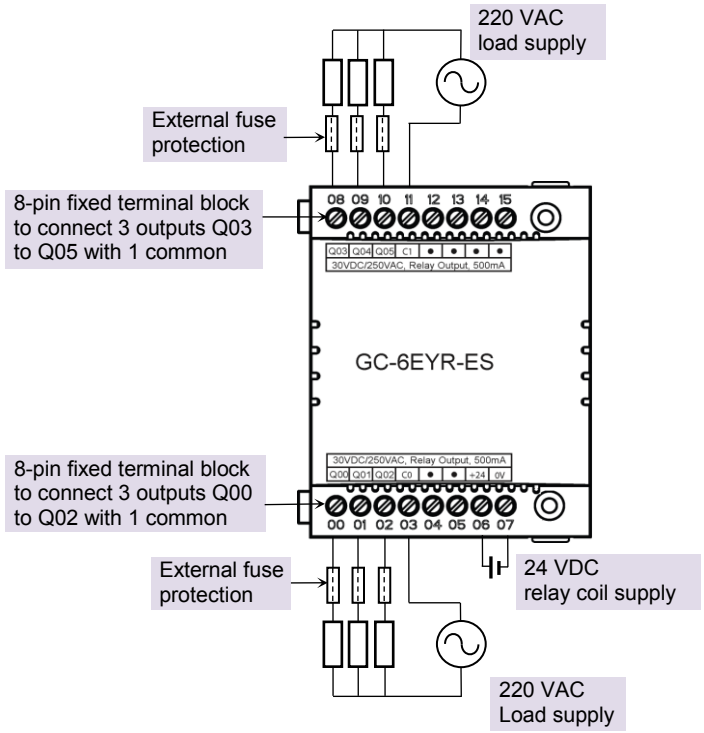


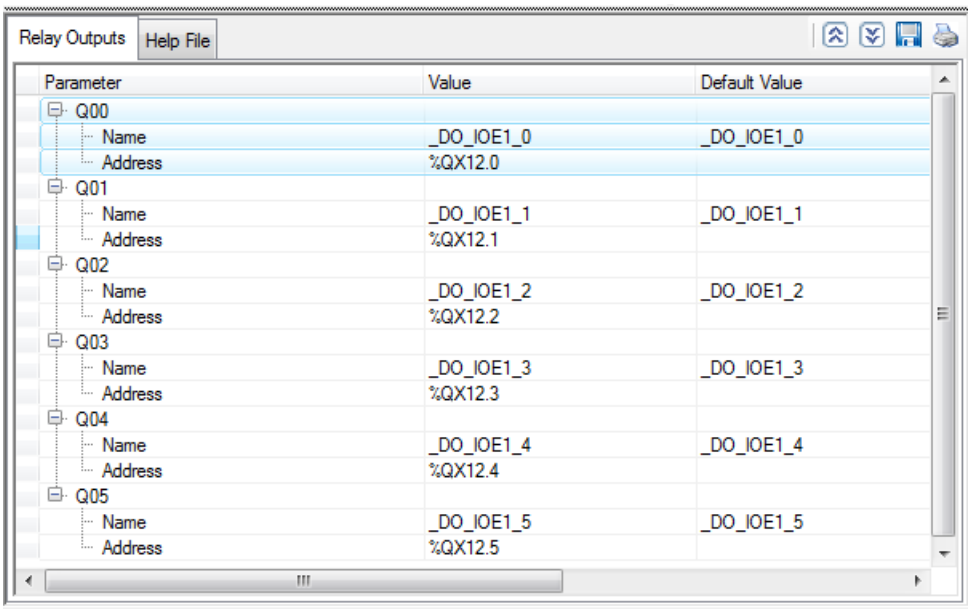
Figure 18: Wiring GC-6EYR-ES

5.3 Configuration and Programming

For GOC I/O Extension units, I/O memory map is fixed.

GC-6EYR-ES consumes %QB12 when fixed in IO1 slot and %QB14 when fixed in IO2 slot.

Integrated tool " Hardware Configuration <R> provides Relay Output configuration as shown below.

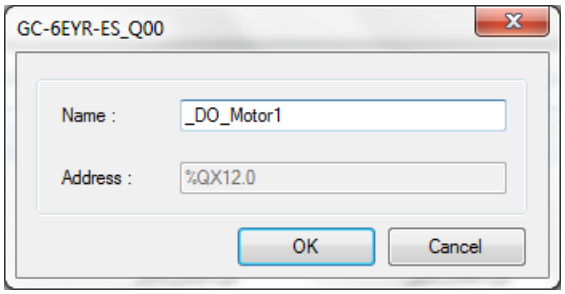


Default configuration provides predefined symbolic naming for each output. There is no other configuration required.

For output Q00, symbolic name is `_DO_IOE1_0` and address is %QX12.0.

Prefix is `_DO_`. Text `IOE1_0` indicates that unit is fixed in IO1 slot and output is Q00.

User can change the name with fixed prefix as `_DO_` e.g. `_DO_MOTOR1`.after selecting output and clicking on respective highlight.



Based on this configuration, " Hardware Configuration <R> generates a Global Variable List " ImplicitIOList <R> which provides declaration of all the I/O points for configured units as well as predefined variables as per the feature configured depending of type of unit configured.

Screen shot below shows output list for GC-6EYR-ES unit fixed in IO1 slot.

```
(*                              IOE1 : GC-6EYR-ES                                    *)
(*****************************************************************************************)
  _IOE1UnitState                       :     BYTE;

  _IOE1No24V        AT     %IX12.7     :     BOOL;
(*--------Relay Output--------*)
  _DO_Motor1        AT     %QX12.0     :     BOOL;          (* RO 00, IOE1 *)
  _DO_IOE1_1        AT     %QX12.1     :     BOOL;          (* RO 01, IOE1 *)
  _DO_IOE1_2        AT     %QX12.2     :     BOOL;          (* RO 02, IOE1 *)
  _DO_IOE1_3        AT     %QX12.3     :     BOOL;          (* RO 03, IOE1 *)
  _DO_IOE1_4        AT     %QX12.4     :     BOOL;          (* RO 04, IOE1 *)
  _DO_IOE1_5        AT     %QX12.5     :     BOOL;          (* RO 05, IOE1 *)
```

The table below provides the details of global variables related to GC-6EYR-ES.

☐=1 for unit fixed in IO1 slot, ☐=2 for unit fixed in IO2 slot.

Global Variables	Data Type	Description
`_IOE☐UnitState`	BYTE	Holds current state IO Extension unit as **0**: No unit is fixed in IO slot or No unit is detected in IO slot by CPU **1**: Mismatch between configured unit and attached unit in IO slot. **2**: Configured unit is detected and it is in configuration state **100**: Configured unit is detected, configured successfully and is in running condition.

The table below provides the details of I/O bits related to GC-6EYR-ES.

I/O Variables	Address		Description
	IO1 Slot	IO2 Slot	
`_IOE1No24V`	%IX12.7	%IX12.7	Becomes TRUE if 24 VDC relay coil supply to extension unit is absent.
`_DO_IOE☐_0`	%QX12.0	%QX14.0	Holds ON/OFF status of extension unit output Q00
`_DO_IOE☐_1`	%QX12.1	%QIX14.1	Holds ON/OFF status of extension unit output Q01
`_DO_IOE☐_2`	%QX12.2	%QX14.2	Holds ON/OFF status of extension unit output Q02
`_DO_IOE☐_3`	%QX12.3	%QX14.3	Holds ON/OFF status of extension unit output Q03
`_DO_IOE☐_4`	%QX12.4	%QX14.4	Holds ON/OFF status of extension unit output Q04
`_DO_IOE☐_5`	%QX12.5	%QX14.5	Holds ON/OFF status of extension unit output Q05
	%QX12.6	%QX14.6	Not available
	%QX12.7	%QX14.7	Not available

For GC-6EYR-ES, input byte %QB13 and %QB15 as well as input bytes %IB13 to %IB15 are not used

6 I/O Extension Unit GC-8ET-ESS

This I/O extension unit (GC-8ET-ESS) provides 4 point 24 VDC digital inputs and 4 point 24VDC transistor outputs. It allows sink or source type connections for 4 inputs and source type of connections for 4 outputs. It can be fixed in any IO slot on the back side of Main unit.

6.1 Specifications

Item	Description	
Digital Inputs (Sink/ Source type)		
Number of Inputs	04	
Mode of operation	Sink or source, depends on connections	
Voltage rating	24 VDC (18 to 30 V including ripple)	
ON / OFF voltage	**ON voltage:** 18 VDC minimum **OFF voltage:** 5 VDC maximum	
Current rating	6 mA at 24 VDC	
OFF current	3.8 mA maximum	
Input impedance	5.2 KΩ	
Transition delay	10ms (Default filter time)	
Digital Outputs (Source type)		
Number of Outputs	04	
Type of output	Transistor source type	
Voltage rating	24 VDC (18 to 30 V including ripple)	
Current rating	1.5 A per output, 1 common point. Paralleling of outputs is possible	
ON voltage drop	0.6V DC maximum	
OFF state leakage current	10 µA maximum	
Response Time	**OFF to ON:** 300 µs	**ON to OFF:** 300 µs
Isolation	1.5 KV optical from internal bus	
Detection	No 24 VDC supply	
Load supply	24 VDC (18 to 30 V including ripple),	
	Reverse polarity protection	

GC-8ET-ESS specifications continue…

General	
I/O status indication	On LCD screen on Main unit
Isolation	Optical 1.5 KV between input and internal circuit Optical 1.5 KV between output and internal circuit
Protection	Output Short circuit protection. Fast demagnetization for inductive loads
Method of termination	For inputs, 1 no., 8-pin terminal block (fixed, screw type) located at upper side of unit
	For outputs, 1 no., 8-pin terminal block (fixed, screw type) located at lower side of unit.
Dimensions (in mm	61.5 (W) x 75 (H) x 24.5 (D)
Weight (in grams)	60

6.2 Wiring

I/O extension unit provides two 8-pin fixed terminal blocks for wiring I/O devices. One is located at lower side of unit is for transistor (source type) outputs and another is located at upper side of unit is for digital (sink/ source type) inputs.

As 1 common is provided for a group of 4 inputs, all the inputs can be either connected for source type of operation or sink type of operation at a time as shown in figure below. Figure also shows connection of transistor outputs Q00 to Q03 as source type of outputs.

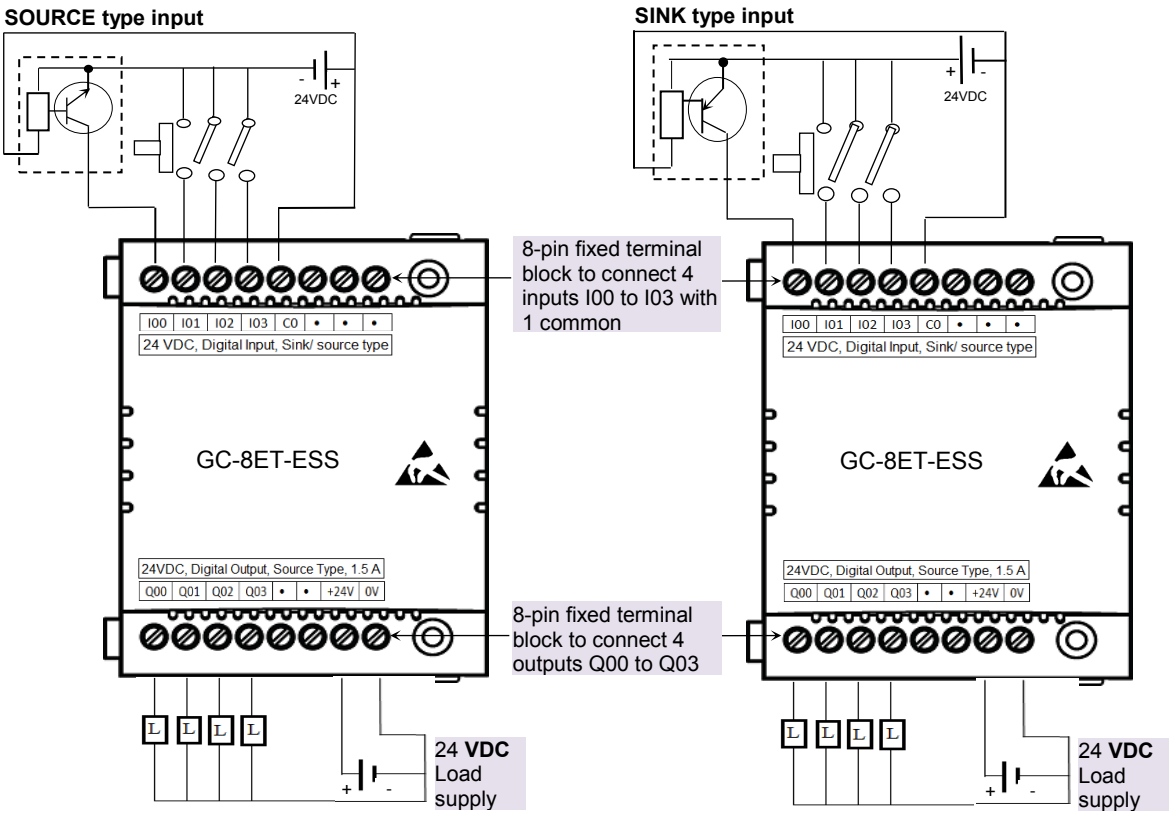
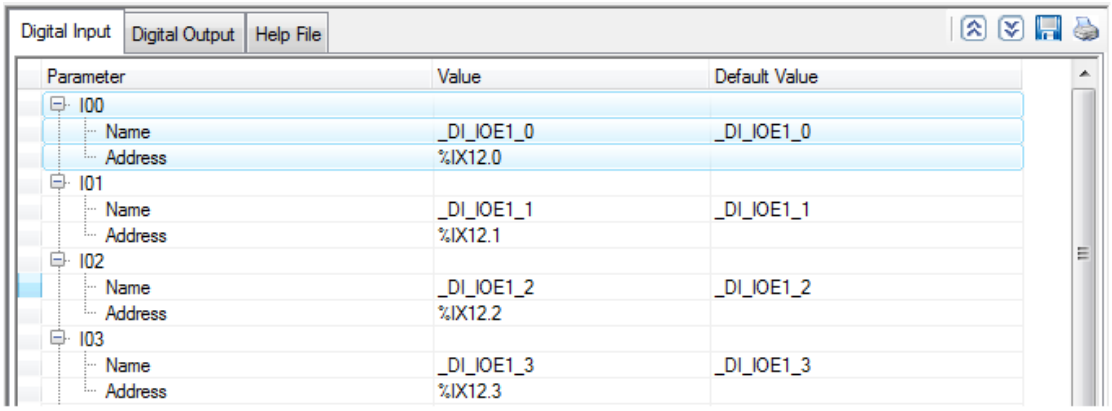
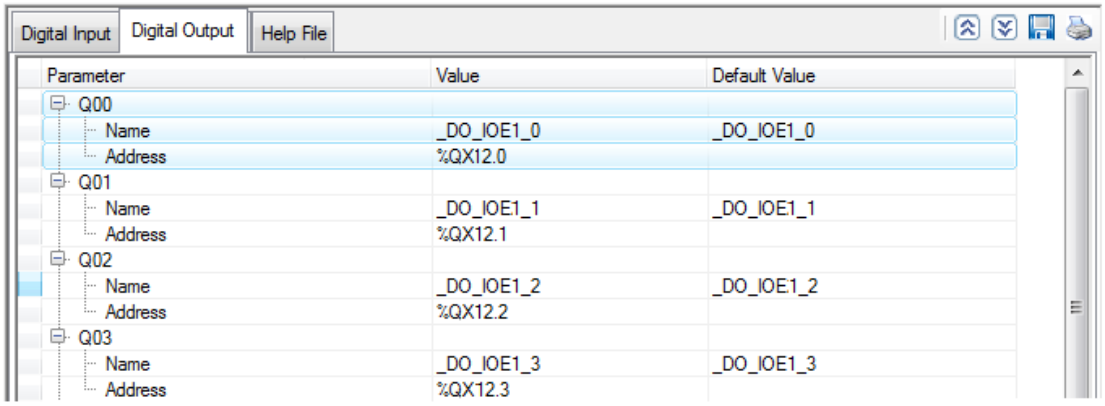


Figure 19: Wiring GC-8ET-ESS

6.3 Configuration and Programming

For GOC I/O Extension units, I/O memory map is fixed. GC-8ET-ESS consumes %IB12 and %QB12 when fixed in IO1 slot.

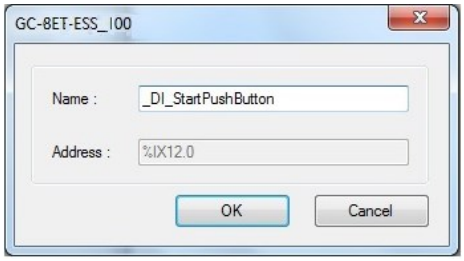




***Note**:* Similarly, GC-8ET-ESS consumes %IB14 and %QB14 when fixed in IO2 slot.

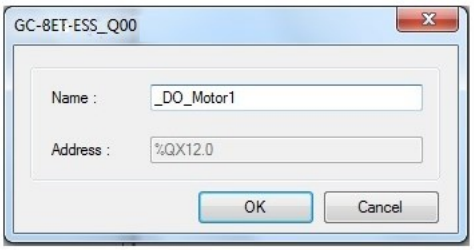
Default configuration provides predefined symbolic naming of each input and output. There is no other configuration required.
For input I00, name is _DI_IOE1_0 and address is %IX12.0.
Prefix is _DI_. Text IOE1_0 indicates that unit is fixed in slot1 and input is I00. Similarly, for output, name is _DO_Text IOE1_0 and address is %QX12.0.
Following screen shot shows I/O list for GC-8ET-ESS unit fixed in slot1.

Based on this configuration, in integrated tool Hardware Configuration <R> generates Global Variable List, ImplicitIOList <R> which provides declaration of all inputs for configured unit.




```
(*--------Digital Input--------*)
    _DI_StartPushButton AT      %IX12.0 :       BOOL;           (* DI 00, IOE1 *)
```

Similarly, based on configuration, in integrated ImplicitIOList <R> tool generates Global Variable List, Hardware Configuration <R> which provides declaration of all outputs for configured unit.




```
(*--------Digital Output---------*)
_DO_Motor1          AT      %QX12.0 :       BOOL;           (* DO 00, IOE1 *)
```

After adding, GC-8ET-ESS I/O extension unit using Hardware Configuration Tool, click on CoDeSys tab “Resources”→ ImplicitIOList <R> , then by default, following variable list can be viewed.

```
(*****************************************************************************************************************)
(*                                  IOE1 : GC-8ET-ESS
(*****************************************************************************************************************)
    _IOE1UnitState                      :       BYTE;

    _IOE1No24V          AT      %IX12.7 :       BOOL;
(*--------Digital Input--------*)
    _DI_IOE1_0          AT      %IX12.0 :       BOOL;           (* DI 00, IOE1 *)
    _DI_IOE1_1          AT      %IX12.1 :       BOOL;           (* DI 01, IOE1 *)
    _DI_IOE1_2          AT      %IX12.2 :       BOOL;           (* DI 02, IOE1 *)
    _DI_IOE1_3          AT      %IX12.3 :       BOOL;           (* DI 03, IOE1 *)
(*--------Digital Output--------*)
    _DO_IOE1_0          AT      %QX12.0 :       BOOL;           (* DO 00, IOE1 *)
    _DO_IOE1_1          AT      %QX12.1 :       BOOL;           (* DO 01, IOE1 *)
    _DO_IOE1_2          AT      %QX12.2 :       BOOL;           (* DO 02, IOE1 *)
    _DO_IOE1_3          AT      %QX12.3 :       BOOL;           (* DO 03, IOE1 *)
```

The table below provides the details of global variables related to GC-8ET-ESS.

☐=1 for unit fixed in IO1 slot, ☐=2 for unit fixed in IO2 slot.

Global Variables	Data Type	Description
_IOE☐UnitState	BYTE	Holds current state IO Extension unit as **0**: No unit is fixed in IO slot or No unit is detected in IO slot by CPU **1**: Mismatch between configured unit and attached unit in IO slot. **2**: Configured unit is detected and it is in configuration state **100**: Configured unit is detected, configured successfully and is in running condition.

The table below provides the details of I/O bits related to GC-8ET-ESS.

☐=1 for unit fixed in IO1 slot, ☐=2 for unit fixed in IO2 slot,

Input Variables	Address		Description
	IO1 Slot	IO2 Slot	
_DI_IOE☐_0	%IX12.0	%IX14.0	Holds ON/OFF status of extension unit input I00
_DI_IOE☐_1	%IX12.1	%IX14.1	Holds ON/OFF status of extension unit input I01
_DI_IOE☐_2	%IX12.2	%IX14.2	Holds ON/OFF status of extension unit input I02
_DI_IOE☐_3	%IX12.3	%IX14.3	Holds ON/OFF status of extension unit input I03
_IOE☐No24V	%IX12.7	%IX14.7	Becomes TRUE if 24 VDC supply to extension unit is absent.
Output Variables	**Address**		**Description**
	IO1 Slot	IO2 Slot	
_DO_IOE☐_0	%QX12.0	%QX14.0	Holds ON/OFF status of extension unit output Q00
_DO_IOE☐_1	%QX12.1	%QIX14.1	Holds ON/OFF status of extension unit output Q01
_DO_IOE☐_2	%QX12.2	%QX14.2	Holds ON/OFF status of extension unit output Q02
_DO_IOE☐_3	%QX12.3	%QX14.3	Holds ON/OFF status of extension unit output Q03

7 I/O Extension Unit GC-4AD-12

This is 4 Channel analog voltage/current input extension unit that provides 12-bit resolution. User can attach up to 2 I/O extension units on the back side of Main unit.

7.1 Specifications

Item		Description			
Number of inputs		4 channels voltage/current, differential, non-isolated, 12-bit resolution			
Input types (individual channel is software configurable)		**Voltage**		**Current**	
		0 to 10 VDC	-10 to 10 VDC	0 to 20 mA	4 to 20 mA
Output data		0 to 4000	-2000 to 2000	0 to 4000	0 to 4000
1-bit resolution		3 mV	6 mV	6 µA	6 µA
Overall accuracy (% of full scale) *1	At 25°C	±0.4	±0.4	±0.5	±0.5
	At 55°C	±0.6	±0.6	±0.7	±0.7
Input impedance		1 MΩ		250 Ω	
Unit updation time (Depends on no. of avg. samples or digital filter time constant but in synchronization with PLC scan)		Channel data updation = Controller scan time x Number of averaging samples. Channel data updation = Controller scan time + (Digital Filter Time Constant*5) minimum			
Absolute maximum input		±30 VDC/ 30 mA			
Isolation		No isolation			
Method of termination		2 nos. 8-pin terminal blocks, fixed, screw type			
Status indication		On LCD screen			
Dimensions (in mm)		61.5 (W) x 75 (H) x 24.5 (D)			
Weight (in grams)		60			

*1 Accuracy is measured with default filter time of 50 msec

7.2 Wiring

I/O extension unit provides 3 terminals per channel **Vin**, **Iin** and **C**.

Voltage input is connected between terminals **Vin** and **C**.

Current input is connected between terminals **Iin** and **C** with **Vin** and **Iin** connected together externally.

Refer section *Wiring*, before wiring analog input sensors to I/O extension unit.

- For current input, make sure to connect terminals **Vin** and **Iin** together.
- It is recommended to use 2-core shielded twisted pair cable for carrying analog signal.
- Connect cable shield at extension unit end directly to a good quality earth in the control panel. It is recommended to keep cable shield at sensor end unconnected.
- The Earthing resistance should be 100 Ω or less.
- For unused channels, connect **Vin** and **C** terminals together.

The wiring diagram shows how to connect field input devices to unit. For an example, channel 0 and 1 are connected to voltage input and channel 2 and 3 are connected to current input.

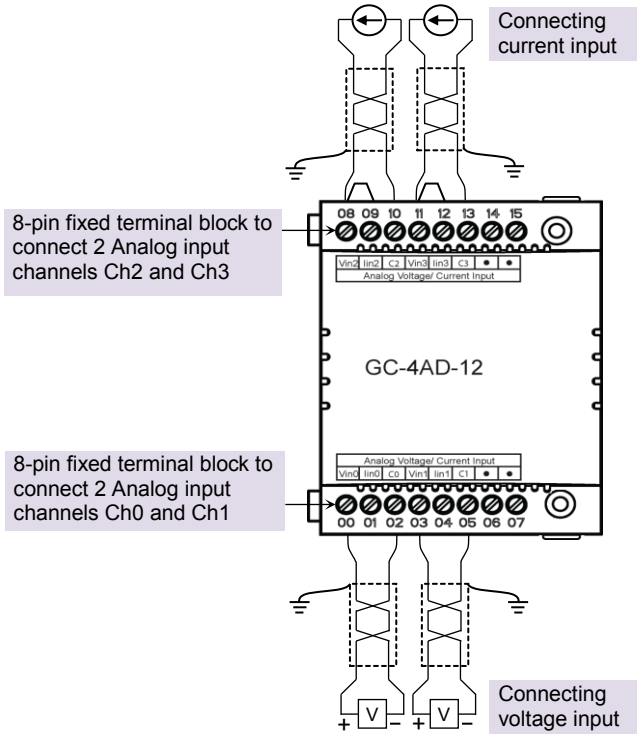


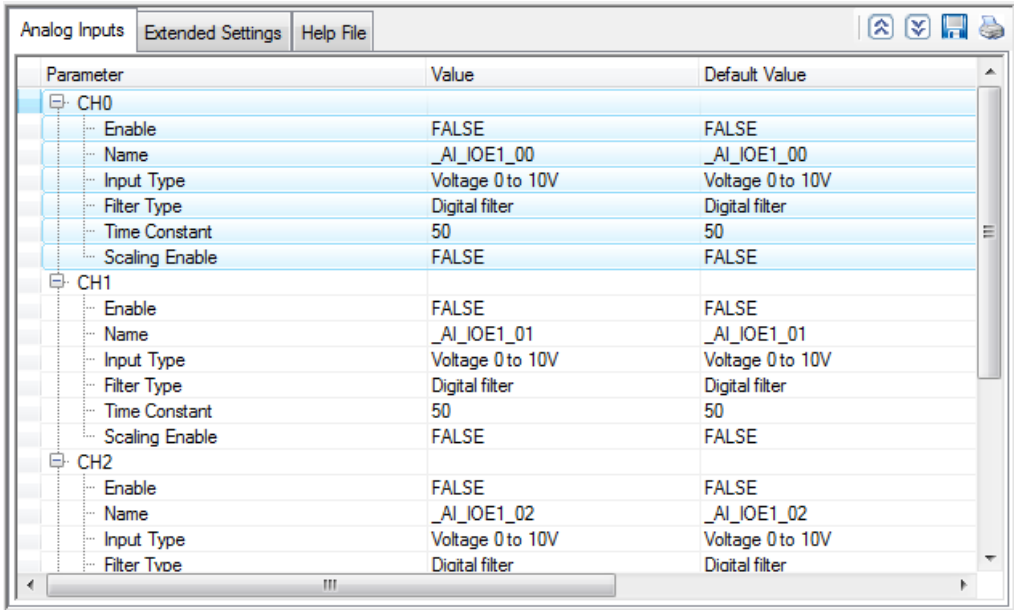
Figure 20: Wiring GC-4AD-12

7.3 Configuration and Programming

For GOC I/O Extension units, I/O memory map is fixed.

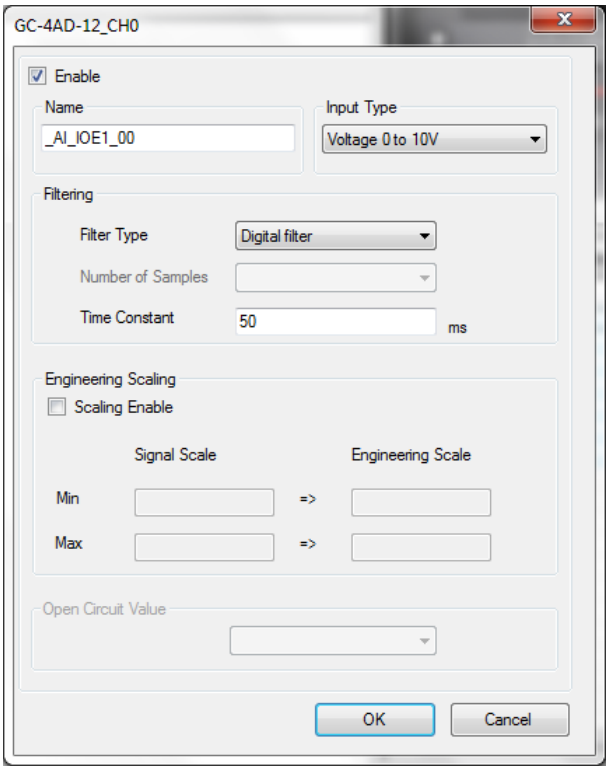
GC-4AD-12 consumes %IB12 and %QB12 when fixed in IO1 slot. It consumes %IB14 and %QB14 when fixed in IO2 slot.

Integrated tool " Hardware Configuration <R> provides Analog Inputs configuration as shown below.

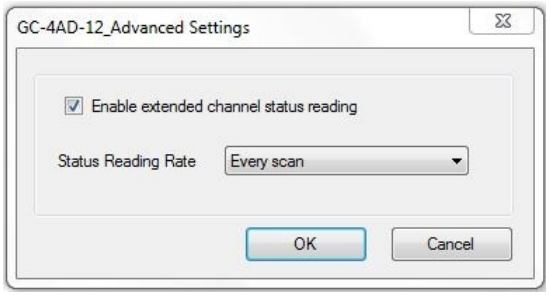


It provides default configuration and user can change it as per application requirement by selecting channel and clicking on respective highlight.

The dialogue below shows Analog Input configuration for individual channel.



The dialogue below shows Extended Settings for a unit.



The table below provides the details of configuration required for analog input channel.

Parameters	Options	Description
Analog Input configuration		
Enable	Checkbox checked	Analog input channel is enabled.
	Checkbox unchecked	Analog input channel is disabled. *1
Name	`_AI_IOE☐_00` to `_AI_IOE☐_03` (☐= 1 for unit fixed in IO1 slot 2- for unit fixed in IO2 slot)	The variable with user defined symbolic name holds analog input channel data. These global variables `_AI_IOE☐_00` to `_AI_IOE☐_03` are declared in Global Variable List ImplicitIOList <R> if respective channel is enabled. User can change the name with fixed prefix as `_AI_` e.g. `_AI_TemeperatureZone1`.

*1 For unused channel, ensure that channel inputs **Vin** and **C** are connected together.

Configuration required for analog input channel…

Parameter	Options	Description
Input Type	Voltage 0 to 10V(Default) Voltage -10V to 10V Current 0 to 20mA Current 4 to 20mA	Selection for type of analog input as per application requirement.
Filter type	No Filter Averaging Digital Filter (Default)	Selection for signal conditioning by software.
Number of Samples	4, 8, 16, 32	Selection for number of samples for moving average. Channel data updation = Controller scan time x Number of averaging samples. Apply averaging for slowly varying analog input signal.
Time Constant	Default value 50 ms.	Enter digital filer time constant value from 10 to 5000 ms. Channel data updation = Controller scan time + (Time Constant*5) Apply digital filter for analog input signal with erroneous fast variations.
Enable Scaling	Checkbox unchecked	Channel data (available in global variables `_AI_IOE☐_00` to `_AI_IOE☐_03)` holds value as per basic resolution. e.g. 0 to 4000 if input signal is 0 to 10 VDC for input type 0 to 10VDC.
	Checkbox checked	Channel data (available in global variables `_AI_IOE☐_00` to `_AI_IOE☐_03)` scaling to engineering units as per user defined values of Min and Max
Engineering Scaling Min	Default value is 0	Enter channel data value required for - 0V input for input type 0 to 10VDC - -10V input for input type -10 to 10VDC - 0mA input for input type 0 to 10VDC - 4mA input for input type 0 to 10VDC
Engineering Scaling Min	Default value is 100	Enter channel data value required for - 10V input for input type 0 to 10VDC, -10 to 10VDC - 20mA input for input type 0 to 20mA, 4 to 20mA
Open Circuit Value	0 (Default) Minimum Value Maximum Value Last Value	Applicable for input type 4 to 20mA. Defines value of channel data in case if open circuit is detected at input channel. Minimum value and maximum depends upon Engineering scaling selected.

Configuration required for analog input channel…

Parameter	Options	Description
Extended Settings		
Enable extended channel status reading	Checked : Extended channel status reading enabled Unchecked : Extended channel status reading disabled	If checked, enables status reading for all channels. Channel status read from unit is available in global variable array `_IOE☐ChannelStatus`.
Status reading rate	Every scan 100ms 1000ms	Defines channel status reading interval.

Based on this configuration, " Hardware Configuration <R> generates a global variable list " ImplicitIOList <R> which provides declaration of all the I/O points for configured units as well as predefined variables as per the feature configured depending of type of unit configured.

Screen shot below shows default variable list for GC-4AD-12 unit fixed in IO1 slot when all the 4 channels are enabled.

```
(*                                  IOE1 : GC-4AD-12                                                                    *)
(*************************************************************************************************************************)
    _IOE1UnitState                          :       BYTE;
(*--------Analog Input--------*)
    _AI_IOE1_00                             :       REAL;           (* AI 00, IOE1 *)
    _AI_IOE1_01                             :       REAL;           (* AI 01, IOE1 *)
    _AI_IOE1_02                             :       REAL;           (* AI 02, IOE1 *)
    _AI_IOE1_03                             :       REAL;           (* AI 03, IOE1 *)

    _IOE1CH0OK          AT      %IX12.0     :       BOOL;
    _IOE1CH1OK          AT      %IX12.1     :       BOOL;
    _IOE1CH2OK          AT      %IX12.2     :       BOOL;
    _IOE1CH3OK          AT      %IX12.3     :       BOOL;
    _IOE1ADCFault       AT      %IX12.6     :       BOOL;

    _IOE1CH0Enable      AT      %QX12.0     :       BOOL;
    _IOE1CH1Enable      AT      %QX12.1     :       BOOL;
    _IOE1CH2Enable      AT      %QX12.2     :       BOOL;
    _IOE1CH3Enable      AT      %QX12.3     :       BOOL;

    _IOE1ChannelStatus                      :       ARRAY [0..3] OF WORD;   (*Channel Status, IOE1 *)

(*************************************************************************************************************************)
```

The table below provides the details of global variables related to GC-4AD-12.

☐=1 for unit fixed in IO1 slot, ☐=2 for unit fixed in IO2 slot,

Global Variables	Data Type	Description
`_IOE☐UnitState`	BYTE	Holds current state IO Extension unit as **0**: No unit is fixed in IO slot or No unit is detected in IO slot by CPU **1**: Mismatch between configured unit and attached unit in IO slot. **2**: Configured unit is detected and it is in configuration state **100**: Configured unit is detected, configured successfully and is in running condition.

Global variables related to GC-4AD-12…

<table>
<tr><th>Global Variables</th><th>Data Type</th><th colspan="2">Description</th></tr>
<tr><td rowspan="7">_AI_IOE☐_00</td><td rowspan="7">REAL</td><td colspan="2">Holds analog input channel 0 data.
The table below provides channel data available when Engineering Scaling is not enabled.</td></tr>
<tr><td>Input Type</td><td>Channel Data</td></tr>
<tr><td>0 to 10 VDC</td><td>0 to 4000</td></tr>
<tr><td>-10 to 10 VDC</td><td>-2000 to 2000</td></tr>
<tr><td>0 to 20 mA</td><td>0 to 4000</td></tr>
<tr><td>4 to 20 mA</td><td>0 to 4000</td></tr>
<tr><td colspan="2">In case if Engineering Scaling is enabled, channel data holds value as per Min and Max values defined for Engineering Scaling</td></tr>
<tr><td>_AI_IOE☐_01</td><td>REAL</td><td colspan="2">Holds analog input channel1 data.as explained for _AI_IOE☐_00</td></tr>
<tr><td>_AI_IOE☐_02</td><td>REAL</td><td colspan="2">Holds analog input channel2 data.as explained for _AI_IOE☐_00</td></tr>
<tr><td>_AI_IOE☐_03</td><td>REAL</td><td colspan="2">Holds analog input channel3 data.as explained for _AI_IOE☐_00</td></tr>
<tr><td rowspan="6">_IOE☐ChannelStatus</td><td rowspan="6">ARRAY [0..3] OF WORD</td><td colspan="2">Holds status of channels if Extended Settings is enabled.
Each array element is assigned for individual channel e.g._IOE☐ChannelStatus[0] holds status of channel 0.
Details of bits of status word as follows</td></tr>
<tr><td>Bit No</td><td>Details</td></tr>
<tr><td>0</td><td>Channel enable status
0 - Disabled
1 - Enabled</td></tr>
<tr><td>1</td><td>Channel configuration
0 - Valid
1 - Invalid</td></tr>
<tr><td>2</td><td>Open circuit application for 4 to 20mA input type
0 - No open circuit
1 - Open circuit</td></tr>
<tr><td>3 - 15</td><td>Reserved</td></tr>
</table>

The table below provides the details of I/O bits related to GC-4AD-12.

☐=1 for unit fixed in IO1 slot, ☐=2 for unit fixed in IO2 slot,

I/O Variables	Address		Description
	IO1 Slot	IO2 Slot	
_IOE☐CH0OK	%IX12.0	%IX14.0	**TRUE**: - Respective channel is enabled and healthy. **FALSE**: - Respective channel is disabled. - Enabled channel has invalid configuration. - Open circuit detected for input type 4 to 20mA
_IOE☐CH1OK	%IX12.1	%IX14.1	
_IOE☐CH2OK	%IX12.2	%IX14.2	
_IOE☐CH3OK	%IX12.3	%IX14.3	
_IOE☐ADCFault	%IX12.6	%IX14.6	ADC fault is detected on unit.
_IOE☐CH0Enable	%QX12.0	%QX14.0	**TRUE** - Enable respective channel through *Hardware Configuration Tool*. **FALSE:** - Disable respective channel through *Hardware Configuration Tool*.
_IOE☐CH1Enable	%QX12.1	%QX14.1	
_IOE☐CH2Enable	%QX12.2	%QX14.2	
_IOE☐CH3Enable	%QX12.3	%QX14.3	

8 I/O Extension Unit GC-4DA-12

This is 4 Channel analog voltage/current output extension unit that provides 12-bit resolution. User can attach up to 2 I/O extension units on the back side of Main unit.

8.1 Specifications

Item		Description			
Number of outputs		4 channels voltage/current, non-isolated, 12-bit resolution			
Output types (individual channel is software configurable)		Voltage		Current	
		0 to 10 VDC	-10 to +10 VDC	0 to 20 mA	4 to 20 mA
Input data		0 to 4000	-2000 to 2000	0 to 4000	0 to 4000
1-bit resolution		5 mV	2.5 mV	5 µA	5 µA
Overall accuracy (% of full scale)	At 25°C	±0.3	±0.3	±0.3	±0.3
	At 55°C	±0.4	±0.4	±0.4	±0.4
Load		> 5 KΩ		0 to 500 Ω	
Unit updation time		In sync with output scan			
Output settling time		2 ms			
Isolation		No isolation			
Output protection		Short circuit protection for voltage output			
Unit supply		24 VDC (18 to 30 VDC including ripple)			
Method of termination		2 nos. 8-pin terminal blocks, fixed, screw type			
Status indication		On LCD screen			
Dimensions (in mm)		61.5 (W) x 75 (H) x 24.5 (D)			
Weight (in grams)		60			

8.2 Wiring

I/O extension unit provides 3 terminals per channel **Vo**, **Io** and **C**. Voltage output is generated between terminals **Vo** and **C**. Whereas current output is generated between terminals **Io** and **C**.

Refer section *Wiring*, before wiring analog output devices to I/O extension unit.

- It is recommended to use 2-core shielded twisted pair cable for carrying analog signal.
- Connect cable shield at extension unit end directly to a good quality earth in the control panel. It is recommended to keep cable shield at sensor end unconnected.
- The Earthing resistance should be 100 Ω or less.

The wiring diagram shows how to connect field output devices to extension unit. For an example, channel 0 and 1 are configured for voltage output and channel 2 and 3 are configured for current output.

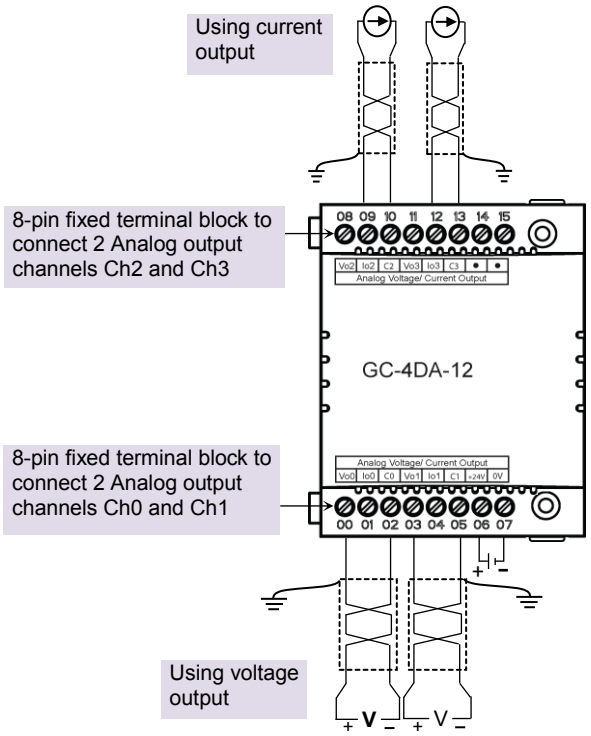


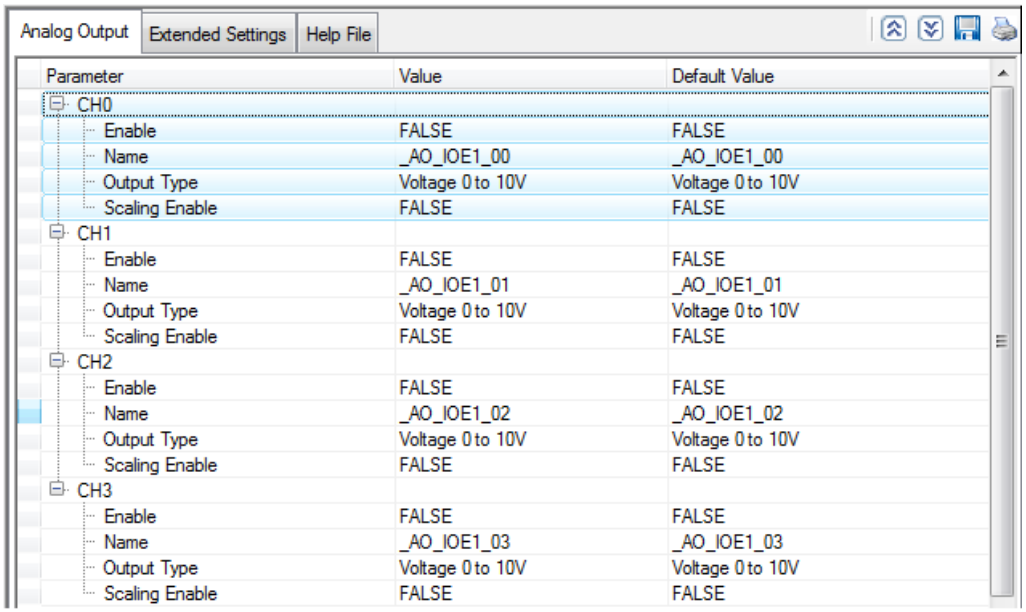
Figure 21: Wiring GC-4DA-12

8.3 Configuration and Programming

For GOC I/O Extension units, I/O memory map is fixed.

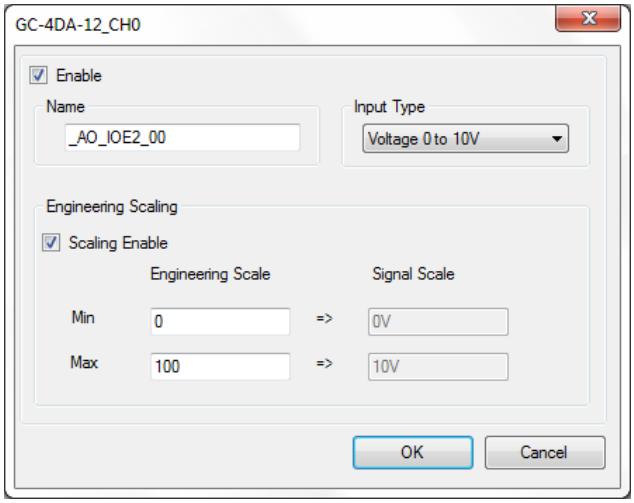
GC-4DA-12 consumes %IB12 and %QB12 when fixed in IO1 slot. It consumes %IB14 and %QB14 when fixed in IO2 slot.

Integrated tool " Hardware Configuration <R> provides Analog Outputs configuration as shown below.

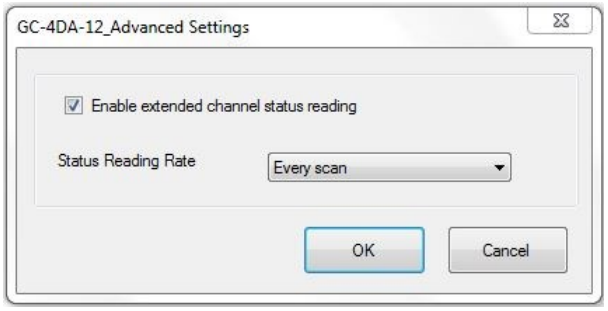


It provides default configuration and user can change it as per application requirement by selecting channel and clicking on respective highlight.

The dialogue below shows Analog Output configuration for individual channel.



The dialogue below shows Extended Settings for a unit.



The table below provides the details of configuration required for analog output channel.

Parameters	Options	Description
Analog Output configuration		
Enable	Checkbox checked	Analog output channel is enabled.
	Checkbox unchecked	Analog output channel is disabled.
Name	_AO_IOE☐_00 to _AO_IOE☐_03 (☐= 1 for unit fixed in IO1 slot 2 for unit fixed in IO2 slot)	The variable with user defined symbolic name holds analog output channel data. Global variables _AO_IOE☐_00 to _AO_IOE☐_03 are declared in global variable list ImplicitIOList <R> if respective channel is enabled. User can change the name with fixed prefix as _AO_ e.g. _AO_Velocity.
Output Type	Voltage 0 to 10V(Default) Voltage -10V to 10V Current 0 to 20mA Current 4 to 20mA	Selection for type of output as per application requirement.
Enable Scaling	Checkbox unchecked	Channel data (hold by global variables _AO_IOE☐_00 to _AO_IOE☐_03) can have value as per basic resolution e.g. 0 to 4000 to generate 0 to 10 VDC for output type 0 to 10VDC.
	Checkbox checked	Channel data (hold by global variables _AO_IOE☐_00 to _AO_IOE☐_03) scaling to engineering units as per user defined values of Min and Max.
Engineering Scaling Min	Default value is 0	Enter channel data value necessary to generate - 0V output for output type 0 to 10VDC - -10V output for output type -10 to 10VDC - 0mA output for output type 0 to 20mA - 4mA output for output type 4 to 20mA
Engineering Scaling Min	Default value is 100	Enter channel data value necessary to generate - 10V output for output type 0 to 10VDC, -10 to 10VDC - 20mA output for output type 0 to 20mA, 4 to 20mA

Configuration required for analog output channel…

Parameters	Options	Description
Extended Settings		
Enable extended channel status reading	Checked : Extended channel status reading enabled Unchecked : Extended channel status reading disabled	If checked, enables status reading for all channels. Channel status read from unit is available in global variable array `_IOE☐ChannelStatus`.
Status reading rate	Every scan 100ms 1000ms	Defines channel status reading interval.

Based on this configuration, " Hardware Configuration <R> generates a global variable list " ImplicitIOList <R> which provides declaration of all the I/O points for configured units as well as predefined variables as per the feature configured depending of type of unit configured.

Screen shot below shows default variable list for GC-4DA-12 unit fixed in IO1 slot when all the 4 channels are enabled.

```
(*                                  IOE1 : GC-4DA-12                                                                    *)
(*************************************************************************************************************************)
   _IOE1UnitState                          :       BYTE;
(*--------Analog Output--------*)
   _AO_IOE1_00                             :       REAL;           (*AO 00, IOE1 *)
   _AO_IOE1_01                             :       REAL;           (*AO 01, IOE1 *)
   _AO_IOE1_02                             :       REAL;           (*AO 02, IOE1 *)
   _AO_IOE1_03                             :       REAL;           (*AO 03, IOE1 *)

   _IOE1CH0OK          AT      %IX12.0     :       BOOL;
   _IOE1CH1OK          AT      %IX12.1     :       BOOL;
   _IOE1CH2OK          AT      %IX12.2     :       BOOL;
   _IOE1CH3OK          AT      %IX12.3     :       BOOL;
   _IOE1No24V          AT      %IX12.7     :       BOOL;

   _IOE1CH0Enable      AT      %QX12.0     :       BOOL;
   _IOE1CH1Enable      AT      %QX12.1     :       BOOL;
   _IOE1CH2Enable      AT      %QX12.2     :       BOOL;
   _IOE1CH3Enable      AT      %QX12.3     :       BOOL;

   _IOE1ChannelStatus                      :       ARRAY [0..3] OF WORD;   (*Channel Status, IOE1 *)

(*************************************************************************************************************************)
```

The table below provides the details of global variables related to GC-4DA-12.

☐=1 for unit fixed in IO1 slot, ☐=2 for unit fixed in IO2 slot,

Global Variables	Data Type	Description
`_IOE☐UnitState`	BYTE	Holds current state IO Extension unit as **0**: No unit is fixed in IO slot or No unit is detected in IO slot by CPU **1**: Mismatch between configured unit and attached unit in IO slot. **2**: Configured unit is detected and it is in configuration state **100**: Configured unit is detected, configured successfully and is in running condition.

Global variables related to GC-4DA-12…

<table>
<tr><th>Global Variables</th><th>Data Type</th><th>Description</th></tr>
<tr><td>_AO_IOE☐_00</td><td>REAL</td><td>Holds analog output channel 0 data.
The table below provides channel data available when Engineering Scaling is not enabled.
<table>
<tr><th>Channel Data</th><th>Output Type</th></tr>
<tr><td>0 to 4000</td><td>0 to 10 VDC</td></tr>
<tr><td>-2000 to 2000</td><td>-10 to 10 VDC</td></tr>
<tr><td>0 to 4000</td><td>0 to 20 mA</td></tr>
<tr><td>0 to 4000</td><td>4 to 20 mA</td></tr>
</table>
In case if Engineering Scaling is enabled, channel data holds value as per Min and Max values defined for Engineering Scaling to generate proportional output</td></tr>
<tr><td>_AO_IOE☐_01</td><td>REAL</td><td>Holds analog output channel1 data.as explained for _AO_IOE☐_00</td></tr>
<tr><td>_AO_IOE☐_02</td><td>REAL</td><td>Holds analog output channel2 data.as explained for _AO_IOE☐_00</td></tr>
<tr><td>_AO_IOE☐_03</td><td>REAL</td><td>Holds analog output channel3 data.as explained for _AO_IOE☐_00</td></tr>
<tr><td>_IOE☐ChannelStatus</td><td>ARRAY [0..3] OF WORD</td><td>Holds status of channels if Extended Settings is enabled.
Each array element is assigned for individual channel e.g._IOE☐ChannelStatus[0] holds status of channel 0.
Details of bits of status word as follows
<table>
<tr><th>Bit No</th><th>Details</th></tr>
<tr><td>0</td><td>Channel enable status
0 - Disabled
1 - Enabled</td></tr>
<tr><td>1</td><td>Channel configuration
0 - Valid
1 - Invalid</td></tr>
<tr><td>2</td><td>Channel data (written to global variables _AO_IOE☐_0n) is out of range as defined by basic resolution or engineering scaling
0: Data count valid
1: Data count invalid</td></tr>
<tr><td>3</td><td>Open circuit or short circuit detected at output
0 - No open circuit
1 - Open circuit</td></tr>
<tr><td>4 - 15</td><td>Reserved</td></tr>
</table></td></tr>
</table>

The table below provides the details of I/O bits related to GC-4DA-12.

☐=1 for unit fixed in IO1 slot, ☐=2 for unit fixed in IO2 slot,

I/O Variables	Address		Description
	IO1 Slot	IO2 Slot	
_IOE☐CH0OK	%IX12.0	%IX14.0	**TRUE**: - Respective channel is enabled and healthy. **FALSE**: - Respective channel is disabled. - Enabled channel has invalid configuration. - Open circuit or short circuit detected at output
_IOE☐CH1OK	%IX12.1	%IX14.1	
_IOE☐CH2OK	%IX12.2	%IX14.2	
_IOE☐CH3OK	%IX12.3	%IX14.3	
_IOE☐No24V	%IX12.7	%IX14.7	**TRUE**: - Unit supply absent - Unit supply polarity reversed - Unit supply below specified 18 VDC
_IOE☐CH0Enable	%QX12.0	%QX14.0	**TRUE** - Enable respective channel through *Hardware Configuration Tool*. **FALSE:** - Disable respective channel through *Hardware Configuration Tool*.
_IOE☐CH1Enable	%QX12.1	%QX14.1	
_IOE☐CH2Enable	%QX12.2	%QX14.2	
_IOE☐CH3Enable	%QX12.3	%QX14.3	

9 I/O Extension Unit GC-4UAD-10

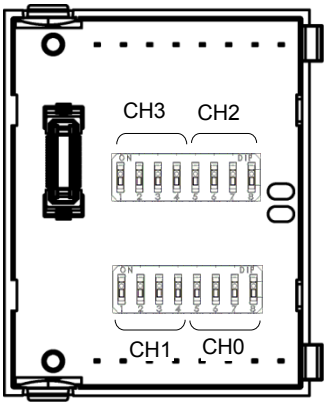
This is 4 Ch. analog voltage/ current/ 3-wire PT100 input extension unit that provides 10-bit resolution. User can attach 2 I/O extension units on the back side of Main unit.

9.1 Specifications

Item		Description		
Number of inputs		4 channels voltage/current/ 3-wire PT100 (385), single ended, non-isolated, 10-bit resolution		
Input types (Configurable through DIP switch setting on unit backside)		**Voltage**	**Current**	**3-Wire PT100**
		0 to 10 VDC	0 to 20 mA	-50 to 150°C
Output data		0 to 1000	0 to 1000	-50.0 to 150.0
1-bit Resolution		10 mV	20 µA	0.24°C
Overall accuracy (% of full scale) *1	At 25°C	±0.3	±0.3	±0.4
	At 55°C	±0.4	±0.4	±1.5
Input impedance		1 MΩ	135 Ω	Not applicable
Sensor excitation		Not applicable		1 mA
Lead wire resistance		Not applicable		20 Ω max. per wire
Unit updation time (Depends on no. of avg. samples or digital filter time constant but in synchronization with PLC scan)		Channel data updation = Controller scan time x Number of averaging samples. Channel data updation = Controller scan time + (Digital Filter Time Constant*5) minimum		
Absolute maximum input		±30 VDC/ 30 mA		
Isolation		No isolation		
Method of termination		2 nos. 8-pin terminal blocks, fixed, screw type		
Status indication		On LCD screen		
Dimensions (in mm)		61.5 (W) x 75 (H) x 24.5 (D)		
Weight (in grams)		60		

*1 Accuracy is measured with default filter time of 50 msec

For each analog input channel, 4-DIP switch settings are provided for input type selection, which is placed on the back side of extension unit.



Input Type	DIP switch settings							
	1	2	3	4	5	6	7	8
	Ch1/ Ch3				Ch0/ Ch2			
3 wire PT100 (385)	ON	ON	OFF	OFF	ON	ON	OFF	OFF
Voltage	OFF	OFF	ON	OFF	OFF	OFF	ON	OFF
Current	OFF	OFF	OFF	ON	OFF	OFF	OFF	ON

For any other undefined settings of DIP switch, for all input types (3-wire PT100, voltage, current) channel data holds maximum count of respective configured input type

User should ensure that DIP switch settings are set as per input type configuration through Hardware Configuration Tool. In case of mismatch or invalid setting, channel data behavior will be as explained in the table below

Input Type configuration through Tool	DIP Switch setting	Behaviour of analog input channel
0 to 10V	Current [0 to 20 mA]	Channel data holds minimum count of configured input type
	PT 100 [-50° to 150°C]	Channel holds maximum count of configured input type
	Any other setting	
0 to 20mA	Voltage [0 to 10 VDC]	Channel holds minimum count of configured input type
	PT 100 [-50° to 150°C]	Channel holds maximum count of configured input type
	Any other setting	
3-wire PT 100 [-50° to 150°C]	Current [0 to 20 mA]	Channel holds minimum count of configured input type
	Voltage [0 to 10 VDC]	
	Any other setting	Channel holds maximum count of configured input type

9.2 Wiring

I/O extension unit provides 4 terminals per channel **CS**, **V**, **I** and **C**.

1. Voltage input is connected between **V** and **C**.
2. Current input is connected between **I** and **C**.
3. 3-wire PT100 sensor is connected between **V** and **C** with lead compensation cable connected to **CS**.

The wiring diagram shows how to connect field input devices to I/O extension unit. For an example, channel 0 is connected to voltage input, channel 2 is connected to current input and channel 3 is connected to 3-wire PT100 sensor.

Refer section *Wiring*, before wiring analog input sensors to I/O extension unit.

- It is recommended to use 2-core shielded twisted pair cable for carrying analog signal.
- For analog sensors, PT100 sensors, use cable provided/recommended by the sensor manufacturer. Follow the recommendations provided by sensor manufacturer
- Connect cable shield at I/O extension unit end directly to a good quality earth. It is recommended to keep cable shield at sensor end unconnected.
- The Earthing resistance should be 100 Ω or less.

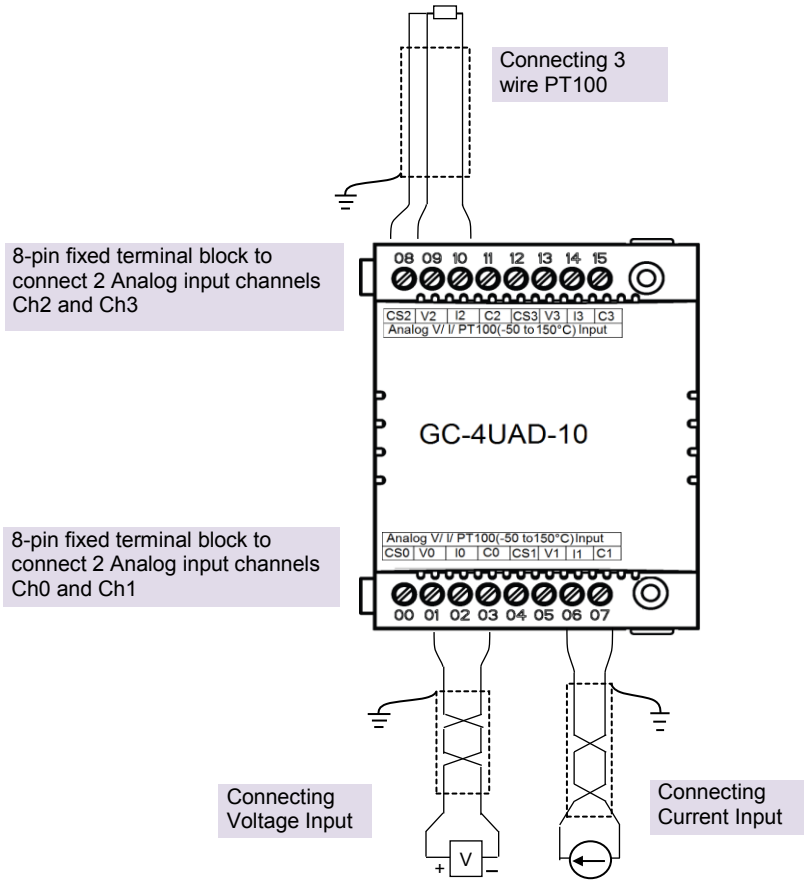


Figure 22: Wiring GC-4UAD-10

For PT100 [-50 to 150°C] type of input, sensor is connected between terminals **V** and **C** with lead compensation cable connected to **CS.**

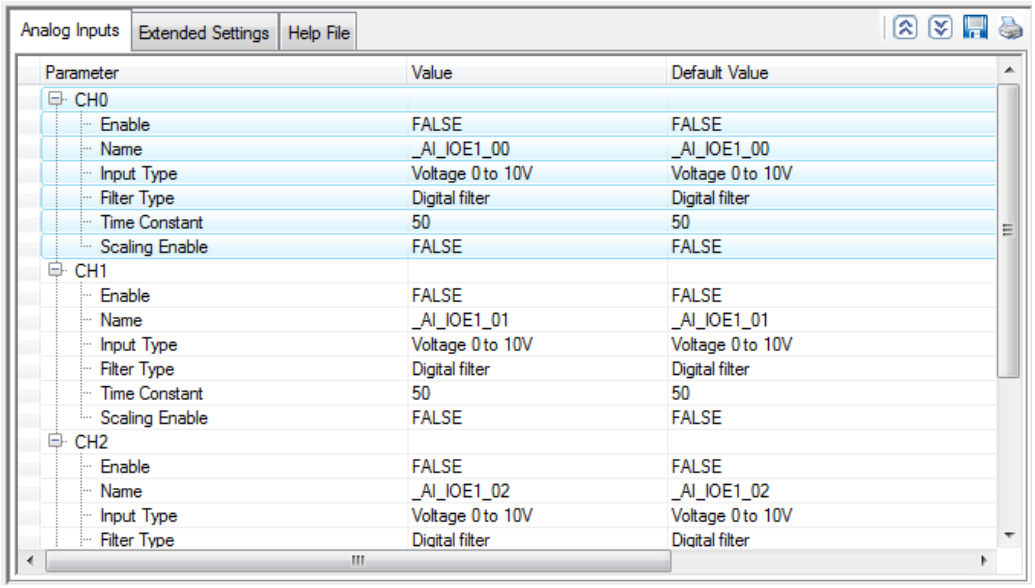
If user connects **voltage**/ **current** input to **CS** terminal, then it damages connected input channel.

9.3 Configuration and Programming

For GOC I/O Extension units, I/O memory map is fixed.

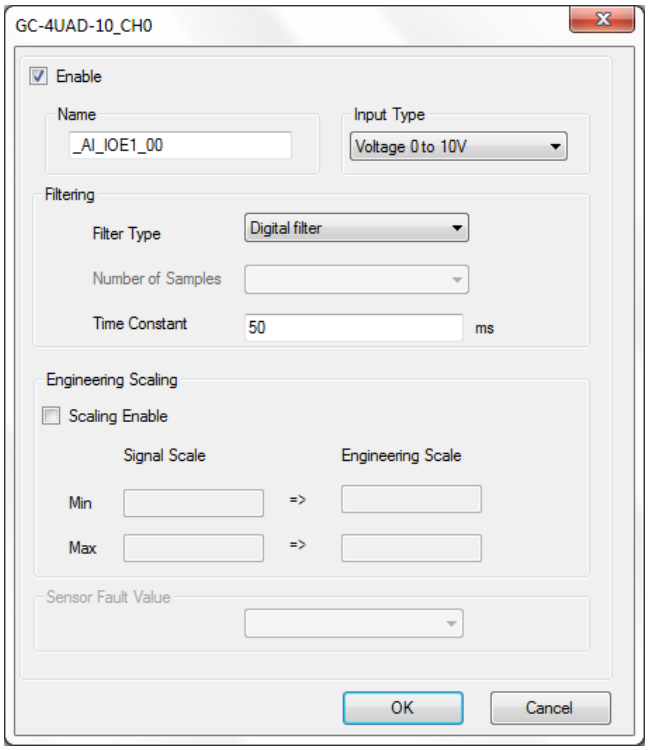
GC-4UAD-10 consumes %IB12 and %QB12 when fixed in IO1 slot. It consumes %IB14 and %QB14 when fixed in IO2 slot.

Integrated tool " Hardware Configuration <R> provides Analog Inputs configuration as shown below.

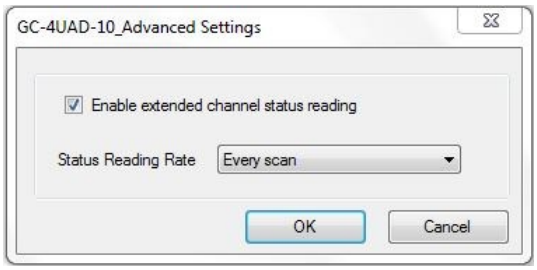


It provides default configuration user can change it as per application requirement by selecting channel and clicking on respective highlight.

The dialogue below shows Analog Input configuration for individual channel.



The dialogue below shows Extended Settings for a unit.



The table below provides the details of configuration required for analog input channel.

Parameter	Options	Description
Analog Input configuration		
Enable	Checkbox checked	Analog input channel is enabled.
	Checkbox unchecked	Analog input channel is disabled.
Name	`_AI_IOE☐_00` to `_AI_IOE☐_03` (☐= 1 for unit fixed in IO1 slot 2 for unit fixed in IO2 slot)	The variable with user defined symbolic name holds analog input channel data. Global variables `_AI_IOE☐_00` to `_AI_IOE☐_03` are declared in global variable list ImplicitIOList <R> if respective channel is enabled. User can change the name with fixed prefix as `_AI_` e.g. `_AI_TemeperatureZone1`.

Configuration required for analog input channel…

Parameters	Options	Description
Input Type	Voltage 0 to 10V (Default) Current 0 to 20mA PT100 -50 to 150ºC	Selection for type of input as per application requirement. For input type selection, it is necessary to set DIP switch settings provided on the back side of unit also. Refer section *Specifications*
Filter type	No Filter Averaging Digital Filter (Default)	Selection for signal conditioning by software
Number of Samples	4, 8, 16, 32	Selection for number of samples for moving average. Channel data updation = Controller scan time x Number of averaging samples. Apply averaging for slowly varying analog input signal
Time Constant	Default value 50 ms	Enter digital filer time constant value from 10 to 5000 ms Channel data updation = Controller scan time + (Time Constant*5). Apply digital filter for analog input signal with erroneous fast variations.
Enable Scaling	Checkbox unchecked	Channel data holds value as per basic resolution e.g. 0 to 1000 if input signal is 0 to 10 VDC for input type 0 to 10VDC.
	Checkbox checked	Channel data scaling to engineering units as per user defined values of Min and Max. User defined Engineering Scaling is not supported for input type PT100 -50 to 150ºC.
Engineering Scaling Min	Default value=0	Enter channel data value required for - 0V input for input type 0 to 10VDC - 0mA input for input type 0 to 20mA
Engineering Scaling Max	Default value 100	Enter channel data value required for - 10V input for input type 0 to 10VDC, - 20mA input for input type 0 to 20mA
Sensor Fault Value	0 (Default) Minimum Value Maximum Value Last Value	Applicable for input type PT100 (-50 to 150ºC). Defines value of channel data in case if sensor fault is detected at input channel. Minimum value and maximum depends upon Engineering scaling selected.
Extended Settings		
Enable extended channel status reading	Checked : Extended channel status reading enabled Unchecked : Extended channel status reading disabled	If checked, enables status reading for all channels. Channel status read from unit is available in global variable array `_IOE□ChannelStatus`.
Status reading rate	Every scan (Default) 100ms 1000ms	Defines channel status reading interval.

Based on this configuration, " Hardware Configuration <R> tool generates a global variable list " ImplicitIOList <R> which provides declaration of all the I/O points for configured units as well as predefined variables as per the feature configured depending of type of unit configured.

Screen shot below shows default variable list for GC-4UAD-10 unit fixed in IO1 slot when all the 4 channels are enabled.

```
(*                         IOE1 : GC-4UAD-10                                   *)
(*****************************************************************************)
  _IOE1UnitState                    :    BYTE;
(*--------Analog Input--------*)
  _AI_IOE1_00                       :    REAL;           (* AI 00, IOE1 *)
  _AI_IOE1_01                       :    REAL;           (* AI 01, IOE1 *)
  _AI_IOE1_02                       :    REAL;           (* AI 02, IOE1 *)
  _AI_IOE1_03                       :    REAL;           (* AI 03, IOE1 *)

  _IOE1CH0OK        AT   %IX12.0   :    BOOL;
  _IOE1CH1OK        AT   %IX12.1   :    BOOL;
  _IOE1CH2OK        AT   %IX12.2   :    BOOL;
  _IOE1CH3OK        AT   %IX12.3   :    BOOL;


  _IOE1CH0Enable    AT   %QX12.0   :    BOOL;
  _IOE1CH1Enable    AT   %QX12.1   :    BOOL;
  _IOE1CH2Enable    AT   %QX12.2   :    BOOL;
  _IOE1CH3Enable    AT   %QX12.3   :    BOOL;

  _IOE1ChannelStatus             :    ARRAY [0..3] OF WORD;   (*Channel Status, IOE1 *)

(*****************************************************************************)
```

The table below provides the details of global variables related to GC-4UAD-10.

☐=1 for unit fixed in IO1 slot, ☐=2 for unit fixed in IO2 slot,

<table>
<tr><th>Global Variables</th><th>Data Type</th><th colspan="2">Description</th></tr>
<tr><td>_IOE☐UnitState</td><td>BYTE</td><td colspan="2">Holds current state IO Extension unit as
0: No unit is fixed in IO slot or No unit is detected in IO slot by CPU
1: Mismatch between configured unit and attached unit in IO slot.
2: Configured unit is detected and it is in configuration state
100: Configured unit is detected, configured successfully and is in running condition.</td></tr>
<tr><td rowspan="6">_AI_IOE☐_00</td><td rowspan="6">REAL</td><td colspan="2">Holds analog input channel0 data.
The table below provides channel data available when Engineering Scaling is not enabled.</td></tr>
<tr><th>Input Type</th><th>Channel Data</th></tr>
<tr><td>0 to 10 VDC</td><td>0 to 1000</td></tr>
<tr><td>0 to 20 mA</td><td>0 to 1000</td></tr>
<tr><td>3-wire PT100 [-50 to 150ºC]</td><td>-50.0 to 150.0</td></tr>
<tr><td colspan="2">In case if Engineering Scaling is enabled, channel data holds value as per Min and Max values defined for Engineering Scaling.
User defined Engineering Scaling is not applicable for input type 3-wire PT100 [-50 to 150ºC].</td></tr>
</table>

Global variables related to GC-4UAD-10…

Global Variables	Data Type	Description
_AI_IOE☐_01	REAL	Holds analog input channel1 data.as explained for _AI_IOE☐_00
_AI_IOE☐_02	REAL	Holds analog input channel2 data.as explained for _AI_IOE☐_00
_AI_IOE☐_03	REAL	Holds analog input channel3 data.as explained for _AI_IOE☐_00
_IOE☐ChannelStatus	ARRAY [0..3] OF WORD	Holds status of channels if Extended Settings is enabled. Each array element is assigned for individual channel e.g._IOE☐ChannelStatus[0] holds status of channel 0. Details of bits of status word as follows

Bit No	Details
0	Channel enable status 0 - Disabled 1 - Enabled
1	Channel configuration 0 - Valid 1 - Invalid
2	PT100 sensor fault 0 - No open circuit 1 - Open circuit
3 - 15	Reserved

The table below provides the details of I/O bits related to GC-4UAD-10.

☐=1 for unit fixed in IO1 slot, ☐=2 for unit fixed in IO2 slot,

I/O Variables	Address		Description
	IO1 Slot	IO2 Slot	
_IOE☐CH0OK	%IX12.0	%IX14.0	**TRUE**: - Respective channel is enabled and healthy. **FALSE**: - Respective channel is disabled. - Enabled channel has invalid configuration. - Open circuit for PT100 sensor
_IOE☐CH1OK	%IX12.1	%IX14.1	
_IOE☐CH2OK	%IX12.2	%IX14.2	
_IOE☐CH3OK	%IX12.3	%IX14.3	
_IOE☐CH0Enable	%QX12.0	%QX14.0	**TRUE** - Enable respective channel through *Hardware Configuration Tool*. **FALSE:** Disable respective channel through *Hardware Configuration Tool*.
_IOE☐CH1Enable	%QX12.1	%QX14.1	
_IOE☐CH2Enable	%QX12.2	%QX14.2	
_IOE☐CH3Enable	%QX12.3	%QX14.3	

10 I/O Extension Unit GC-4UAD-10E

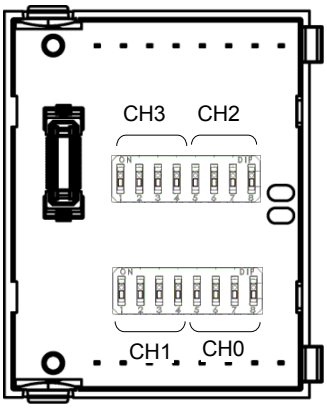
This is 4 Ch. analog voltage/ current/ 3-wire PT100 input extension unit that provides 10-bit resolution. User can attach up to 2 I/O extension units on the back side of Main unit.

10.1 Specifications

Item		Description		
Number of inputs		4 channels voltage/current/ 3-wire PT100 (385), single ended, non-isolated, 10-bit resolution		
Input types (Configurable through DIP switch setting on unit backside)		**Voltage**	**Current**	**3-Wire PT100**
		0 to 10 VDC	0 to 20 mA	-50 to 450°C
Output data		0 to 1000	0 to 1000	-50.0 to 450.0
1-bit Resolution		10 mV	20 µA	0.6°C
Overall accuracy (% of full scale) *1	At 25°C	±0.3	±0.3	±0.4
	At 55°C	±0.4	±0.4	±1.5
Input impedance		1 MΩ	135 Ω	Not applicable
Sensor excitation		Not applicable		1 mA
Lead wire resistance		Not applicable		20Ω max. per wire
Unit updation time (Depends on no. of avg. samples or digital filter time constant but in synchronization with PLC scan)		Channel data updation = Controller scan time x Number of averaging samples. Channel data updation = Controller scan time + (Digital Filter Time Constant*5) minimum		
Absolute maximum input		±30 VDC/ 30 mA		
Isolation		No isolation		
Method of termination		2 nos. 8-pin terminal blocks, fixed, screw type		
Status indication		On LCD screen		
Dimensions (in mm)		61.5 (W) x 75 (H) x 24.5 (D)		
Weight (in grams)		60		

*1 Accuracy is measured with default filter time of 50 msec

For each analog input channel, 4-DIP switch settings are provided for input type selection, which is placed on the back side of extension unit.



	DIP switch settings							
Input Type	1	2	3	4	5	6	7	8
	Ch1/ Ch3				Ch0/ Ch2			
3 wire PT100 (385)	ON	ON	OFF	OFF	ON	ON	OFF	OFF
Voltage	OFF	OFF	ON	OFF	OFF	OFF	ON	OFF
Current	OFF	OFF	OFF	ON	OFF	OFF	OFF	ON

For any other undefined settings of DIP switch, for all input types (3-wire PT100, voltage, current) channel data holds maximum count of respective configured input type

User should ensure that DIP switch settings are set as per input type configuration through Hardware Configuration Tool. In case of mismatch or invalid setting, channel data behavior will be as explained in the table below

Input Type configuration through Tool	DIP Switch setting	Behaviour of analog input channel
0 to 10V	Current [0 to 20 mA]	Channel holds minimum count for configured input type
	PT 100 [-50° to 450°C]	Channel holds maximum count for configured input type
	Any other setting	
0 to 20mA	Voltage [0 to 10 VDC]	Channel holds minimum count for configured input type
	PT 100 [-50° to 450°C]	Channel holds maximum count for configured input type
	Any other setting	
3-wire PT 100 [-50° to 450°C]	Current [0 to 20 mA]	Channel holds minimum count for configured input type
	Voltage [0 to 10 VDC]	
	Any other setting	Channel holds maximum count for configured input type

10.2 Wiring

I/O extension unit provides 4 terminals per channel **CS**, **V**, **I** and **C**.

1. Voltage input is connected between **V** and **C**.
2. Current input is connected between **I** and **C**.
3. 3-wire PT100 sensor is connected between **V** and **C** with lead compensation cable connected to **CS**.

The wiring diagram shows how to connect field input devices to I/O extension unit. For an example, channel 0 is connected to voltage input, channel 2 is connected to current input and channel 3 is connected to 3-wire PT100 sensor.

Refer section *Wiring*, before wiring analog input sensors to I/O extension unit.

- It is recommended to use 2-core shielded twisted pair cable for carrying analog signal.
- For analog sensors, PT100 sensors use cable provided/recommended by the sensor manufacturer. Follow the recommendations provided by sensor manufacturer
- Connect cable shield at I/O extension unit end directly to a good quality earth. It is recommended to keep cable shield at sensor end unconnected.
- The Earthing resistance should be 100 Ω or less.

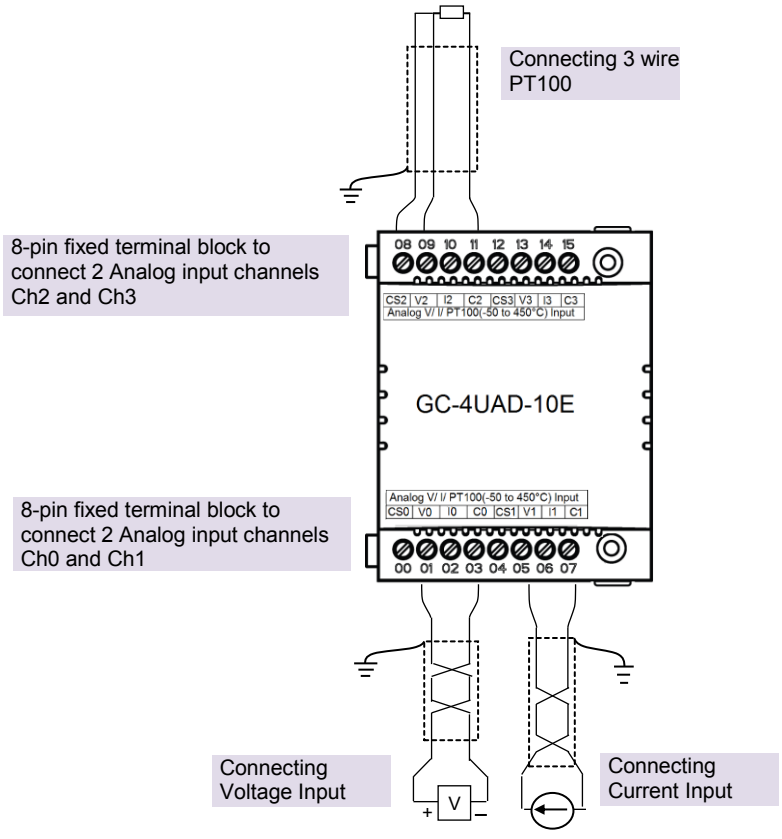


Figure 23: Wiring GC-4UAD-10E

For PT100 [-50 to 450°C] type of input, sensor is connected between terminals **V** and **C** with lead compensation cable connected to **CS.**

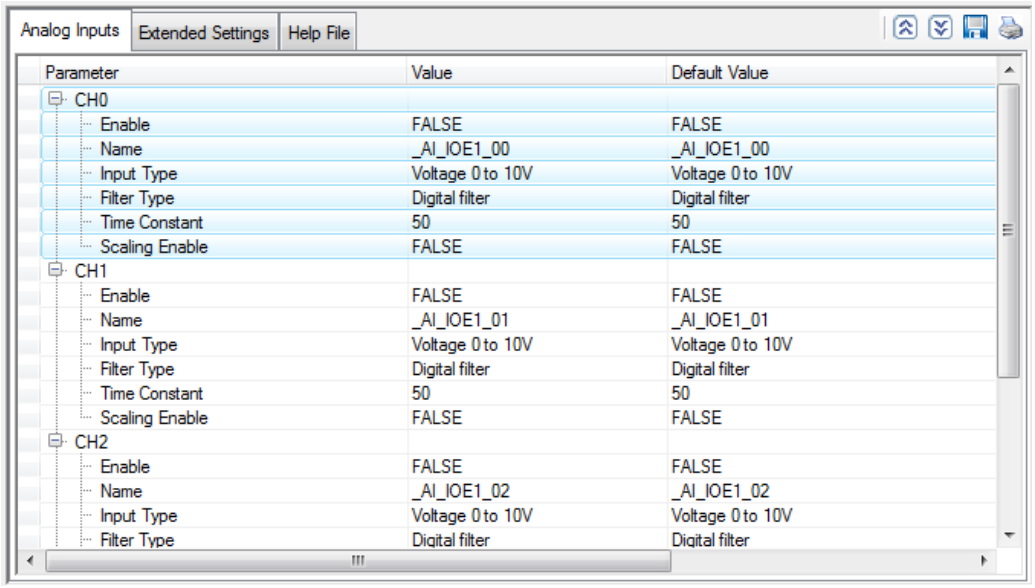
If user connects **voltage**/ **current** input to **CS** terminal, then it damages connected input channel.

10.3 Configuration and Programming

For GOC I/O Extension units, I/O memory map is fixed.

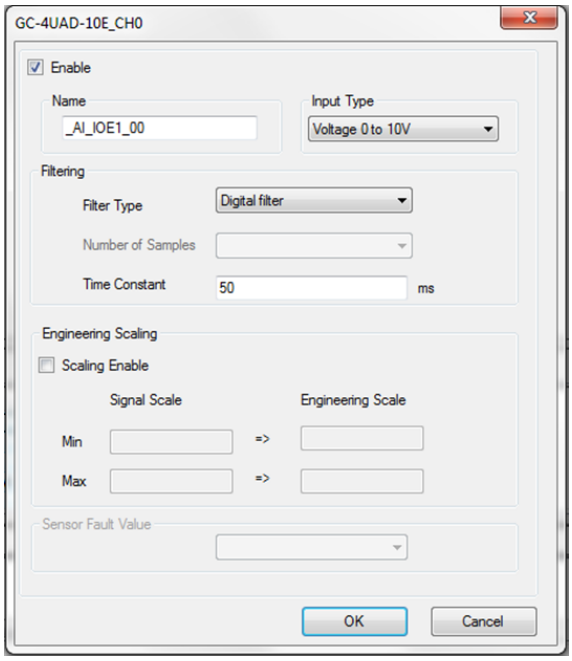
GC-4UAD-10E consumes %IB12 and %QB12 when fixed in IO1 slot. It consumes %IB14 and %QB14 when fixed in IO2 slot.

Integrated tool " Hardware Configuration <R> provides Analog Inputs configuration as shown below.

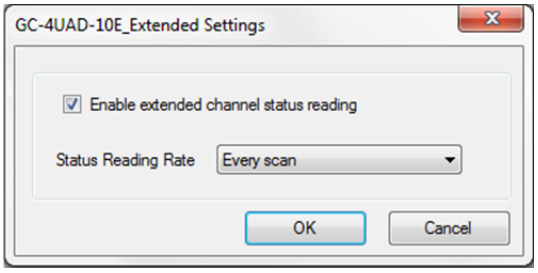


It provides default configuration user can change it as per application requirement by selecting channel and clicking on respective highlight.

The dialogue below shows Analog Input configuration for individual channel.



The dialogue below shows Extended Settings for a unit.



The table below provides the details of configuration required for analog input channel.

Parameter	Options	Description
Analog Input configuration		
Enable	Checkbox checked	Analog input channel is enabled.
	Checkbox unchecked	Analog input channel is disabled.
Name	`_AI_IOE☐_00` to `_AI_IOE☐_03` (☐= 1 for unit fixed in IO1 slot 2 for unit fixed in IO2 slot)	The variable with user defined symbolic name holds analog input channel data. Global variables `_AI_IOE☐_00` to `_AI_IOE☐_03` are declared in global variable list ImplicitIOList <R> if respective channel is enabled. User can change the name with fixed prefix as `_AI_` e.g. `_AI_TemeperatureZone1`.

Configuration required for analog input channel…

Parameters	Options	Description
Input Type	Voltage 0 to 10V (Default) Current 0 to 20mA PT100 -50 to 450ºC	Selection for type of input as per application requirement. For input type selection, it is necessary to set DIP switch settings provided on the back side of unit also. Refer section *Specifications*
Filter type	No Filter Averaging Digital Filter (Default)	Selection for signal conditioning by software
Number of Samples	4, 8, 16, 32	Selection for number of samples for moving average. Channel data updation = Controller scan time x Number of averaging samples. Apply averaging for slowly varying analog input signal
Time Constant	Default value 50 ms	Enter digital filer time constant value from 10 to 5000 ms Channel data updation = Controller scan time + (Time Constant*5). Apply digital filter for analog input signal with erroneous fast variations.
Enable Scaling	Checkbox unchecked	Channel data holds value as per basic resolution e.g. 0 to 1000 if input signal is 0 to 10 VDC for input type 0 to 10VDC.
	Checkbox checked	Channel data scaling to engineering units as per user defined values of Min and Max. User defined Engineering Scaling is not supported for input type PT100 -50 to 450ºC.
Engineering Scaling Min	Default value=0	Enter channel data value required for - 0V input for input type 0 to 10VDC - 0mA input for input type 0 to 20mA
Engineering Scaling Max	Default value=100	Enter channel data value required for - 10V input for input type 0 to 10VDC, - 20mA input for input type 0 to 20mA
Sensor Fault Value	0 (Default) Minimum Value Maximum Value Last Value	Applicable for input type PT100 (-50 to 450ºC). Defines value of channel data in case if sensor fault is detected at input channel. Minimum value and maximum depends upon Engineering scaling selected.
Extended Settings		
Enable extended channel status reading	Checked : Extended channel status reading enabled Unchecked : Extended channel status reading disabled	If checked, enables status reading for all channels. Channel status read from unit is available in global variable array `_IOE□ChannelStatus`.
Status reading rate	Every scan (Default) 100ms 1000ms	Defines channel status reading interval.

Based on this configuration, " Hardware Configuration <R> tool generates a global variable list " ImplicitIOList <R> which provides declaration of all the I/O points for configured units as well as predefined variables as per the feature configured depending of type of unit configured.

Screen shot below shows default variable list for GC-4UAD-10E unit fixed in IO1 slot when all the 4 channels are enabled.

```
(*****************************************************************************************************************)
(*                                        IOE1 : GC-4UAD-10E                                                     *)
(*****************************************************************************************************************)
    _IOE1UnitState                          :       BYTE;
(*--------Analog Input--------*)
    _AI_IOE1_00                             :       REAL;           (* AI 00, IOE1 *)
    _AI_IOE1_01                             :       REAL;           (* AI 01, IOE1 *)
    _AI_IOE1_02                             :       REAL;           (* AI 02, IOE1 *)
    _AI_IOE1_03                             :       REAL;           (* AI 03, IOE1 *)

    _IOE1CH0OK          AT      %IX12.0     :       BOOL;
    _IOE1CH1OK          AT      %IX12.1     :       BOOL;
    _IOE1CH2OK          AT      %IX12.2     :       BOOL;
    _IOE1CH3OK          AT      %IX12.3     :       BOOL;


    _IOE1CH0Enable      AT      %QX12.0     :       BOOL;
    _IOE1CH1Enable      AT      %QX12.1     :       BOOL;
    _IOE1CH2Enable      AT      %QX12.2     :       BOOL;
    _IOE1CH3Enable      AT      %QX12.3     :       BOOL;
```

The table below provides the details of global variables related to GC-4UAD-10E.

☐=1 for unit fixed in IO1 slot, ☐=2 for unit fixed in IO2 slot,

<table>
<tr><th>Global Variables</th><th>Data Type</th><th>Description</th></tr>
<tr><td>_IOE☐UnitState</td><td>BYTE</td><td>Holds current state IO Extension unit as
0: No unit is fixed in IO slot or No unit is detected in IO slot by CPU
1: Mismatch between configured unit and attached unit in IO slot.
2: Configured unit is detected and it is in configuration state
100: Configured unit is detected, configured successfully and is in running condition.</td></tr>
<tr><td>_AI_IOE☐_00</td><td>REAL</td><td>Holds analog input channel0 data.
The table below provides channel data available when Engineering Scaling is not enabled.
<table>
<tr><th>Input Type</th><th>Channel Data</th></tr>
<tr><td>0 to 10 VDC</td><td>0 to 1000</td></tr>
<tr><td>0 to 20 mA</td><td>0 to 1000</td></tr>
<tr><td>3-wire PT100 [-50 to 450ºC]</td><td>-50.0 to 450.0</td></tr>
</table>
In case if Engineering Scaling is enabled, channel data holds value as per Min and Max values defined for Engineering Scaling.
User defined Engineering Scaling is not applicable for input type 3-wire PT100 [-50 to 450ºC].</td></tr>
</table>

Global variables related to GC-4UAD-10E…

Global Variables	Data Type	Description
_AI_IOE☐_01	REAL	Holds analog input channel1 data.as explained for _AI_IOE☐_00
_AI_IOE☐_02	REAL	Holds analog input channel2 data.as explained for _AI_IOE☐_00
_AI_IOE☐_03	REAL	Holds analog input channel3 data.as explained for _AI_IOE☐_00
_IOE☐ChannelStatus	ARRAY [0..3] OF WORD	Holds status of channels if Extended Settings is enabled. Each array element is assigned for individual channel e.g._IOE☐ChannelStatus[0] holds status of channel 0. Details of bits of status word as follows

Bit No	Details
0	Channel enable status 0 - Disabled 1 - Enabled
1	Channel configuration 0 - Valid 1 - Invalid
2	PT100 sensor fault 0 - No open circuit 1 - Open circuit
3 - 15	Reserved

The table below provides the details of I/O bits related to GC-4UAD-10E.

☐=1 for unit fixed in IO1 slot, ☐=2 for unit fixed in IO2 slot,

I/O Variables	Address		Description
	IO1 Slot	IO2 Slot	
_IOE☐CH0OK	%IX12.0	%IX14.0	**TRUE**: - Respective channel is enabled and healthy. **FALSE**: - Respective channel is disabled. - Enabled channel has invalid configuration. - Open circuit for PT100 sensor
_IOE☐CH1OK	%IX12.1	%IX14.1	
_IOE☐CH2OK	%IX12.2	%IX14.2	
_IOE☐CH3OK	%IX12.3	%IX14.3	
_IOE☐CH0Enable	%QX12.0	%QX14.0	**TRUE** - Enable respective channel through *Hardware Configuration Tool*. **FALSE:** Disable respective channel through *Hardware Configuration Tool*.
_IOE☐CH1Enable	%QX12.1	%QX14.1	
_IOE☐CH2Enable	%QX12.2	%QX14.2	
_IOE☐CH3Enable	%QX12.3	%QX14.3	

11 I/O Extension Unit GC-4UAD-16

This is 4 Ch. universal analog voltage/ current/ thermocouple/ milli volt / 3-wire PT100/ P1000 input extension unit that provides 16-bit resolution. User can attach up to 2 I/O extension units on the back side of Main unit.

11.1 Specifications

<table>
<tr><th>Item</th><th colspan="5">Description</th></tr>
<tr><td>Number of inputs</td><td colspan="5">4 channel voltage/ current/ thermocouple/ milli volt/ 3-wire PT100 /PT1000,
Differential, non-isolated, 16-bit resolution</td></tr>
<tr><td rowspan="12">Input types</td><td rowspan="2">Input Type</td><td rowspan="2">Resolution</td><td rowspan="2">Output Data</td><td colspan="2">Overall Accuracy (% of FSD)</td></tr>
<tr><td>At 25°C</td><td>At 55°C</td></tr>
<tr><td>0 to 10 Vdc</td><td>0.15 mV</td><td>0 to 64000</td><td>±0.2</td><td>±0.3</td></tr>
<tr><td>±10 Vdc</td><td>0.3 mV</td><td>-32000 to 32000</td><td>±0.2</td><td>±0.3</td></tr>
<tr><td>±100 mV</td><td>3 µV</td><td>-32000 to 32000</td><td>±0.1</td><td>±0.2</td></tr>
<tr><td>0 to 20 mA</td><td>0.3 µA</td><td>0 to 64000</td><td>±0.2</td><td>±0.3</td></tr>
<tr><td>4 to 20 mA</td><td>0.3 µA</td><td>0 to 64000</td><td>±0.2</td><td>±0.3</td></tr>
<tr><td>PT100 (385)</td><td>0.1 °C</td><td>-200.0 to 850.0</td><td>±0.3</td><td>±0.6</td></tr>
<tr><td>PT100 (385)</td><td>0.01 °C</td><td>-50.00 to 250.00</td><td>±0.5</td><td>±1</td></tr>
<tr><td>PT1000 (385)</td><td>0.01 °C</td><td>-50.0 to 250.0</td><td>±0.4</td><td>±0.6</td></tr>
<tr><td>J Type Tc</td><td>0.1 °C</td><td>-100.0 to 1200.0</td><td>±0.5</td><td>±1</td></tr>
<tr><td>K Type Tc</td><td>0.1 °C</td><td>-100.0 to 1372.0</td><td>±0.5</td><td>±1</td></tr>
<tr><td>Scaling to engineering units</td><td colspan="5">For voltage, milli volt -and current input types</td></tr>
<tr><td>Input impedance</td><td colspan="5">Voltage Input: > 1 MΩ,
Current Input: 124Ω,
Thermocouple/ mVolts input: > 100 KΩ</td></tr>
<tr><td>Sensor excitation</td><td colspan="5">For PT100, 1 mA
For PT1000, 0.1 mA</td></tr>
<tr><td>Lead wire resistance</td><td colspan="5">30Ω max. per wire [Applicable only for 3-wire PT100, PT1000 input types]</td></tr>
<tr><td>Cold junction compensation</td><td colspan="5">Range: From 0 to 100 °C for thermocouple input</td></tr>
<tr><td>Channel updation time</td><td colspan="5">[(3* PLC Scan Time) + Channel Conversion Time] * (Number of Channels Enabled)
Channel conversion time for individual input type is,
Voltage/ mVolt input : 50 ms
Current input : 25 ms
PT100/ PT1000 input/ CJC sensor : 100 ms
Thermocouple Input : 200 ms</td></tr>
<tr><td>Absolute maximum input</td><td colspan="5">±30 VDC/ ±30 mA</td></tr>
<tr><td>Method of termination</td><td colspan="5">2 nos. 8-pin terminal blocks, fixed, screw type</td></tr>
<tr><td>Status indication</td><td colspan="5">On LCD screen</td></tr>
<tr><td>Dimensions (in mm)</td><td colspan="5">61.5 (W) x 75 (H) x 24.5 (D)</td></tr>
<tr><td>Weight (in grams)</td><td colspan="5">60</td></tr>
</table>

11.2 Wiring

I/O extension unit provides 4 terminals per channel **RT+**, **RT-**, **VI+** and **VI-**.

1. Voltage input is connected between **VI+** and **VI-**.
2. Current input is connected between **VI-** and **RT-**, along with short link between terminals **VI+** and **RT-**.
3. 3-wire PT100/ PT1000 sensor is connected between **RT+** and **RT-** along with lead compensation wire connected to **VI-**.
4. Thermocouple/ mVolts input is connected between **RT+** and **RT-**.

The wiring diagram shows how to connect field input devices to I/O extension unit. For an example, channel 0 is connected to voltage input, channel 1 is connected to current input, channel 2 is connected to 3-wire PT100 sensor and channel 3 is connected to thermocouple input.

Refer section *Wiring*, before wiring analog input sensors to I/O extension unit.

It is recommended to use 2-core shielded twisted pair cable for carrying analog signal.

For sensors especially for thermocouple and PT100/ PT1000, use cable provided/ recommended by the sensor manufacturer. Follow the recommendations provided by sensor manufacturer.

Connect cable shield at I/O extension unit end directly to a good quality earth. It is recommended to keep cable shield at sensor end unconnected.

The Earthing resistance should be 100 Ω or less.

It is recommended to use thermocouple with isolated tip. Accuracy will be hampered, if non isolated type of thermocouple element is used and if it gets connected to improper earth.

Ensure that input signal is connected to relevant terminals as per configured input type. Unit may get damaged with wrong connections.

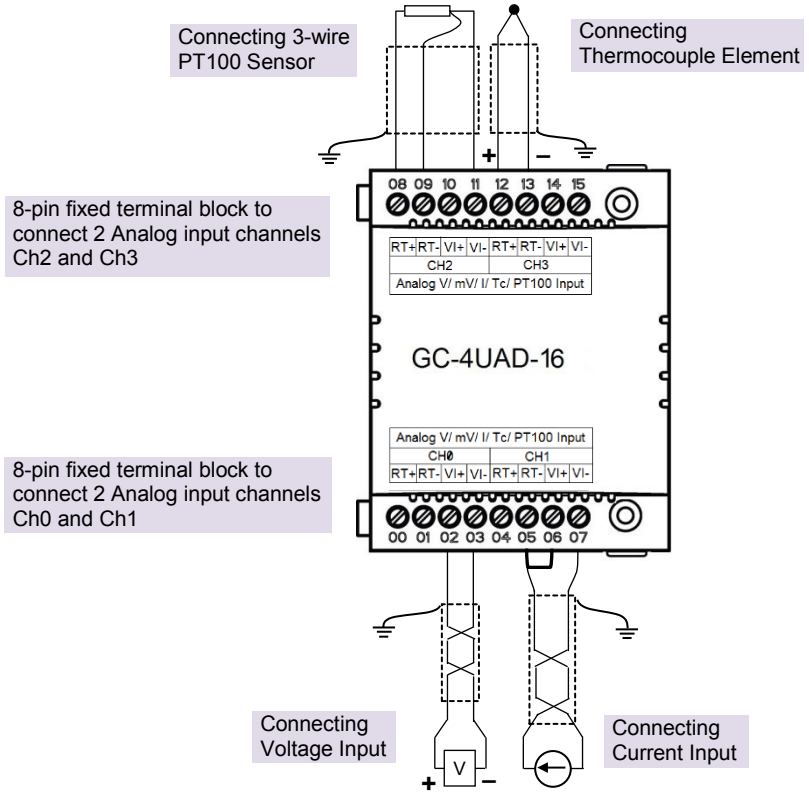


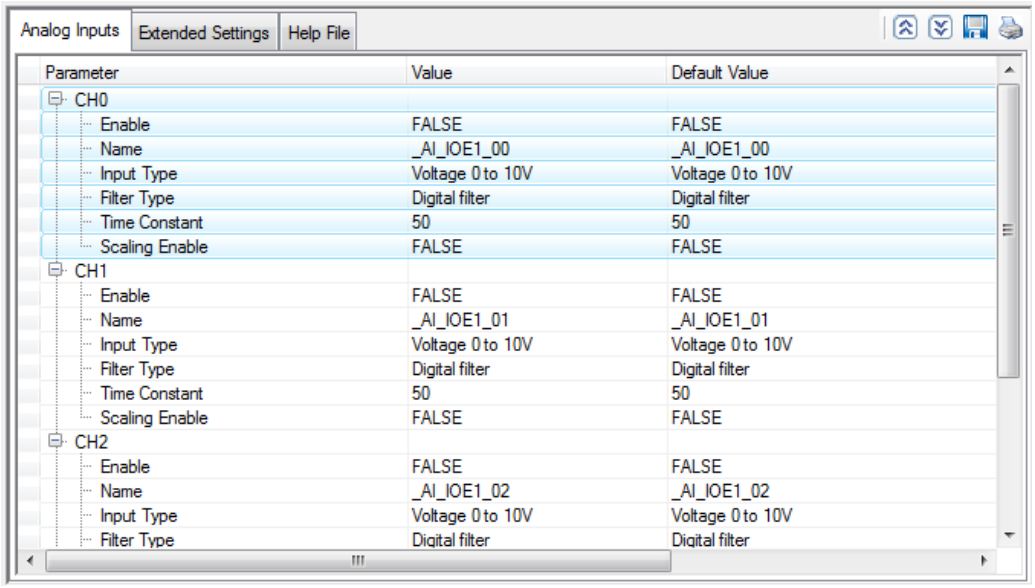
Figure 24: **Wiring GC-4UAD-16**

11.3 Configuration and Programming

For GOC I/O Extension units, I/O memory map is fixed.

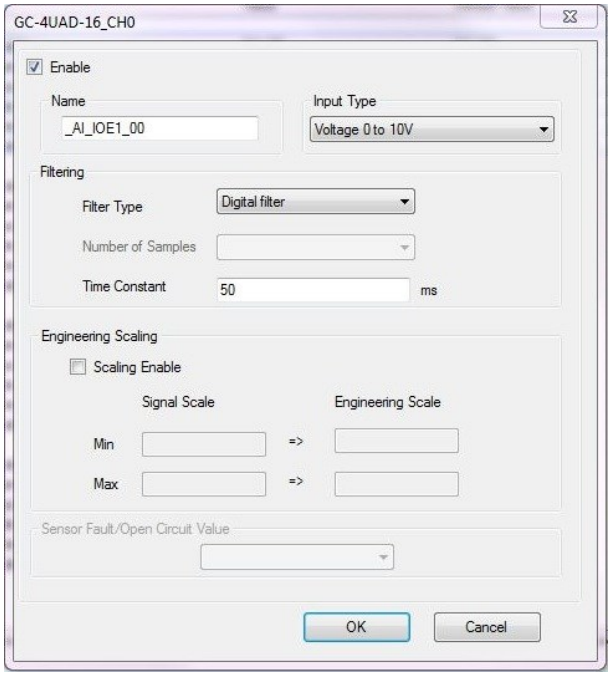
GC-4UAD-16 consumes %IB12 and %QB12 when fixed in IO1 slot. It consumes %IB14 and %QB14 when fixed in IO2 slot.

Integrated tool " Hardware Configuration <R> provides Analog Inputs configuration as shown below.



It provides default configuration. User can change it as per application requirement by selecting channel and clicking on respective highlight.

The dialogue below shows Analog Input configuration for individual channel.



The dialogue below shows Extended Settings for a unit.

The table below provides the details of configuration required for analog input channel.

Parameter	Options	Description
Analog Input configuration		
Enable	Checkbox checked	Analog input channel is enabled.
	Checkbox unchecked	Analog input channel is disabled.
Name	`_AI_IOE☐_00` to `_AI_IOE☐_03` (☐= 1 for unit fixed in IO1 slot 2 for unit fixed in IO2 slot)	The variable with user defined symbolic name holds analog input channel data. Global variables `_AI_IOE☐_00` to `_AI_IOE☐_03` are declared in global variable list ImplicitIOList <R> if respective channel is enabled. User can change the name with fixed prefix as `_AI_` e.g. `_AI_TemeperatureZone1`.
Input Type	Voltage: 0 to 10Vdc (Default) -10 to +10Vdc -100 to +100mV Current: 0 to 20mA 4 to 20mA PT100: 200 to 600°C -200 to 250°C PT1000 : -200 to 250°C Thermocouple: J type K type	Selection for type of input as per application requirement.
Filter type	No Filter Averaging Digital Filter (Default)	Selection for signal conditioning by software
Number of Samples	4, 8, 16, 32	Selection for number of samples for moving average. Channel data updation = Controller scan time x Number of averaging samples. Apply averaging for slowly varying analog input signal
Time Constant	Default value 50 ms	Enter digital filer time constant value from 10 to 5000 ms Channel data updation = Controller scan time + (Time Constant*5). Apply digital filter for analog input signal with erroneous fast variations.

Scaling Enable	Checkbox unchecked	Channel data holds value as per basic resolution e.g. 0 to 64000 if input signal is 0 to 10 VDC for input type 0 to 10VDC.
	Checkbox checked	Channel data scaling to engineering units as per user defined values of Min and Max. User defined Engineering Scaling is not supported for thermocouple and 3-wire PT100/ PT1000 input types.
Engineering Scaling Min	Default value=0	Enter channel data value as per the application requirement. - 0 for input type 0 to 10VDC - -100 for input type ±10VDC, ±100mV - 0 for input type 0 to 20mA, 4 to 20mA
Engineering Scaling Max	Default value=100	Enter channel data value as per the application requirement. - 100 for input types 0 to 10VDC, ±10VDC, ±100mV, for input type 0 to 20mA, 4 to 20mA
Sensor Fault/ Open circuit Value	0 (Default) Minimum Value Maximum Value Last Value	Applicable for thermocouple, 4 to 20mA and 3-wire PT100/ 1000 input types. Defines value of channel data in case if sensor fault is detected at input channel *1 Minimum value and maximum depends upon Engineering scaling selected.

*1 Sensor fault detection is provided for thermocouple, 4 to 20mA and 3-wire PT100/ 1000 input types Sensor fault is detected in case if sensor is open/ gets disconnected/ in case of malfunctioning.

Extended Settings		
Enable extended channel status reading	Checked : Extended channel status reading enabled Unchecked : Extended channel status reading disabled	If checked, enables status reading for all channels. Channel status read from unit is available in global variable array `_IOE☐ChannelStatus`.
Status reading rate	Every scan (Default) 100ms 1000ms	Defines channel status reading interval.

Based on this configuration, " Hardware Configuration <R> tool generates a global variable list " ImplicitIOList <R> which provides declaration of all the I/O points for configured units as well as predefined variables as per the feature configured depending of type of unit configured.

Screen shot below shows default variable list for GC-4UAD-16 unit fixed in IO1 slot when all the 4 channels are enabled.

```
(*****************************************************************************************************************)
(*                                    IOE1 : GC-4UAD-16                                                          *)
(*****************************************************************************************************************)
    _IOE1UnitState                          :       BYTE;
(*--------Analog Input--------*)
    _AI_IOE1_00                             :       REAL;           (* AI 00, IOE1 *)
    _AI_IOE1_01                             :       REAL;           (* AI 01, IOE1 *)
    _AI_IOE1_02                             :       REAL;           (* AI 02, IOE1 *)
    _AI_IOE1_03                             :       REAL;           (* AI 03, IOE1 *)

    _IOE1CH0OK          AT      %IX12.0     :       BOOL;
    _IOE1CH1OK          AT      %IX12.1     :       BOOL;
    _IOE1CH2OK          AT      %IX12.2     :       BOOL;
    _IOE1CH3OK          AT      %IX12.3     :       BOOL;

    _IOE1CH0Enable      AT      %QX12.0     :       BOOL;
    _IOE1CH1Enable      AT      %QX12.1     :       BOOL;
    _IOE1CH2Enable      AT      %QX12.2     :       BOOL;
    _IOE1CH3Enable      AT      %QX12.3     :       BOOL;
```

The table below provides the details of global variables related to GC-4UAD-16.

☐=1 for unit fixed in IO1 slot, ☐=2 for unit fixed in IO2 slot,

<table>
<tr><th>Global Variables</th><th>Data Type</th><th>Description</th></tr>
<tr><td>_IOE☐UnitState</td><td>BYTE</td><td>Holds current state IO Extension unit as
0: No unit is fixed in IO slot or No unit is detected in IO slot by CPU
1: Mismatch between configured unit and attached unit in IO slot.
2: Configured unit is detected and it is in configuration state
100: Configured unit is detected, configured successfully and is in running condition.</td></tr>
<tr><td>_AI_IOE☐_00</td><td>REAL</td><td>Holds analog input channel0 data.
The table below provides channel data available when Engineering Scaling is not enabled.
<table>
<tr><th>Input Type</th><th>Channel Data</th></tr>
<tr><td>0 to 10 Vdc</td><td>0 to 64000</td></tr>
<tr><td>±10 Vdc</td><td>-32000 to 32000</td></tr>
<tr><td>±100 mV</td><td>-32000 to 32000</td></tr>
<tr><td>0 to 20mA</td><td>0 to 64000</td></tr>
<tr><td>4 to 20mA</td><td>0 to 64000</td></tr>
<tr><td>PT100</td><td>-200.0 to 850.0</td></tr>
<tr><td>PT100</td><td>-50.00 to 250.00</td></tr>
<tr><td>PT1000</td><td>-50.0 to 250.0</td></tr>
<tr><td>J Type Tc</td><td>-100.0 to 1200.0</td></tr>
<tr><td>K Type Tc</td><td>-100.0 to 1372.0</td></tr>
</table></td></tr>
</table>

		In case if Engineering Scaling is enabled, channel data holds value as per Min and Max values defined for Engineering Scaling. User defined Engineering Scaling is not supported for thermocouple and 3-wire (PT100, PT1000) input types.

User defined setting for “Engineering Scale” is applicable only for voltage and current input types. (i.e. 0 to 10Vdc, ±10Vdc, ±100 mV, 0 to 20mA and 4 to 20mA input types).

Global variables related to GC-4UAD-16…

Global Variables	Data Type	Description
_AI_IOE☐_01	REAL	Holds analog input channel1 data as explained for _AI_IOE☐_00
_AI_IOE☐_02	REAL	Holds analog input channel2 data as explained for _AI_IOE☐_00
_AI_IOE☐_03	REAL	Holds analog input channel3 data as explained for _AI_IOE☐_00
_AI_IOE☐_CJCData	INT	Holds CJC data
_IOE☐ChannelStatus	ARRAY [0..3] OF WORD	Holds status of channels if Extended Settings is enabled. Each array element is assigned for individual channel e.g._IOE☐ChannelStatus [0] holds status of channel 0. Details of bits of status word as follows

Bit No	Details
0	Channel enable status 0 - Disabled 1 - Enabled
1	Channel configuration 0 - Invalid 1 - Valid
2	Sensor fault 0 - No open circuit 1 - Open circuit for Thermocouple, PT100, PT1000 and 4 to 20mA input ranges and CJC sensor fault.
3	CJC Sensor fault 0 - CJC sensor is healthy 1 - CJC Sensor is faulty
4 - 15	Reserved

***Note**:* CJC data i.e. terminal temperature is updated when any channel is configured for any input type.

The table below provides the details of I/O bits related to GC-4UAD-16.

☐=1 for unit fixed in IO1 slot, ☐=2 for unit fixed in IO2 slot,

I/O Variables	Address		Description
	IO1 Slot	IO2 Slot	
_IOE☐CH0OK	%IX12.0	%IX14.0	**TRUE**: - Respective channel is enabled and healthy. **FALSE**: - Respective channel is disabled. - Enabled channel has invalid configuration. - Open circuit for PT100/ PT1000 sensor, thermocouple and, 4 to 20mA input types.
_IOE☐CH1OK	%IX12.1	%IX14.1	
_IOE☐CH2OK	%IX12.2	%IX14.2	
_IOE☐CH3OK	%IX12.3	%IX14.3	
_IOE☐CJCFault	%IX12.4	%IX14.4	**TRUE:** - CJC faulty - CJC Sensor Open or Short - CJC value below 0°C or beyond 100°C **FALSE:** CJC healthy
_IOE☐ADCFault	%IX12.6	%IX14.6	**TRUE:** - ADC faulty **FALSE:** - ADC healthy
_IOE☐CH0Enable	%QX12.0	%QX14.0	**TRUE** - Enable respective channel through *Hardware Configuration Tool*. **FALSE:** - Disable respective channel through *Hardware Configuration Tool*.
_IOE☐CH1Enable	%QX12.1	%QX14.1	
_IOE☐CH2Enable	%QX12.2	%QX14.2	
_IOE☐CH3Enable	%QX12.3	%QX14.3	

***Note**:* In module memory, CJC data gets updated for any input channel irrespective of input type configuration. It is recommended to consider CJC count for thermocouple input type configuration only.

12 I/O Extension Unit GC-4HSXPTY

This I/O extension unit (GC-4HSXPTY) provides 2 Ch. High Speed Counter/Encoder, 24VDC Sink/Source type inputs and 2 Ch. Pulse Train Output differential type outputs. It can be fixed only in IO1 slot on the backside of Main unit.

12.1 Specifications

Item	Description	
High Speed Counter		
No of Channels	2	
Input Type	Sink/ Source (18 to 30 VDC including ripple).	
Current	7 mA max @ 24 VDC	
Frequency	100 KHz max for single phase counter with hardware and software direction control 50 KHz max for quadrature encoder interface	
Min On/ OFF Time	4.0 µS	
Isolation	Photo coupler isolation, 2.5 KV from input circuit.	
Input Terminals	I00/A1, I01/B1, I02/Z1 and C0 for HSC/Encoder input1. I03/A2, I04/B2, I05/Z2 and C1 for HSC/Encoder input2.	
Counting Range	-2147483648 to 2147483647	
Modes of Operation for HSC (Software Selectable)		
Digital Input	10 ms (Digital filter) provided HSC functionality is not configured at this I/O extension unit or Main unit	
Single Phase Counter	Software direction control	100 KHz
	Hardware direction control	
Quadrature Encoder A phase, B phase	Without Z	50 KHz max (individual phase)
	With Z	
HSC Interrupt Event		
EVENT_HSC10_CMP, EVENT_HSC13_CMP	Interrupt event execution when HSC count become equals with preset comparison value for channel 10/13.	

Specifications of GC-4HSXPTY…

Item	Description	
Z Input		
No of channels	2, inputs I02 and I05	
Input type	Sink/ Source (18 to 30 VDC with ripple)	
Min On Time	4.0 µS	
Min Off Time	200 µS	
Modes of Operation for Z_Config (Software Selectable)	Ignore occurrence of Z	
	Reset CV on occurrence of Z	
Z INTERRUPT EVENT		
EVENT_IX12_R_TRIG, EVENT_IX15_R_TRIG	Interrupt event execution on rising edge of Z pulse for input I02, input I05.	
EVENT_IX12_F_TRIG, EVENT_IX15_F_TRIG	Interrupt event execution on falling edge of Z pulse for input I02, input I05.	
Pulse Train Output		
No of Channels	2	
Output Type	Differential output @ 5V	
Current	40 mA max	
Frequency	100 KHz max	
Duty Cycle	50%	
Pulse Output Range	0 to 4294967295	
Mode of Operation	Pulse and Direction	
Protection	No output short circuit protection	
Isolation from internal circuit	No isolation from internal circuit	
Output Terminals	**Pulse output:** P1+, P1- for Ch1, P2+, P2- for Ch2	
	Direction output: D1+, D1- for Ch1, D2+, D2- for Ch2	
	8-pin terminal block (fixed, screw type)	1 no. at upper side for input
		1 no. at lower side for output

User can have HSC functionality and Hardware input rising/ falling edge interrupt functionality either from Main unit (channel numbers 0/3) or I/O Extension unit (channel numbers 10/13) at a time.

When any functionality is configured on Main unit, user should not fix I/O Extension unit GC-4HSXPTY in IO1 slot. User is informed and care is taken during configuration using Hardware Configuration Tool.

12.2 Wiring

I/O extension unit provides two 8-pin fixed terminal blocks for wiring I/O devices. One is located at lower side of unit is for pulse train outputs (Differential type 5 VDC) and another is located at upper side of unit is for high speed (sink/ source type) inputs.

I/O extension unit provides 1 common for a group of 3 inputs. For an example, as shown below in figure the wiring diagram shows inputs I00 to I02 connected for Sink and Source type of operation..

At output side, load is connected between differential outputs P1+, P1- for pulse output and between D1+, D1- for direction output.

The wiring diagram shows how to connect field input devices like sink and source operation.

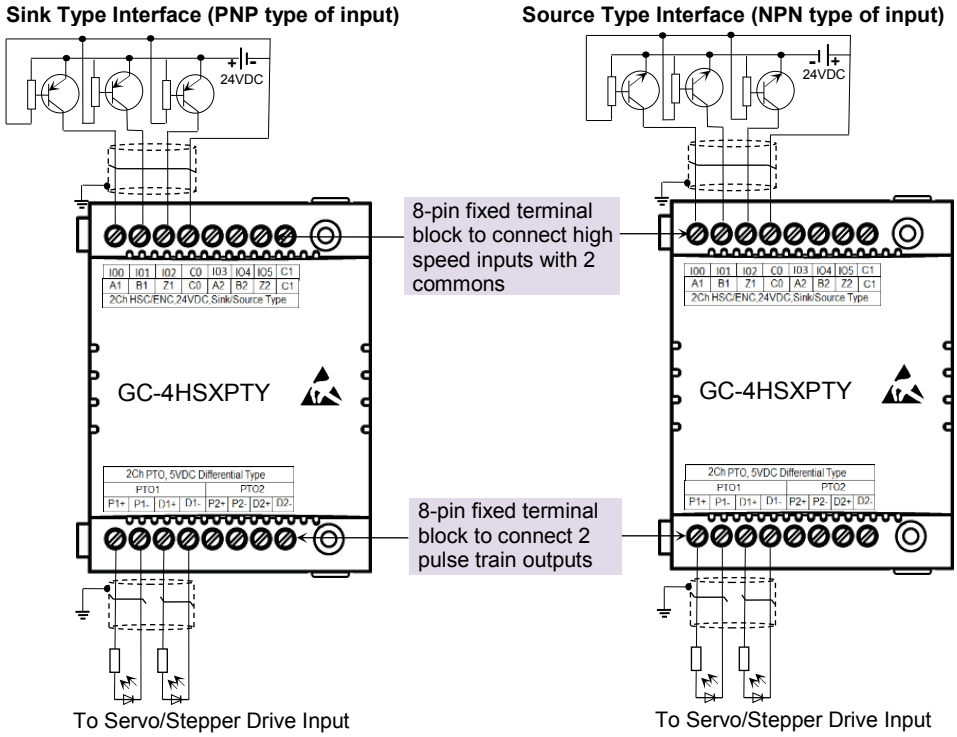


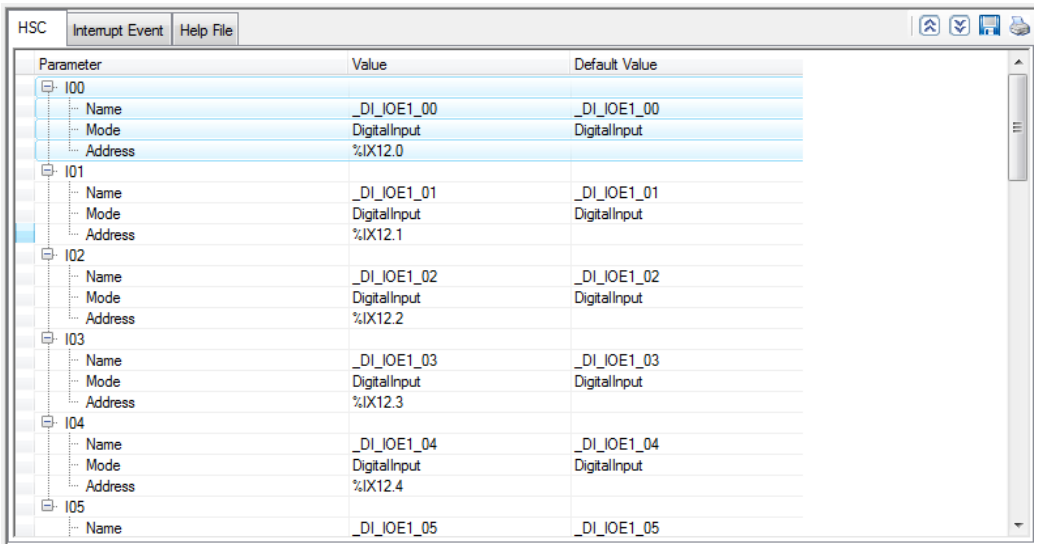
Figure 25: Wiring GC-4HSXPTY

Avoid short circuit at pulse train output. It may damage hardware permanently

12.3 Configuration and Programming

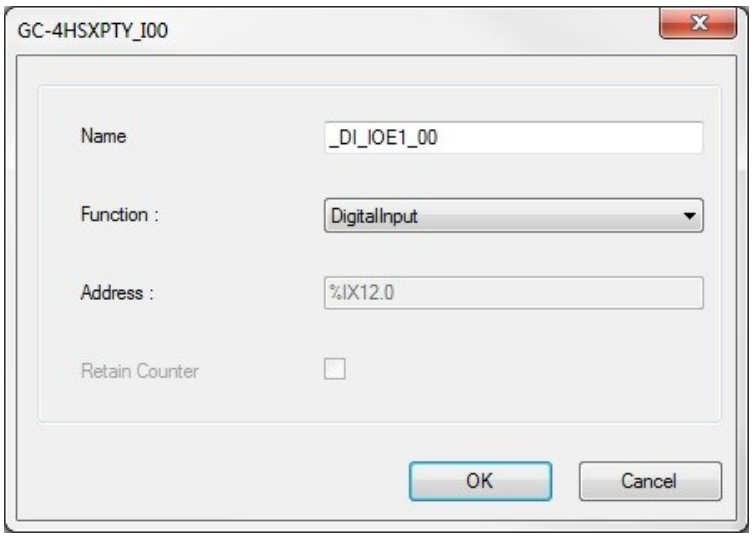
For GOC I/O Extension units, I/O memory map is fixed.
GC-4HSXPTY consumes %IB12 and %QB12 when fixed in IO1 slot.

Integrated tool " Hardware Configuration <R> provides HSC Input configuration as shown below.



GC-4HSXPTY extension unit is configurable and accessible only in IO1 slot.

Default configuration i.e. Digital Input, provides predefined symbolic naming of each input.
For input I00, name is _DI_IOE1_0 and address is %IX12.0.
Prefix is _DI_. Text IOE1_0 indicates that unit is fixed in IO1 slot and input is I00.
Based on this configuration, integrated tool " Hardware Configuration <R> generates Global Variable List, " ImplicitIOList <R> which provides declaration of all inputs for configured unit.



After adding, GC-4HSXPTY I/O extension unit to IO1 slot, click on CoDeSys tab "Resources"→ "ImplicitIOList <R>, then by default, following variable list can be viewed.

```
(*****************************************************************************************************************)
(*                                    IOE1 : GC-4HSXPTY
(*****************************************************************************************************************)
    _IOE1UnitState                          :       BYTE;
(*--------Digital Input--------*)
    _DI_IOE1_00         AT      %IX12.0     :       BOOL;           (* DI 00, IOE1 *)
    _DI_IOE1_01         AT      %IX12.1     :       BOOL;           (* DI 01, IOE1 *)
    _DI_IOE1_02         AT      %IX12.2     :       BOOL;           (* DI 02, IOE1 *)
    _DI_IOE1_03         AT      %IX12.3     :       BOOL;           (* DI 03, IOE1 *)
    _DI_IOE1_04         AT      %IX12.4     :       BOOL;           (* DI 04, IOE1 *)
    _DI_IOE1_05         AT      %IX12.5     :       BOOL;           (* DI 05, IOE1 *)

(*****************************************************************************************************************)
```

Output bits from %QB12 are not is use.

The table below provides the details of global variables related to GC-4HSXPTY

Global Variables	Data Type	Description
_IOE1UnitState	BYTE	Holds current state IO Extension unit as **0**: No unit is fixed in IO slot or No unit is detected in IO slot by CPU **1**: Mismatch between configured unit and attached unit in IO slot. **2**: Configured unit is detected and it is in configuration state **100**: Configured unit is detected, configured successfully and is in running condition.

The table below provides the details of I/O bits related to GC-4HSXPTY.

I/O Variables	Address IO1 Slot	Description
_DI_IOE1_0	%IX12.0	Holds ON/OFF status of extension unit input I00
_DI_IOE1_1	%IX12.1	Holds ON/OFF status of extension unit input I01
_DI_IOE1_2	%IX12.2	Holds ON/OFF status of extension unit input I02
_DI_IOE1_3	%IX12.3	Holds ON/OFF status of extension unit input I03
_DI_IOE1_4	%IX12.4	Holds ON/OFF status of extension unit input I04
_DI_IOE1_5	%IX12.5	Holds ON/OFF status of extension unit input I05

12.3.1 High Speed Input Configuration

User can configure mode of operation and inputs are consumed accordingly as shown in the table below.

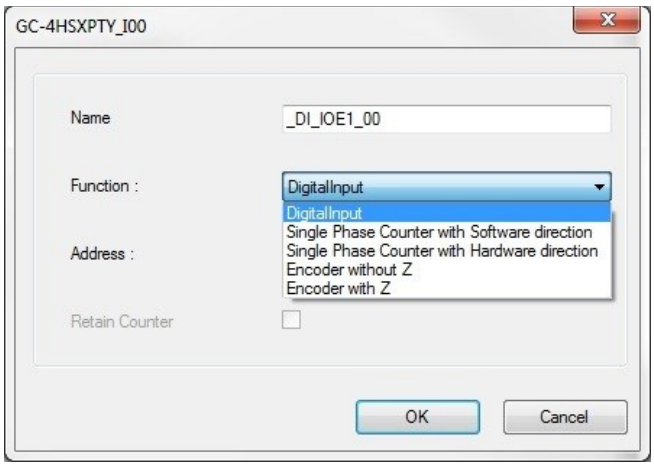
Description \ Inputs	I00	I01	I02	I03	I04	I05
Single phase counter with software direction control	HSC10	-	-	HSC13	-	-
	100 KHz	-	-	100 KHz	-	-
Single phase counter with Hardware direction control	HSC10		-	HSC13		-
	100KHz		-	100KHz		-
Quadrature encoder A phase, B phase	ENCAB10		-	ENCAB13		-
	50 KHz		-	50 KHz		-
Quadrature encoder A phase, B phase with Z pulse	ENCABZ10			ENCABZ13		
	50 KHz			50 KHz		

Refer *High Speed Counter* for more detail.

User can have HSC functionality and Hardware input rising/ falling edge interrupt functionality either from Main unit (channel numbers 0/3) or I/O Extension unit (channel numbers 10/13) at a time.

When any functionality is configured on Main unit, user should not fix I/O Extension unit GC-4HSXPTY in IO1 slot. User is informed and care is taken during configuration using Hardware Configuration Tool.

User can configure counter functionality for digital input I00 and I03 by selecting respective input from digital input configuration tab and clicking on respective highlight.



Based on this configuration, " Hardware Configuration <R> generates a Global Variable List " ImplicitIOList <R> which provides declaration of predefined global variables.

The list below shows declaration of predefined variables for configuration of single phase counter with software direction control.

```
(*-------- High Speed Counter 10 '_DI_IOE1_00' Parameters--------*)
    _HSC10_En                :      BOOL;          (* Enable Counting HSC10 *)
    _HSC10_Dir               :      BOOL;          (* Counting Direction HSC10 *)
    _HSC10_Reset             :      BOOL;          (* Reset HSC10 count *)
    _HSC10_Load              :      BOOL;          (* Load Preset value HSC10 *)
    _HSC10_PV                :      DINT;          (* Preset value HSC10 *)
    _HSC10_CV                :      DINT;          (* Current value HSC10 *)
```

The list below shows declaration of predefined variables for configuration of single phase counter with hardware direction control.

```
(*-------- High Speed Counter 10 with Hardware Direction '_DI_IOE1_00' Parameters--------*)
    _HSC10_En                :      BOOL;          (* Enable Counting HSC10 *)
    _HSC10_Reset             :      BOOL;          (* Reset HSC10 count *)
    _HSC10_Load              :      BOOL;          (* Load Preset value HSC10 *)
    _HSC10_PV                :      DINT;          (* Preset value HSC10 *)
    _HSC10_CV                :      DINT;          (* Current value HSC10 *)
```

The list below shows declaration of predefined variables for configuration of encoder at digital inputs I00, I01 without Z marker input.

```
(*--------ENCODER 10 '_DI_IOE1_00' Parameters--------*)
    _ENC10_En                :      BOOL;          (* Enable Counting ENCODER10 *)
    _ENC10_Reset             :      BOOL;          (* Reset ENCODER10 count *)
    _ENC10_Load              :      BOOL;          (* Load Preset value ENCODER10 *)
    _ENC10_PV                :      DINT;          (* Preset value ENCODER10 *)
    _ENC10_CV                :      DINT;          (* Current Value ENCODER10 *)
```

The list below shows declaration of predefined variables for configuration of encoder with Z marker at digital inputs I00, I01, I02.

```
(*--------ENCODER 10 '_DI_IOE1_00' Parameters--------*)
    _ENC10_En                :      BOOL;          (* Enable Counting ENCODER10 *)
    _ENC10_Reset             :      BOOL;          (* Reset ENCODER10 count *)
    _ENC10_Load              :      BOOL;          (* Load Preset value ENCODER10 *)
    _ENC10_PV                :      DINT;          (* Preset value ENCODER10 *)
    _ENC10_Z_En              :      BOOL;          (* Enable Z Signal of ENCODER10 *)
    _ENC10_CV                :      DINT;          (* Current Value ENCODER10 *)
```

The table below provides the details of predefined global variables related to high speed counter/ encoder interface.

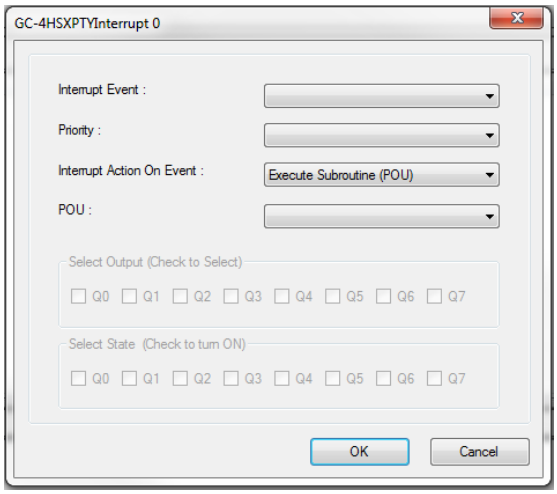
☐=10 for counter configuration for input I00 and encoder configuration at input I00/A1, I01/B1 and I02/Z1

☐=13 for counter configuration for input I03 and encoder configuration at input I03/A2, I04/B2 and I05/Z2

Single Phase Counter with Software Direction	Single Phase Counter with Hardware Direction	Encoder Without Z	Encoder With Z	Data Type	Description
_HSC☐_En	_HSC☐_En	_ENC☐_En	_ENC☐_En	BOOL	If TRUE, counter starts counting incoming pulses and updates counter value CV. If FALSE, counter does not count pulses and CV holds last counter value.
_HSC☐_Dir	Not applicable	Not applicable	Not applicable	BOOL	If TRUE, counter counts in upward direction. If FALSE, counter counts in downward direction.
_HSC☐_Reset	_HSC☐_Reset	_ENC☐_Reset	_ENC☐_Reset	BOOL	If TRUE, CV is reset to 0.
_HSC☐_Load	_HSC☐_Load	_ENC☐_Load	_ENC☐_Load	BOOL	If TRUE, counter value gets modified to a value defined by PV and CV shows modified counter value
_HSC☐_PV	_HSC☐_PV	_ENC☐_PV	_ENC☐_PV	DINT	If Load is TRUE, counter value gets modified to a value defined by PV.
Not applicable	Not applicable	Not applicable	_ENC☐_Z_En	BOOL	If FALSE, Z marker pulse is ignored If TRUE, counter is reset to 0 on occurrence of Z marker pulse.
_HSC☐_CV	_HSC☐_CV	_ENC☐_CV	_ENC☐_CV	DINT	Returns counter current value if En is TRUE. CV holds last value as long as RUN is FALSE

12.3.2 Interrupt Configuration

User can configure up to 4 interrupt events. To configure any interrupt event, click on  in Interrupt Event tab. It pops up a dialogue box as shown below.



The table below explains various options available for different parameters.

Parameter	Options	Description
Interrupt Event	EVENT_IX12_R_TRIG	Interrupt event occurs on rising and/or falling edge of digital input I02
	EVENT_IX12_F_TRIG	
	EVENT_IX15_R_TRIG	Interrupt event occurs on rising and/or falling edge of digital input I05
	EVENT_IX15_F_TRIG	
	EVENT_HSC10_CMP	Interrupt event occurs when high speed counter value reaches target value.
	EVENT_HSC13_CMP	
Priority	0 to 15	Select priority of execution in case of occurrence of multiple interrupt events. Priority 0 has higher precedence. Priority should be unique for each event.

Refer *Interrupt Events* for more detail.

Interrupt event configuration

Parameter	Options	Description
Interrupt Action On Event	Execute Subroutine (POU)	Decide action to be performed on occurrence of interrupt event. User can select predefined PROGRAM type POU to be executed on occurrence of interrupt event.
	Set Digital Output	Decide action to be performed on occurrence of interrupt event. User can select switching On/Off output/s Q00 to Q07 on Main unit. User can select output/s and desired state as On/Off. It is recommended to use this option for transistor outputs for better performance.
POU	Select already programmed PROGRAM type POU from the list	This option is available if 'Execute Subroutine (POU)' is selected for 'Interrupt Action On Event' It is necessary to program type POU before configuring interrupt event.
Select Output	Checkout particular output to select	This option is available if 'Set Digital Output' is selected for 'Interrupt Action On Event'. Select output/s from Q00 to Q07 from Main unit which is to be switched to select State (On/Off) on occurrence of selected interrupt event. It is recommended to use this option for transistor outputs for better performance.
Select State	Checkout to turn On selected output.	

Based on this configuration, " Hardware Configuration <R> generates a Global Variable List " ImplicitIOList <R> which provides declaration of predefined global variables.

The list below shows declaration of predefined variables for configuration of select output.

```
(*--------Interrupt Parameters--------*)
    _EVENT_IX12_R_TRIG_Attach      :    BOOL;              (* Attach Interrupt Event*)
    _EVENT_IX12_R_TRIG_ATCH_Done   :    BOOL;              (* Event attached successfully*)
    _EVENT_IX12_R_TRIG_OPMASK      :    BYTE  :=2#00110011;  (*  Outputs selected for updation *)
    _EVENT_IX12_R_TRIG_OPSET       :    BYTE  :=2#00100001;  (*  Output states to update *)
```

The table below provides the details of global variables related to interrupt event configuration.

□□□□□□= Name of the interrupt event configured, # HSC number 10/13,

Global Variables	Data Type	Description
_ClearInterruptQueue	BOOL	Clears queue of pending interrupt events.
_HoldAllInterrupts	BOOL	Holds processing of all attached interrupt events.
□□□□□□_Attach	BOOL	If TRUE, enables specified interrupt and configured interrupt action will be executed on occurrence of event. If FALSE, ignores interrupt event even if it is configured.
□□□□□□_ATCH_Done	BOOL	Holds status of interrupt attach command. This is read only variable. TRUE : Interrupt attached successfully, FALSE : Interrupt not attached.
_HSC#SetCompareValue	BOOL	Applicable only for interrupt event EVENT_HSC#_CMP If TRUE, respective high speed counter compare value is set to value defined by _HSC#CompareValue .
_HSC#CompareValue	DINT	Applicable only for interrupt event EVENT_HSC#_CMP. Holds value for high speed counter comparison and this value is set when Boolean variable _HSC#SetCompareValue is TRUE.
□□□□□□_OPMASK	BYTE	Displays which outputs are updated on occurrence of selected interrupt event. This is set by hardware configuration tool based on interrupt configuration done.
□□□□□□_OPSET	BYTE	Displays output states to be updated on occurrence of selected interrupt event. This is set by hardware configuration tool based on interrupt configuration done.

12.3.3 Pulse Train Output

This unit provides 2 channels of pulse train output. Hardware configuration tool does not provide any configuration for pulse train output.

User can develop program either using FB 'PTO_RUN' from 'GOC35 library' or simple positioning FBs from 'Simple Positioning' library.

The motion library provides FBs to control the motion using pulse train output. The basic FBs are divided into administrative (not driving motion) and motion related sets.

Table below provides list of simple positioning FBs

Motion	MC_MoveAbsolute
	MC_MoveRelative
	MC_MoveVelocity
	MC_Stop
Administrative	MC_ReadActualPosition
	MC_ReadBoolParameter
	MC_ReadParameter
	MC_WriteBoolParameter
	MC_WriteParameter
Derived	MC_Inch
	Position_Profile

Refer "ED-2002-540_GOC35_LibraryManual.pdf" and "ED-2002-544_Simple Positioning Library Manual.pdf" for more details.

Do not use FB 'PTO_RUN' if simple motion FBs is used in application program. Simultaneous usage may lead to malfunctioning.

13 COM Extension Unit GC-RS232-COM

This is RS232 serial communication extension units. User can attach up to 2 I/O extension units on the back side of Main unit.

13.1 Specifications

Item		Description
No. of serial ports		1
Hardware interface		RS232C
Signals		TxD, RxD, GND, Carrier detect *1
Communication parameters	Baud rate (bps)	9600, 19200, 38400, 57600, 115200
	Data bits	7, 8
	Parity	Odd, Even, None
	Stop bits	1, 2
Communication type		Full duplex or half duplex
Connector type		9-pin D male
Isolation		No isolation from main circuit
Dimensions (in mm)		26.0 (W) x 51.0 (H) x 48.0 (D)
Weight (in grams)		40

*1 RS/CS control is not supported

13.2 Wiring

This is 1 port RS232 serial communication unit. It provides 9-pin D male connector on its front side.

The figure below shows front view of COM extension unit with connection details.

Tighten both screws on 9-pin D female connector to avoid malfunctioning due to loose connections.

Do not try to pull out communication cable connector before un-tightening 2 screws. It may cause damage to the electronic hardware /plastic enclosure of COM extension unit.

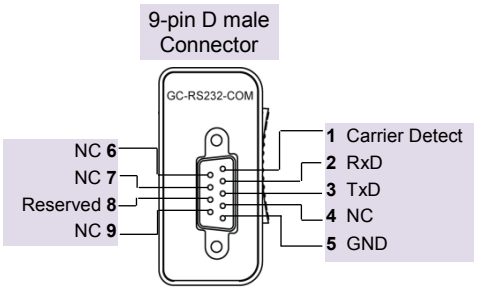


Figure 26: Connection details for GC-RS232-COM

Refer section *Wiring*, before wiring to COM extension unit.

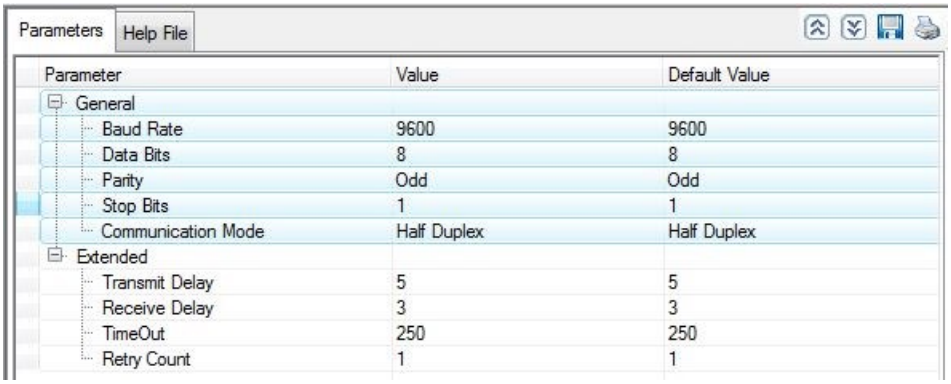
It is recommended to limit RS232 communication cable length to 10 meters maximum.

13.3 Configuration and Programming

For GOC COM Extension units, I/O memory map is fixed.

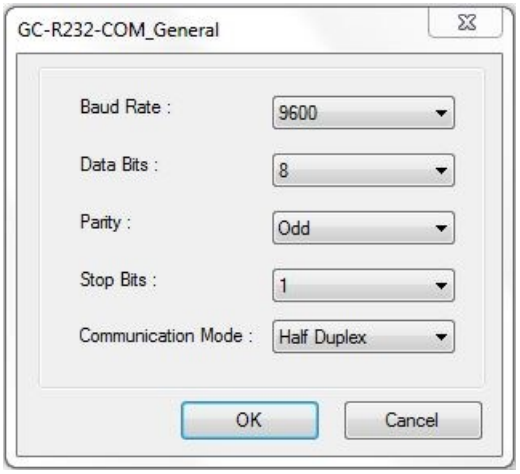
GC-RS232-COM consumes %IB4 and %QB4 when fixed in COM1 slot. It consumes %IB6 and %QB6 when fixed in COM2 slot.

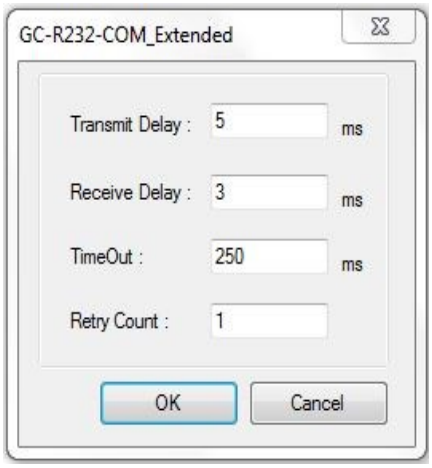
Integrated tool " Hardware Configuration <R> provides parameterization of serial port as shown below.



It provides default configuration and user can change it as per application requirement by selecting parameter and clicking on respective highlight.

The dialogue below shows General and Extended Parameters for RS232 serial com extension unit.





The table below provides the details of parameters required for RS232 serial com extension unit.

Parameters	Options	Description
General		
Baud Rate	9600, 19200, 38400, 57600, 115200	Baud rate in bps
Data Bits	8, 7	Number of data bits
Parity	Odd, None, even	Parity
Stop Bits	1, 2	Number of stop bits
Communication Mode	Half Duplex, Full Duplex	Communication mode as half duplex or full duplex.
Extended		
Transmit Delay	0 to 255 ms (5 ms default)	Hold value of delay in msec between transmission and last reception.
Receive Delay	0 to 255 ms (3 ms default)	Holds value of expected minimum inter frame delay in msec during reception. This helps in deciding start of a new frame especially for binary protocols like Modbus RTU. Any byte received after an interval greater than Rx_Delay is considered as a start of frame. FB ReceiveByte Ex returns data and status (like start of frame, frame error, parity error) of received byte.
TimeOut	0 to 60000 ms (250 ms default)	Timeout in reception of expected response. If expected numbers of bytes are not received before elapse of TimeOut period, then TimeOut Error is declared for built-in protocols like Modbus RTU Master and Master retries. For custom protocols, user can program suitable logic.
Retry Count	0 to 255 (1 default)	Number of retries to be executed in case of communication error. Applicable for Master FB.

Based on this configuration, "Hardware Configuration <R> generates a global variable list "ImplicitIOList <R> which provides declaration of all the I/O points for configured units as well as predefined variables as per the feature configured depending of type of unit configured.

Screen shot below shows default variable list for GC-RS232-COM unit fixed in COM1 slot.

```
(*                                    COM1 : GC-RS232-COM                                                    *)
(*************************************************************************************************************)
    _COM1Comset    : ComSet := (ComSet:=16#305,TX_Delay:=5,RX_Delay:=3,Time_Out:=T#250ms,RetryCount:=1);

    _COM1PortOpen      AT    %IX4.0    :    BOOL;
    _COM1TxBufEmpty    AT    %IX4.1    :    BOOL;
    _COM1TxBufOVF      AT    %IX4.2    :    BOOL;
    _COM1RxBufEmpty    AT    %IX4.4    :    BOOL;
    _COM1RxBufOVF      AT    %IX4.5    :    BOOL;
    _COM1CD            AT    %IX4.7    :    BOOL;

    _COM1Reset         AT    %QX4.0    :    BOOL;
    _COM1ClrTXBuf      AT    %QX4.1    :    BOOL;
    _COM1ClrRXBuf      AT    %QX4.2    :    BOOL;

(*************************************************************************************************************)
```

The table below provides the details of global variables related to GC-RS232-COM.

☐=1 for unit fixed in COM1 slot, ☐=2 for unit fixed in COM2 slot,

Global Variables	Data Type	Description
_COM☐Comset	ComSet	Defines communication parameters. For internal purpose only. Communication can be performed by configuring standard built-in protocols like Modbus RTU master or slave or by executing Serial Com FBs provided in GOC35 library.

The table below provides the details of I/O bits related to GC-RS232-COM.

☐=1 for unit fixed in COM1 slot, ☐=2 for unit fixed in COM2 slot,

I/O Variables	Address		Description
	IO1 Slot	IO2 Slot	
_COM☐PortOpen	%IX4.0	%IX6.0	**TRUE**: Port is opened with defined communication parameters. **FALSE**: Port is not opened.
_COM☐TxBufEmpty	%IX4.1	%IX6.1	**TRUE**: Transmission not on /Completed / Transmit buffer empty **FALSE**: Transmission on.
_COM☐TxBufOVF	%IX4.2	%IX6.2	**TRUE**: Transmit buffer overflow. Number of byte to be transmitted exceeding 255. **FALSE**: Transmit buffer not overflow.
_COM☐RxBufEmpty	%IX4.3	%IX6.3	**TRUE**: Receive buffer not empty. User can read received bytes from receive buffer by executing FB ReceiveByte/ ReceiveByteEx **FALSE**: Receive buffer empty
_COM☐RxBufOVF	%IX4.4	%IX6.4	**TRUE**: Receive buffer overflow. Number of byte to be not read from system buffer is exceeding 255 so extra bytes received are lost. Ensure that received bytes are read by executing FB ReceiveByte/ ReceiveByteEx at proper interval. **FALSE**: Receive buffer not overflow
_COM☐CD	%IX4.7	%IX6.7	Becomes TRUE if carrier detect signal from modem is ON.
_COM☐Reset	%QX4.0	%QX6.0	Reset and restart communication function on rising edge of output bit.
_COM☐ClrTxBuff	%QX4.1	%QX6.1	Clear transmit buffer on rising edge of output bit.
_COM☐ClrRxBuff	%QX4.2	%QX6.2	Clear receive buffer on rising edge of output bit.

14 COM Extension Unit GC-RS422-COM

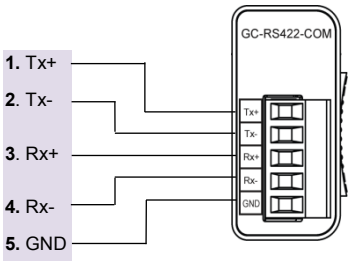
This is RS422/485 serial communication extension units. User can attach up to 2 COM extension units on the back side of Main unit.

14.1 Specifications

Item		Description
No. of serial ports		1
Hardware interface		RS422/ 485, depending upon external connections
Signals		Transmit+, Transmit-, Receive+, Receive-, Ground
		For RS485 interface, D+ → connect signals 'Transmit+' and 'Receive+' together on terminal block D - → connect signals 'Transmit+' and 'Receive+' together on terminal block User can connect terminating resistors externally as applicable.
Communication parameters	Baud rate (bps)	9600, 19200, 38400, 57600, 115200
	Data bits	7, 8
	Parity	Odd, Even, None
	Stop bits	1, 2
Communication type		Full duplex or half duplex
Connector type		5-pin removable terminal block
Isolation		No isolation from main circuit
Dimensions (in mm)		26.0 (W) x 51.0 (H) x 51.2 (D)
Weight (in grams)		40

14.2 Wiring

COM extension unit provides 5-pin removable terminal block on its front side. The figure below shows front view of COM extension unit with connection details.



For RS485 interface,

D+: Connect terminal 1 (Tx+) and terminal 3 (Rx+) together.

D- : Connect terminal 2 (Tx-) and terminal 4 (Rx-) together.

Terminating resistor is not provided on board. Hence, whenever required, connect termination resistor on terminal block externally.

Figure 27: Connection details of GC-RS422-COM

Refer section *Wiring*, before wiring to COM extension unit.

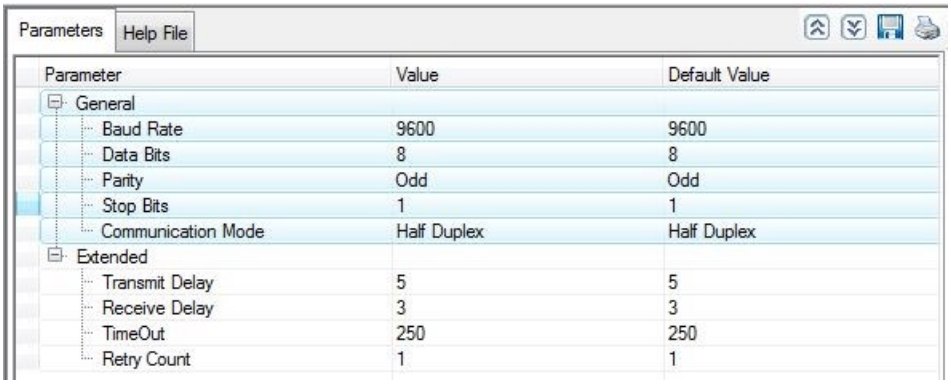
As RS422/485 communication signals are low level signals, it is recommended to limit communication cable length to 25 meters maximum. Install protection devices externally if cable length is greater than 25 meters for safety.

14.3 Configuration and Programming

For GOC COM Extension units, I/O memory map is fixed.

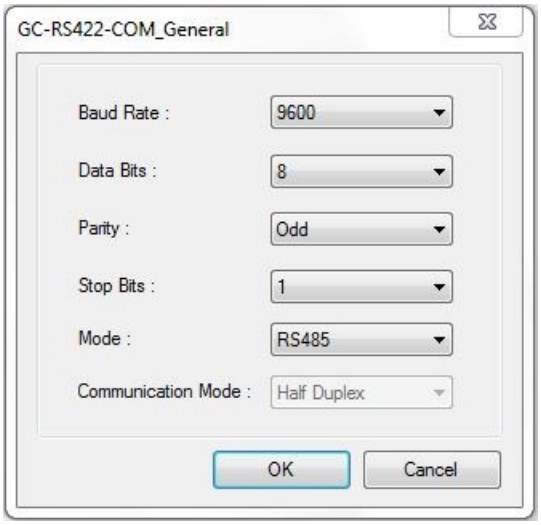
GC-RS422-COM consumes %IB4 and %QB4 when fixed in COM1 slot. It consumes %IB6 and %QB6 when fixed in COM2 slot.

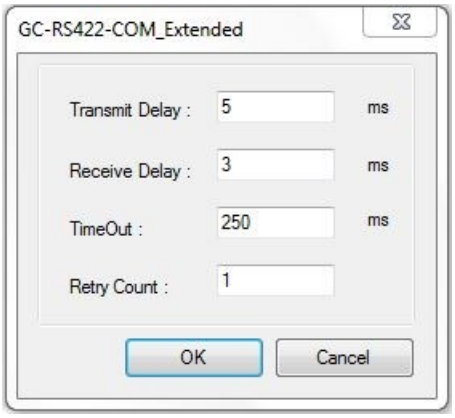
Integrated tool " Hardware Configuration <R> provides parameterization of serial port as shown below.



It provides default configuration and user can change it as per application requirement by selecting parameter and clicking on respective highlight.

The dialogue below shows General and Extended Parameters for RS422 serial com extension unit.





The table below provides the details of parameters required for RS422 serial com extension unit.

Parameters	Options	Description
General		
Baud Rate	**9600**, 19200, 38400, 57600, 115200	Baud rate in bps
Data Bits	**8**, 7	Number of data bits
Parity	**Odd**, None, even	Parity
Stop Bits	**1**, 2	Number of stop bits
Mode	**RS485**, RS422	Physical interface RS485 or RS422
Communication Mode	**Half Duplex**, Full Duplex	Communication mode as half duplex or full duplex (not applicable for RS485).
Extended		
Transmit Delay	0 to 255 ms (5 ms default)	Hold value of delay in msec between transmission and last reception.
Receive Delay	0 to 255 ms (3 ms default)	Holds value of expected minimum inter frame delay in msec during reception. This helps in deciding start of a new frame especially for binary protocols like Modbus RTU. Any byte received after an interval greater than Rx_Delay is considered as a start of frame. FB ReceiveByte Ex returns data and status (like start of frame, frame error, parity error) of received byte.
TimeOut	0 to 60000 ms (250 ms default)	Timeout in reception of expected response. If expected numbers of bytes are not received before elapse of TimeOut period, then TimeOut Error is declared for built-in protocols like Modbus RTU Master and Master retries. For custom protocols, user can program suitable logic.
Retry Count	0 to 255 (1 default)	Number of retries to be executed in case of communication error. Applicable for Master FB.

Based on this configuration, Hardware Configuration <R> generates a global variable list ImplicitIOList <R> which provides declaration of all the I/O points for configured units as well as predefined variables as per the feature configured depending of type of unit configured.

Screen shot below shows default variable list for GC-RS422-COM unit fixed in COM1 slot.

```
(*                              COM1 : GC-RS422-COM                              *)
(*******************************************************************************)
   _COM1Comset    : ComSet := (ComSet:=16#305,TX_Delay:=5,RX_Delay:=3,Time_Out:=T#250ms,RetryCount:=1);

   _COM1PortOpen     AT    %IX4.0   :     BOOL;
   _COM1TxBufEmpty   AT    %IX4.1   :     BOOL;
   _COM1TxBufOVF     AT    %IX4.2   :     BOOL;
   _COM1RxBufEmpty   AT    %IX4.4   :     BOOL;
   _COM1RxBufOVF     AT    %IX4.5   :     BOOL;

   _COM1Reset        AT    %QX4.0   :     BOOL;
   _COM1ClrTXBuf     AT    %QX4.1   :     BOOL;
   _COM1ClrRXBuf     AT    %QX4.2   :     BOOL;

(*******************************************************************************)
```

The table below provides the details of global variables related to GC-RS422-COM.

☐=1 for unit fixed in COM1 slot, ☐=2 for unit fixed in COM2 slot,

Global Variables	Data Type	Description
_COM☐Comset	ComSet	Defines communication parameters. For internal purpose only. Communication can be performed by executing Serial Com FBs provided in GOC35 library.

The table below provides the details of I/O bits related to GC-RS422-COM.

☐=1 for unit fixed in COM1 slot, ☐=2 for unit fixed in COM2 slot,

I/O Variables	Address		Description
	IO1 Slot	IO2 Slot	
_COM☐PortOpen	%IX4.0	%IX6.0	**TRUE**: Port is opened with defined communication parameters. **FALSE**: Port is not opened.
_COM☐TxBufEmpty	%IX4.1	%IX6.1	**TRUE**: Transmission not on /Completed / Transmit buffer empty **FALSE**: Transmission on.
_COM☐TxBufOVF	%IX4.2	%IX6.2	**TRUE**: Transmit buffer overflow. Number of byte to be transmitted exceeding 255. **FALSE**: Transmit buffer not overflow.
_COM☐RxBufEmpty	%IX4.4	%IX6.3	**TRUE**: Receive buffer not empty. User can read received bytes from receive buffer by executing FB ReceiveByte/ ReceiveByteEx **FALSE**: Receive buffer empty
_COM☐RxBufOVF	%IX4.5	%IX6.4	**TRUE**: Receive buffer overflow. Number of byte to be not read from system buffer is exceeding 255 so extra bytes received are lost. Ensure that received bytes are read by executing FB ReceiveByte/ ReceiveByteEx at proper interval. **FALSE**: Receive buffer not overflow
_COM☐Reset	%QX4.0	%QX6.0	Becomes TRUE if carrier detect signal from modem is ON.
_COM☐ClrTxBuff	%QX4.1	%QX6.1	Reset and restart communication function on rising edge of output bit.
_COM☐ClrRxBuff	%QX4.2	%QX6.2	Clear transmit buffer on rising edge of output bit.

15 COM Extension Unit GC-ENET-COM

This is Ethernet communication extension units. User can attach 1 unit in COM1 slot for programming purpose (i.e. CoDeSys) or Modbus TCP communication and CC-Link IEF Basic interface (Master or slave).

Modbus TCP server or CC-Link IEF Basic (master or slave) protocol can be configured by programming function blocks.

15.1 Specifications

15.1.1 Specifications: GC-ENET-COM

Item	Description
Number of Ethernet ports	1*
Physical Level	10/100 Base-TX
Connector type	RJ45 female, shielded
Auto crossover	Yes
Cable type	Category 5e or higher STP (Shielded Twisted Pair) or UTP (Unshielded Twisted Pair),
Max. cable distance	100 meters
Application level	1. Controller programming protocol or Modbus TCP server by programming FB MBUSTCP_Server in application program.
	2. CC-Link IEF Basic master by programming FB CCLIEFBasic_Master_Config in application program or CC-Link IEF Basic slave by programming FB CCLIEFBasic_Slave_Config in application program
No. of simultaneous connections	2
Diagnostics	Green and Yellow LEDs (On RJ45 connector)
Isolation	1500 Vac / 1 minute
Dimensions (in mm)	26.0 (W) x 51.0 (H) x 42.2 (D)
Weight (in grams)	40

*GC-ENET-COM extension unit can be fixed in COM1 slot only.

15.1.2 Specifications: Modbus TCP Server

Item	Description	
No. of simultaneous connections	1 (either CoDesys or single Modbus TCP client at a time)	
Buffer memory	260 bytes transmit buffer and 260 bytes receive buffer	
Modbus commands supported	Function Code	Meaning
	01	Read Coil Status
	05	Force Single Coil
	15	Force Multiple Coils
	02	Read Input Status
	04	Read Input registers
	03	Read Holding Registers
	06	Preset Single Register
	16	Preset Multiple Registers
	23	Read/Write Multiple Registers

Modbus TCP server protocol can be configured using function blocks provided in GOCUtil library.

Following function block is supported to configure GOC as a Modbus TCP server.

1. FB MBUSTCP_Server

User can either connect programming software CoDeSys or Modbus TCP client at a time.

Refer “ED-2002-541_GOCUtil_LibraryManual.pdf” for more details of usage.

15.1.3 Specifications: CC-Link IEF Basic Master

Item	Description			
Station type	Master station			
Number of stations	4 maximum. Ensure that total number of occupied stations does not exceed 4. e.g. if user connects 1 slave device then it can occupy up to 4 stations. If user connects 4 slave devices, each can occupy only 1 station.			
IO Data Size	Depends on number of stations occupied			
	No. of occupied stations	**RX**	**RY**	**RWr**
	1	64 bit	64 bit	32 word
	2	128 bit	128 bit	64 word
	3	192 bit	192 bit	96 word
	4	256 bit	256 bit	128 word
Communication protocol	UDP			
Port number	No. 61450 (Cyclic data)			

CC-Link IEF Basic master protocol can be configured using function blocks provided in GOC35 library.

Following function blocks are supported to configure GOC as a master station.

1. FB CCLIEFBasic_Master_Config
2. FB CCLIEFBasic_Master_Diag
3. FB CCLIEFBasic_Slave_Info

User can configure GOC as CC-Link IEF Basic master or slave at a time.

Refer “ED-2002-540_GOC35_LibraryManual.pdf” for more details of usage.

15.1.4 Specifications: CC-Link IEF Basic Slave

Item	Description			
Station type	Slave station			
Number of occupied stations	1-4 stations (user configurable)			
IO Data Size	Depends on number of stations occupied			
	No. of occupied stations	**RX**	**RY**	**RWr**
	1	64 bit	64 bit	32 word
	2	128 bit	128 bit	64 word
	3	192 bit	192 bit	96 word
	4	256 bit	256 bit	128 word
Communication protocol	UDP			
Port number	No. 61450 (Cyclic data)			
	No. 61451 (Node Search and IP Address Set dedicated for CC-Link IE Field Network Basic)			

CC-Link IEF Basic functionality can be configured using function blocks provided in GOC35 library.

Following function blocks are supported to configure GOC as a slave station.

1. FB CCLIEFBasic_Slave_Config
2. FB CCLIEFBasic_Slave_Diag

User can configure GOC as CC-Link IEF Basic master or slave at a time.

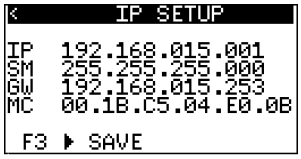
Refer “ED-2002-540_GOC35_LibraryManual.pdf” for more details of usage.

GxWorks3 profile file “0x2071_GC35MH_0x0001_en.CSPP.zip” can be registered in GxWorks3 so that GOC35 will appear as a Slave Station in Module List of “CC-Link IEF Basic Configuration”.

15.2 IP address

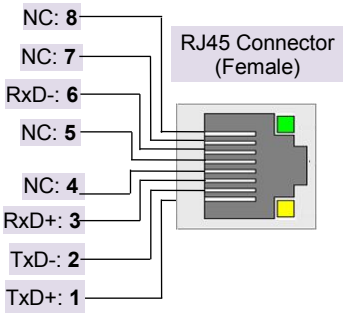
Ethernet port is assigned with default IP address so that the use of Ethernet port is possible. The table below shows network parameters those are factory settings. These parameters can be changed using system menu screen.

IP Address	192.168.15.1
Subnet Mask	255.255.255.0



15.3 Wiring

COM extension unit provides RJ45 female connector on its front side. The figure below shows front view of COM extension unit with connection details.



LED Indications:

Green	Yellow	Significance
○	○	No LINK network.
●	○	LINK network 10 Mbps.
●	●	LINK network 100 Mbps.
–	X	Flashing indicates transmission and reception

Symbols:

○ : Indicates LED OFF
● : Indicates LED ON
– : Indicates LED flashing
X : Ignore LED status

Figure 28: Connection details for GC-ENET-COM

Refer section *Wiring*, before wiring to COM extension unit.

15.4 Configuration and Programming

GC-ENET-COM extension unit is fixed in COM1 slot. Parameters like IP address, subnet mask and gateway address can be set through system menu from LCD display on the controller itself. Refer section *IP Set Up* to know more about IP Settings.

There is no additional configuration and parameterization required using *Hardware Configuration Tool*.

Refer “ED-2002-541_GOCUtil_LibraryManual.pdf” for more details of Modbus TCP server function block.

Refer “ED-2002-540_GOC35_LibraryManual.pdf” for more details of CC-Link IEF Basic function block.

If Ethernet cable is removed while CoDeSys is logged-in to the controller, CoDeSys provides message as Communication Error (#0): Log Out performed. If connection is restored immediately, CoDeSys takes 25 sec to Login again.

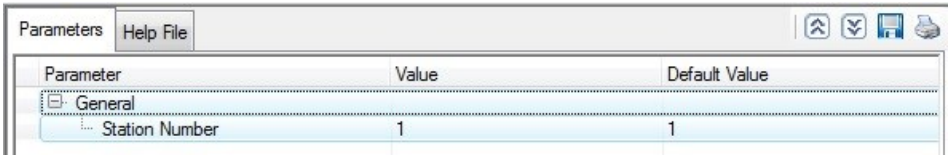
It is recommended to keep Communications timeout (Project → Options → Desktop → Communications timeout) in CoDeSys **> 4 secs** to download/ upload source code through Ethernet COM extension unit via LAN. Timeout may need to be increased depending on data traffic in the LAN or internet.
Default time out of 2.2 sec works fine if PC is connected to the controller directly.

16 Modbus RTU Configuration

User can configure Modbus RTU slave or master protocol for serial com extension unit (GC-RS232-COM, GC-RS422-COM) fixed in any COM (COM1, COM2) slot.

16.1 Modbus RTU Slave

User can configure serial communication port as Modbus RTU slave. Integrated tool "Hardware Configuration <R>" provides parameterization of protocol as shown below.



For Modbus RTU slave functionality, user has to assign a station number. It provides default station number as 1 and user can change it as per application requirement by selecting parameter and clicking on respective highlight.

Dialogue box allow to select Station Number from 1 to 247.

Based on this configuration, "Hardware Configuration <R>" generates a Global Variable List "ImplicitIOList <R>" which provides declaration of all the I/O points for configured units as well as predefined variables as per the feature configured depending of type of unit configured.

Screen shot below shows default variable list for Modbus RTU Slave protocol configured for unit fixed in COM2 slot.

```
(*                          COM2 Protocol : MODBUS RTU SLAVE                                                    *)
(*************************************************************************************************************)
   _COM2ModbusStatus      :      BYTE;
   _COM2ModbusReplies     :      WORD;
   _COM2ModbusErrCount    :      WORD;
```

The table below provides the details of global variables related to Modbus RTU slave protocol.

☐=1 for unit fixed in COM1 slot, ☐=2 for unit fixed in COM2 slot,

<table>
<tr><th>Global Variables</th><th>Data Type</th><th colspan="2">Description</th></tr>
<tr><td rowspan="8">_COM☐ModbusStatus</td><td rowspan="8">BYTE</td><td colspan="2">Returns Status of Modbus RTU slave communication as below</td></tr>
<tr><td>Bit</td><td>Significance</td></tr>
<tr><td>0</td><td>Becomes TRUE if port is opened successfully with valid station number</td></tr>
<tr><td>1</td><td>Becomes TRUE in case of checksum error in any previous query.</td></tr>
<tr><td>2</td><td>Becomes TRUE if timeout is observed in reception of any previous query as well as transmission of any previous response due to any reason</td></tr>
<tr><td>6</td><td>Become TRUE, in case of parity error in received byte</td></tr>
<tr><td>7</td><td>Become TRUE, in case of frame error in received byte</td></tr>
<tr><td colspan="2">Error status is reset only on power recycle. Otherwise it continues to hold last status.</td></tr>
<tr><td>_COM☐ModbusReplies</td><td>WORD</td><td colspan="2">Holds number of successful replies</td></tr>
<tr><td>_COM☐ModbusErrCount</td><td>WORD</td><td colspan="2">Increments by 1 if
- Timeout in reception or transmission
- Checksum mismatch
- Any intermittent byte received is start of frame
- Any byte received has frame/parity error
- Received query contains address (holding register, coil, etc) beyond supported range of slave PLC. (No exception is generated)</td></tr>
</table>

16.2 Modbus RTU Master

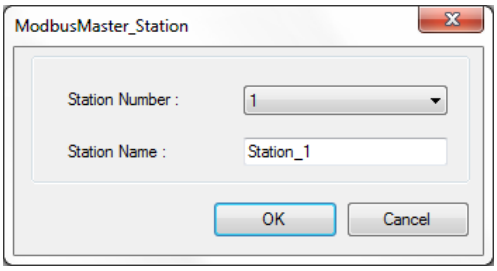
User can configure serial communication port as Modbus RTU slave. Integrated tool Hardware Configuration <R> provides parameterization of protocol as explained below.

There are 3 steps in Modbus RTU Master protocol configuration

Step 1: Add slave devices
Step 2: Prepare scan record list
Step 3: Set record scan mode.

16.2.1 Add slave devices

Click ‘Slave Devices’ tab from configuration pane and click icon in upper left corner. Add Station dialogue opened as



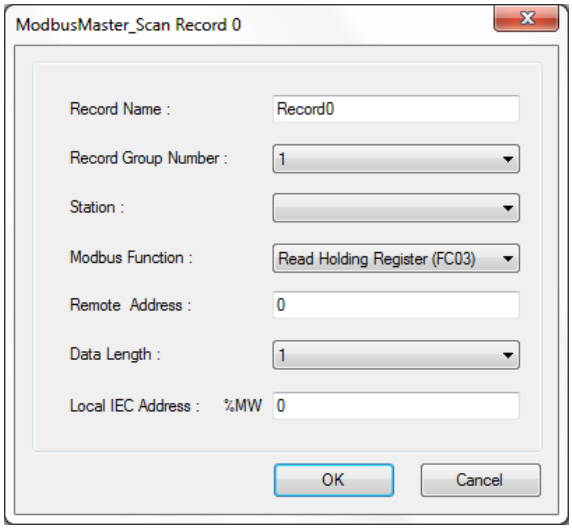
Set station number and station name and click Ok button.
Use same procedure to add other slave devices in network.

16.2.2 Prepare scan record list

‘Scan record’ defines parameters to generate a communication query to a slave device. All scan records are arranged in 'scan record list'.

In runtime GOC generates respective communication queries for slave devices based on parameters of scan records from list.

To add scan record to list click ‘Scan Record List’ tab from configuration window. Click button in upper left corner of configuration window.



The table below provides the details of configuration required for Modbus scan record.

Parameters	Options	Description
Record Name	--	Name of the record. The name being set can be changed to meet the purpose of use. e.g. If record is being used to read frequency of inverter then name can be changed as 'ReadInvFreq'
Record Group Number	0-247	Helps in grouping queries. Read 'Set Record Scan Mode' mentioned on this page for more details.
Station	--	Holds target slave device. Select target device from drop down list.
Modbus Function	Read Coil Status (FC01) Read Input Status (FC02) Read Holding Register FC03) Read Input Register (FC04) Force Single Coil (FC05) Preset Single Register FC06) Force Multiple Coils (FC15) Preset Multiple Registers (FC16)	Defines function code as per Modbus RTU protocol.
Remote Address	0 - 65535	Defines memory address in target Modbus RTU slave device. Remote address is defined as offset from start address. e.g. If function 'Read Holding Register (FC03)' is used to read register from target device with remote address as 10 that means communication query is generated to read Modbus address 40000+10. Where 40000 is starting address of holding registers in target Modbus slave device. Starting address for coils is 00000. Starting address for input registers is 30000 and starting address for input bits is 10000
Data Length	Data Length range varies with function selection Read Coil Status (FC01) : 1-2000 Read Input Status (FC02) : 1-2000 Read Holding Register (FC03):1-125 Read Input Register (FC04) : 1-125 Force Single Coil (FC05) : NA Preset Single Register (FC06) : NA Force Multiple Coils (FC15) : 1-2000 Preset Multiple Registers (FC16) : 1-123	Holds number of addresses (registers /bits) to read / write.
Local IEC Address	For boolean type, %MX0.0 - %MX3999.7 For register type, %MW0 - %MW3998	Holds start address of master controller memory Where read data from slave is stored (response of read queries) Where write data for slave is stored (command to write queries)

Click  button to delete the record from list.

16.2.3 Set record scan mode

Sequence of sending communication queries from 'Scan Record List' is controlled with ‘Record scan mode’.

There are 3 record scan modes available

- Manual
  In this mode selected record from record table is executed on request.
  Manual mode sends communication query defined by global variable '`_COM#ModbusManualRecNo`' once on rising edge of variable '`_COM#ModbusManualTrigger`.
  These variables are available in 'ImplicitIOList'.
- Cyclic
  This is default mode of sending queries.
  Cyclic mode sends queries with Group 1 one by one cyclically.
  Note that queries only with record group = 1 are executed. Other queries are ignored.
- Cyclic Group
  Cyclic group mode sends queries with Group =1 and Group = variable '`_COM#ModbusGrSel`' one by one cyclically.
  For example if '`_COM#ModbusGrSel`' = 4 then queries with record group 1 and 4 are executed.
  Variable '`_COM#ModbusGrSel`' is available as functionality access variable in ImplicitIOList'.

Based on this configuration, " Hardware Configuration <R> generates a Global Variable List " ImplicitIOList <R> which provides declaration of predefined variables as per the feature configured

Screen shot below shows default variable list for Modbus RTU Master protocol configured for unit fixed in COM1 slot.

```
(*****************************************************************************************)
(*                          COM1 Protocol : MODBUS RTU MASTER                            *)
(*****************************************************************************************)
    _COM1ModbusStatus                :      BYTE;
    _COM1ModbusLastRec               :      INT;
    _COM1ModbusLastResult            :      BOOL;
    _COM1RecTable : ARRAY[0..2] OF RTU_RecStruct :=
        (Gr:=1,FNC:=3,SNO:=1,LDT:=M,LAdr:=100,Radr:=0,Dlen:=1),
        (Gr:=1,FNC:=16,SNO:=2,LDT:=M,LAdr:=200,Radr:=0,Dlen:=1);
```

The table below provides the details of global variables related to Modbus RTU master protocol.

☐=1 for unit fixed in COM1 slot, ☐=2 for unit fixed in COM2 slot,

<table>
<tr><th>Global Variables</th><th>Data Type</th><th colspan="2">Description</th></tr>
<tr><td rowspan="8">_COM☐ModbusStatus</td><td rowspan="8">BYTE</td><td colspan="2">Returns Status of Modbus RTU master communication as below</td></tr>
<tr><td>Bit</td><td>Significance</td></tr>
<tr><td>0</td><td>Recent record executed successfully</td></tr>
<tr><td>1</td><td>Checksum error in reception</td></tr>
<tr><td>2</td><td>Timeout Error</td></tr>
<tr><td>3</td><td>Unknown function in record table</td></tr>
<tr><td>4</td><td>End of record table reached</td></tr>
<tr><td colspan="2">Error status is reset only on power recycle. Otherwise it continues to hold last status.</td></tr>
<tr><td>_COM#ModbusLastRec</td><td>INT</td><td colspan="2">Holds record number recently executed</td></tr>
<tr><td>_COM#ModbusLastResult</td><td>BOOL</td><td colspan="2">Holds result of execution of last record. TRUE: Record executed successfully. FALSE: Record not executed successfully. Error observed while executing last record.</td></tr>
<tr><td>_COM#ModbusManualRecNo</td><td>INT</td><td colspan="2">Applicable only if 'Record Scan Mode' = Manual Sets record number to be sent in manual mode at rising edge of Boolean variable _COM#ModbusManualTrigger.</td></tr>
<tr><td>_COM#ModbusManual Trigger</td><td>BOOL</td><td colspan="2">Applicable only if 'Record Scan Mode' = Manual In manual record scan mode i.e. At rising edge of _COM#ModbusManualTrigger, GOC sends query defined by _COM#ModbusManualRecNo</td></tr>
<tr><td>_COM#ModbusGrSel</td><td>BYTE</td><td colspan="2">Applicable only if 'Record Scan Mode' = Group Cyclic Selects the sequence of sending queries defined by _COM#RecTable. In group cyclic record scan mode, GOC sends queries with Gr=1 and Gr=_COM#ModbusGrSel one by one cyclically.</td></tr>
<tr><td>_COM#RecTable</td><td>Array [0..n] of RTU_RecStruct</td><td colspan="2">It is a record table which holds communication records that define Modbus RTU queries. Record table is an array of structure of type RTU_RecStruct. Members of the record are as shown below. Few members of the record are set by hardware configuration tool based on scan record configuration done by user while few members are updated by GOC during runtime while executing serial communication in run time. The members updated by GOC during run time can be used by user (programmer) for diagnosis of communication errors during run time.</td></tr>
</table>

RTU_RecStruc Structure

Member	Data Type	Description
GR	BYTE	Defines group of record. Set by Hardware Configuration Tool as per scan record configuration.
FNC	BYTE	Defines function code as per Modbus RTU protocol Set by Hardware Configuration Tool as per scan record configuration.
SNO	BYTE	Defines target station number. Set by Hardware Configuration Tool as per scan record configuration.
LDT	RTU_TT	For internal use. Set by Hardware Configuration Tool.
LAdr	WORD	Holds start address of Master controller memory (I/Q/M defined by LDT) - where read data (from slave) is stored (response of FNC 1,2,3,4) where write data (to slave) is stored. (for query with FNC 5,6, 15, 16) Set by Hardware Configuration Tool as per scan record configuration
RAdr	WORD	Holds memory address (holding register, coil, etc) in target Modbus RTU slave device. Set by Hardware Configuration Tool as per scan record configuration.
DLen	BYTE	Holds number of addresses to read or write. Start address is defined by RAdr / LAdr. Set by Hardware Configuration Tool as per scan record configuration.
Result	BOOL	Hold last result of execution for the record. TRUE : Record executed successfully. FALSE : Error observed while executing record. Set by controller after execution of record in runtime. Can be used for diagnosis purpose by user.
Queries	DWORD	Holds number of times the record is executed to generate communication query. Updated by controller during execution of record in runtime. Can be used for diagnosis purpose by user.
Replies	DWORD	Holds number of times the record is executed successfully. Updated by controller during execution of record in runtime. Can be used for diagnosis purpose by user.
CHKSUM_Err	DWORD	Holds number of times the checksum error is observed while executing this record. Updated by controller during execution of record in runtime. Can be used for diagnosis purpose by user.
TimeOut_Err	DWORD	Holds number of times the timeout error is observed while executing this record. Updated by controller during execution of record in runtime. Can be used for diagnosis purpose by user.

In CoDeSyS application project members of '_COM#RecTable' can be accessed as '_COM#RecTable.Result', '_COM#RecTable.Queries', ''_COM#RecTable.Replies' etc

17 Status and Diagnostics

Main unit provides 2 LED indications and LCD display on front panel to provide status and diagnostic information useful for troubleshooting. GOC provides system menu screen that provides more information.

17.1 LED Indications

Main unit provides 2 LED indications on front panel. The table below explains the significance of CPU diagnostics related LEDs

Status	Power	Run	Relevant System Variables
	Red	Green	
OFF	No power	User stop, Firmware update New firmware download, System error *1	_SysvarCPU.WSTATUS _SysvarCPU.BCPUSTOPCAUSE
ON	Power ON	Run mode	_SysvarCPU.WSTATUS _SysvarCPU.BINITSTATUS
Blinking 1x	Not applicable	IO Error	_SysvarCPU.WSTATUS
Blinking 2x		Forcing is active	_SysvarCPU.W_REG_STATUS _SysvarCPU.W_IOERR _SysvarCPU.AMODULEORDERINGCODE
Blinking 3x		Scan error	_SysvarCPU.WSTATUS _SysvarCPU.BCPUSTOPCAUSE
Flickering		Memory Error Application Download in progress iKey Error HMI key error	_SysvarCPU.WSTATUS _SysvarCPU.BCPUSTOPCAUSE

*1 When CPU is in STOP mode, LCD screen shows system menu SYSTEM INFO. For more details, refer section *System Info*

CPU goes in Stop mode permanently, when input supply falls below 18 VDC but above 10 VDC. It continues in Stop mode even though input supply is recovered above 18 VDC for safety purpose. Blinking ^ LED associated with **Home** key, indicates this condition. To recover the system, it is necessary to power cycle the Main unit.

Refer section *System Variables*, for more details of system variables.

17.2 System Menu

System Menu screens are predefined screens useful to monitor system status and diagnostics. It also allows user to modify system settings.

System menu structure is as below

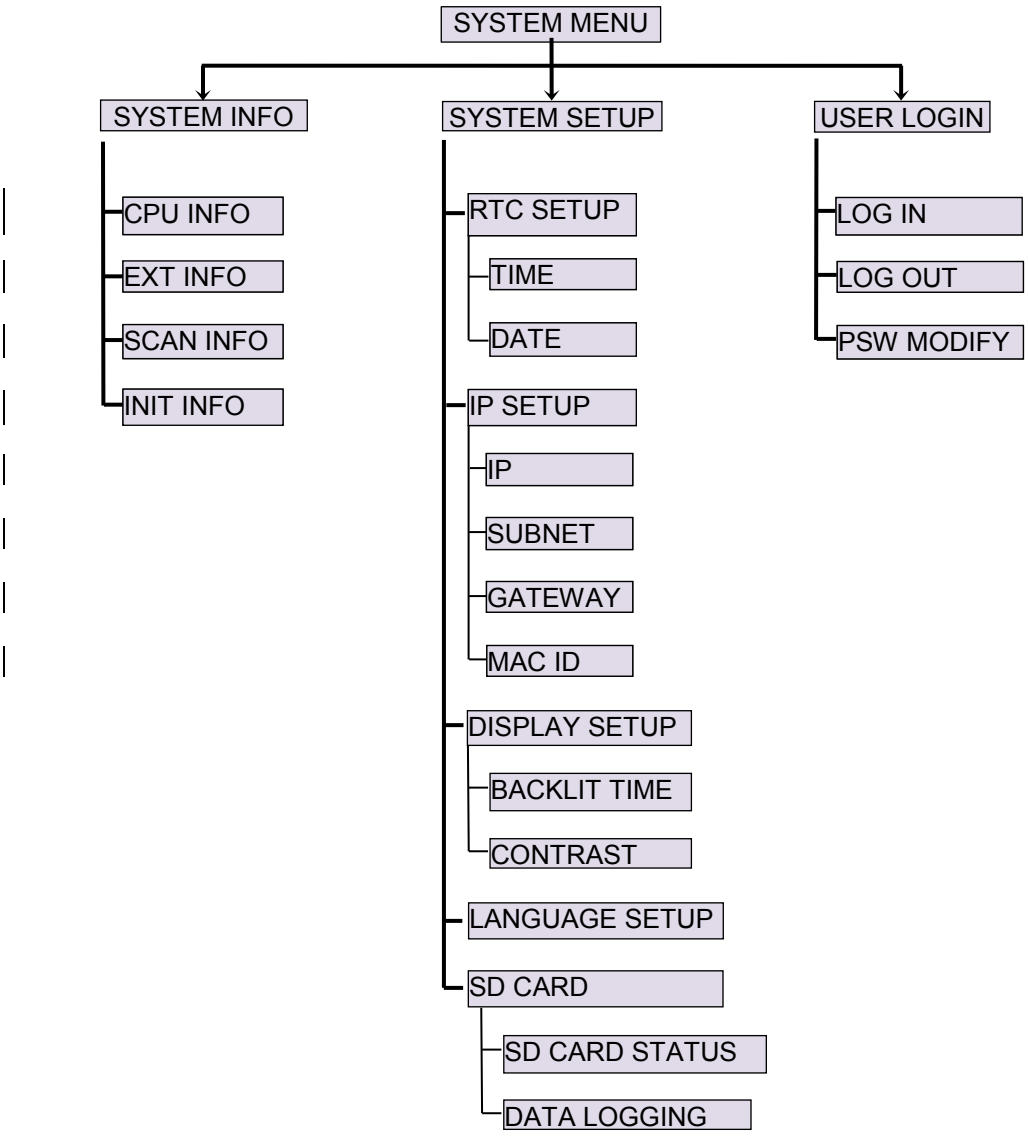


Figure 29: System menu structure

From any user screen, press and hold for 3 sec time duration, display will show SYSTEM MENU screen.

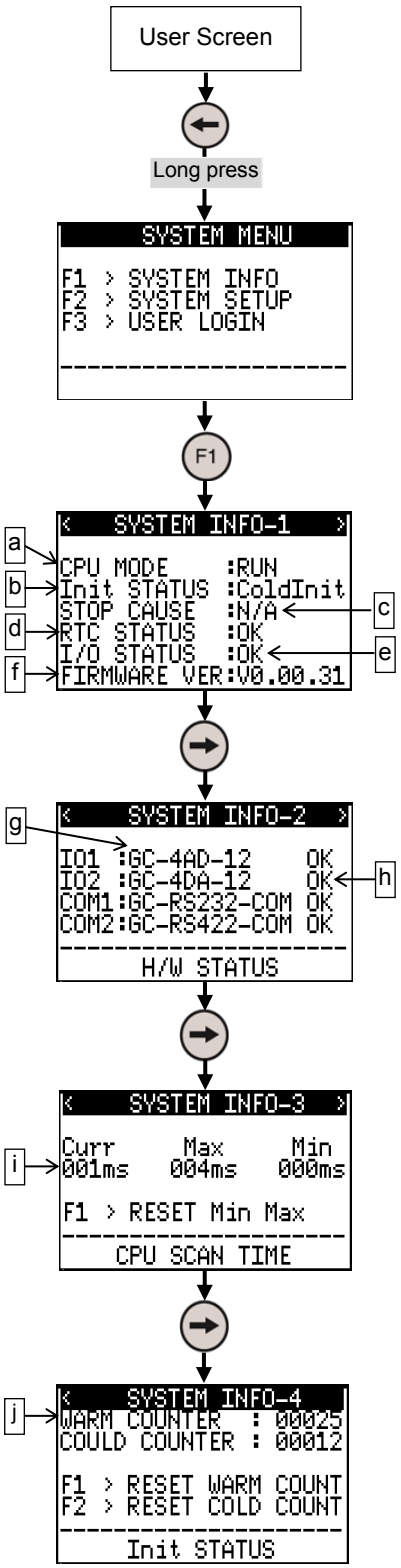
System Menu is available even when CPU is in stop mode due to any reason. During source code download / upload, LCD screen is not updated.

Note:

In menu screen, symbol < located at top left corner, indicates that user can go to previous menu by pressing **Back** key. Symbol > located at top right corner, indicates that user can go to next menu by pressing **Next** key.

17.2.1 System Info

This menu shows CPU and system status, useful for diagnostics and troubleshooting.



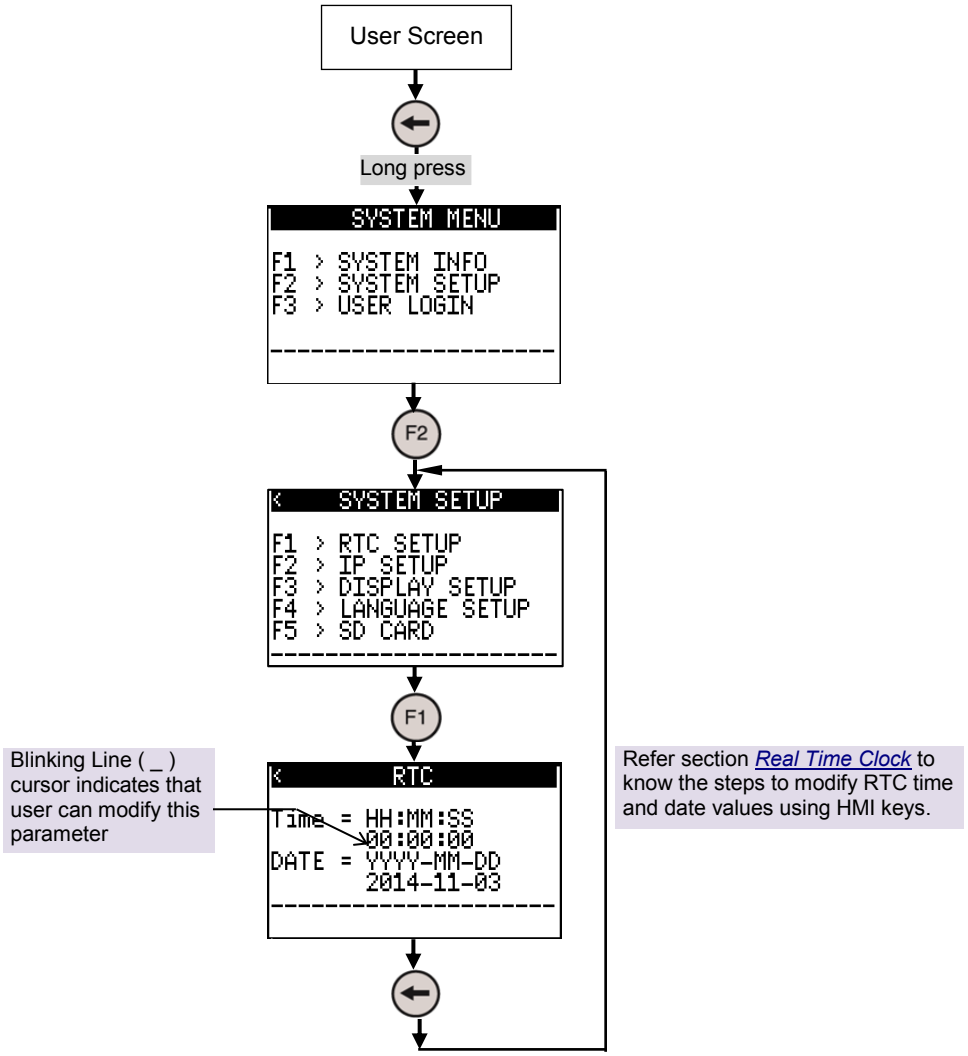
a	Shows CPU mode as RUN or STOP. In RUN mode, CPU executes application program.
b	**CPU Init status** Displays CPU initialization status when CPU is in RUN mode and displays cause of STOP if CPU is in STOP mode HotInit- If power fail has occurred for < 10 ms duration. CPU continues to function normally as if there is no power fail WarmInit- If power fail has occurred for > 10 ms duration or if CoDeSys menu command Online → Reset is issued. Non retained data is reset to 0 or user defined initial value. Retained data holds last value before power OFF. This is healthy initialization. ColdInit- This is faulty initialization or if CoDeSys menu command Online → ResetCold is issued or if program is downloaded. All the data is reset to 0 or user defined initial value. Cause may be hardware fault or external EMI issue.
c	**CPU STOP cause** User- If programmer puts CPU is STOP mode intentionally through programming software. Mem Err – If application program code is invalid Scan Err – If CPU is in STOP mode due to scan time exceeds its set limit (by default it is 250ms) OddAdrErr – If odd address is passed to data type other than BOOL or BYTE It is necessary to download valid and corrected application program again in case of Mem Err, Scan Err and OddAdrErr
d	**Real Time Clock status** OK if RTC data is valid. Err if RTC data is not valid due to loss of super capacitor backup due to any reason
e	**Extension unit status** Shows the status of extension units attached to Main unit as OK or Err.
f	Shows CPU firmware version
g	Shows ordering code of extension unit once detected by CPU at controller power on. CPU does not detect extension unit attached after power on. It displays N/A if expansion is never detected.
h	Shows status of individual expansion unit. Shows OK if CPU detects expansion unit once and is functioning thereafter. For more details, refer Shows ERR if CPU detects expansion unit once and is not communicating afterwards. Shows N\A if expansion is never detected.
i	Shows CPU scan time in msec. Curr shows current scan time. Max shows maximum scan time after power ON Min shows minimum scan time after power ON User can reset Min and Max values by pressing **F1** key.
j	Shows number of occurrences of warm initializations and cold initialization until now. User can reset warm/ cold initialization counter by pressing **F1**/ **F2** key.

17.2.2 System Setup

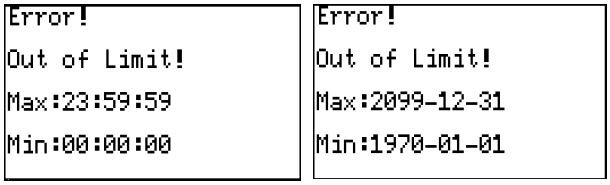
This menu allows user to set some of the system parameters like RTC, IP address for Ethernet communication and LCD display parameters.

17.2.2.1 RTC Setup

User can monitor and modify built-in RTC date and time. It also shows error in case of invalid RTC due to loss of super capacitor backup or any hardware fault.

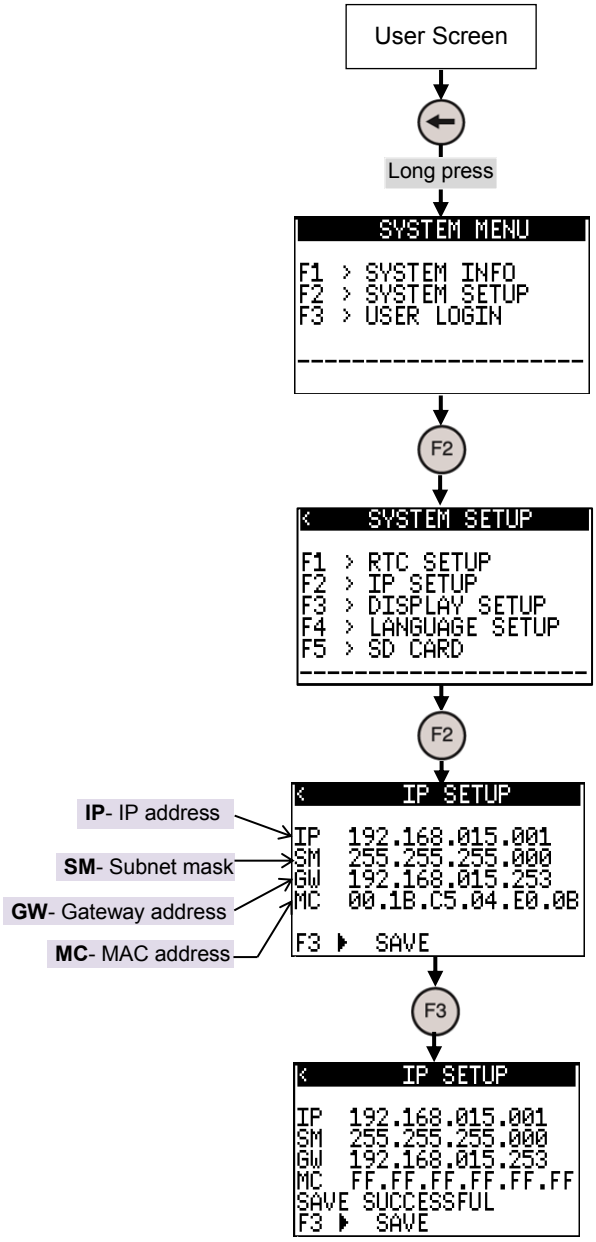


Refer section *Real Time Clock* to know the steps to modify RTC time and date values using HMI keys. When time and date entered is out of range, screen shows Error message momentarily as below.



17.2.2.2 IP Set Up

User can monitor and modify IP address, subnet mask and gateway address if Ethernet COM extension unit is attached at COM1 slot. User can monitor MAC address also. These settings are stored in Ethernet COM extension unit itself.



Refer section *IP Set Up* to know the steps to modify IP parameter values using HMI keys.

Modify settings as required and press F3 key to store settings to Ethernet COM extension unit.

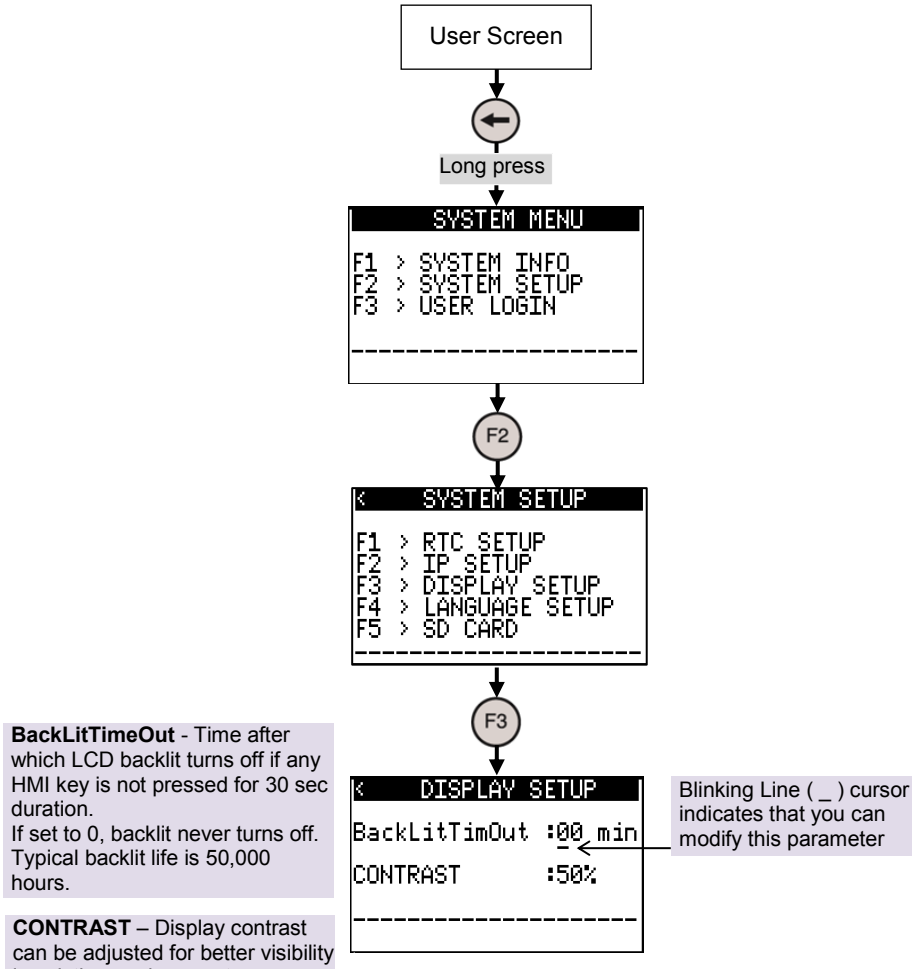
The table below shows error messages displayed during IP setting.

Error Message	Description
MODULE ABSENT	When Ethernet COM extension unit (GC-ENET-COM) is absent.
INVALID IP	IP address is invalid in case if - IP address does not belongs to class A/ B/C. - All host address bits are 0/1.

INVALID SUBNET	Subnet mask is invalid in case if - address is not between valid range 192.0.0.0 to 255.255.255.252 - address is not with contiguous ones from left hand side, in binary format.
INVALID GATEWAY	Gateway address is invalid in case if - Gateway address does not belongs to class A/ B/C. - All host address bits are 0/1.

17.2.2.3 Display Setup

User can set LCD display properties like backlit timeout and display contrasts. Modified settings are stored permanently.

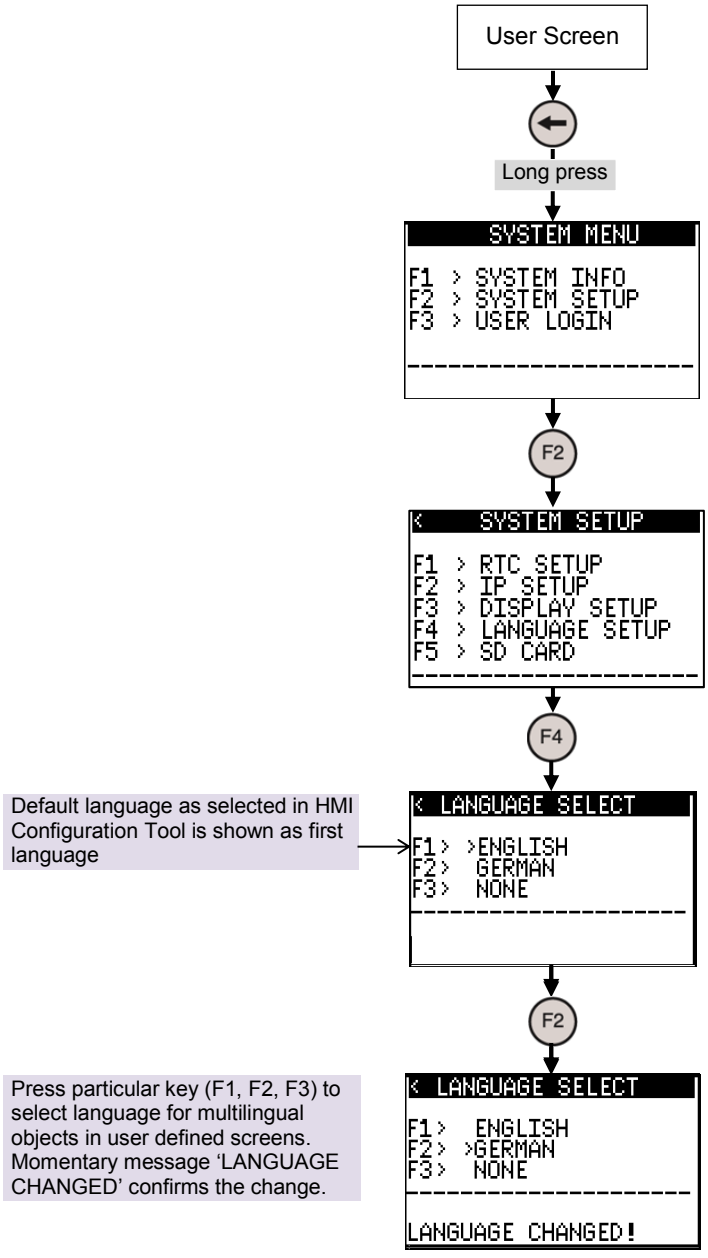


Refer section *HMI Keys* to know the steps to modify display parameter values using HMI keys.
Modify settings as required and press Enter key to store settings.

17.2.2.4 Language Setup

This setup allows user to set language of all the multilingual objects in user define screens. User can program up to 3 languages for multilingual text objects using *HMI Configuration Tool*.

Language setting is remembered even after power cycle. System menu is available in English only.



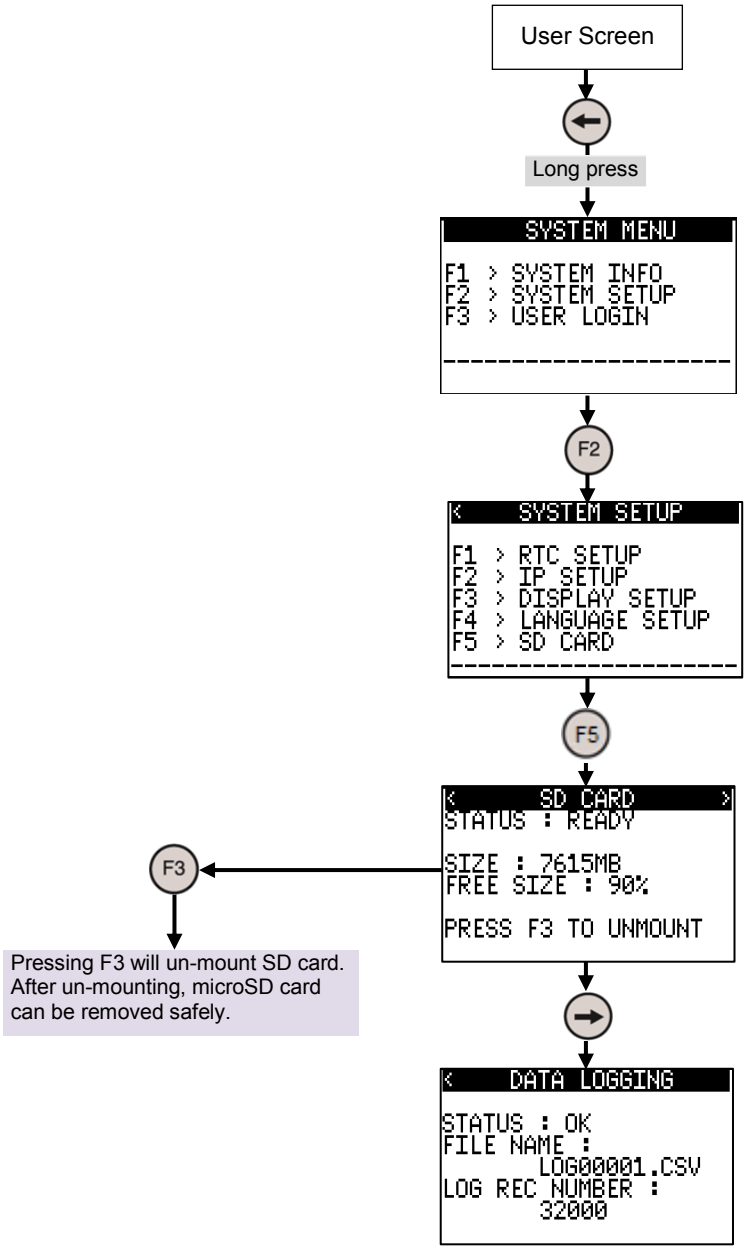
17.2.2.5 SD Card

This section of system menu shows status of SD card and Data logging.

In SD card status screen allows user to check status of SD card, total memory available on SD card and free memory in percentage of total memory.

From SD card screen user can un-mount SD card by pressing F3 key.

Data Logging screen allows user to monitor status of data logging, logging file name and number of records logged in current logging file.



17.2.3 User Login

This menu allows user to login and log out selected level. It also allows to modify password for different levels.

User can assign screen level password protection to restrict unauthorized access to the parameters in a screen. User can program password level for a key with GoTo action. It provides 4 levels of password protection from 1 to 4. Fourth level is highest priority password level. Password is 4 digit numeric password. Default passwords are as below.

Level	Default password
0	No password protection
1	1000
2	2000
3	3000
4	4000

User can login via System Menu as explained on next page.

If password is not login and if user presses key with GoTo action with password protection, then screen displays Log In screen and prompts user for login. After successful login, it switches to the new screen defined by respective GoTo action. Password level 4 has higher priority over other password levels.

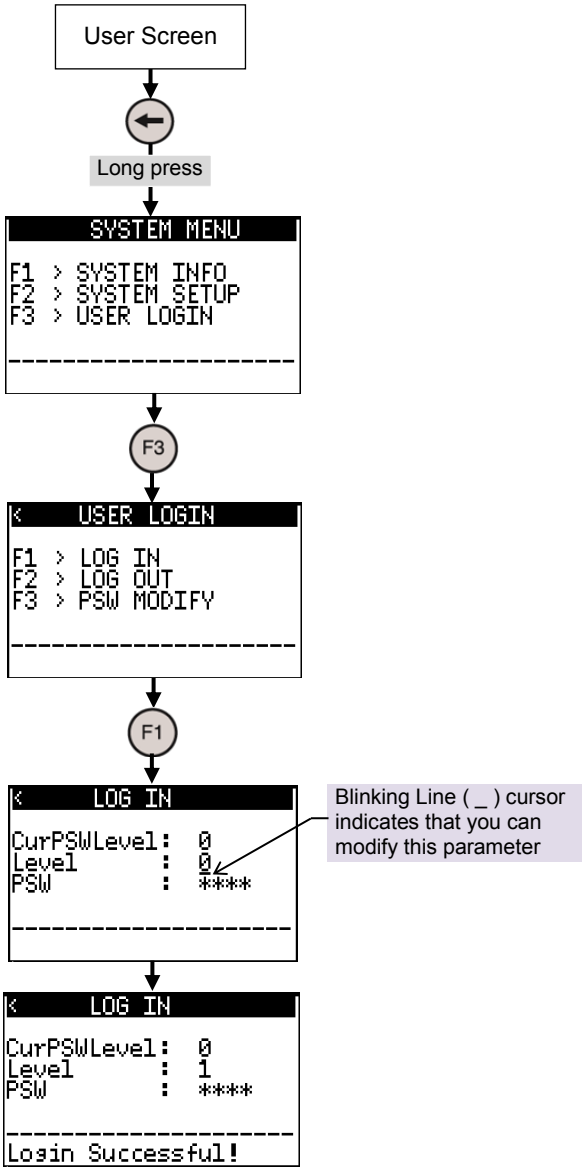
If higher level password is login, then all the lower level passwords are login automatically. e.g. if password level login is 3, user can access screens protected by password levels 1 and 2 but not the screens protected by password level 4.

After password login, if user does not carry any key operation for 3 minutes duration, then password is logout.

User can login one password level at a given time. Whenever user login any password level, previously login level is logout.

17.2.3.1 Log In

User can assign different passwords for different levels. Follow the steps below to login to a particular level.



Refer section *PSW Modify* to know the steps to enter password level and password value using HMI keys.
User can modify individual digit. Selected digit is displayed and all other digits are masked by *****.
Enter correct level and password and press Enter key to complete login.

Message 'Login Successful!' is displayed momentarily for correct login.

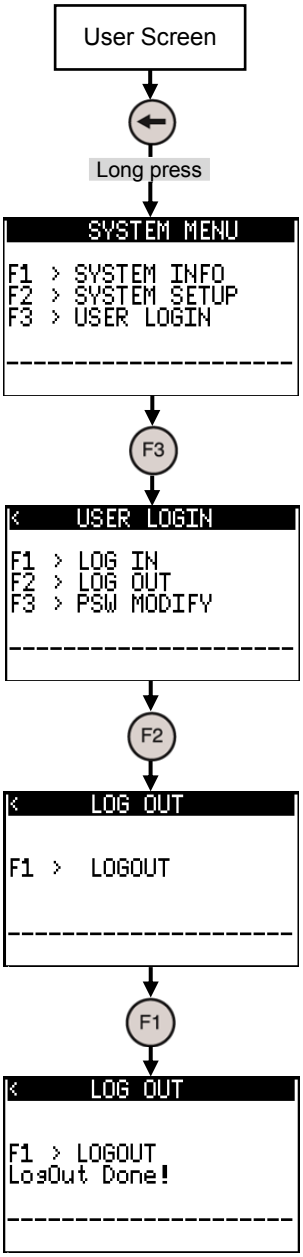
Message 'Login Failed!' is displayed momentarily for invalid password for selected level.

After password login, if user does not carry any key operation for 3 minutes duration, then password is logout automatically.

Use can login one password level at a given time. Whenever user login any password level, previously login level is logout.

17.2.3.2 Log Out

Follow the steps below to logout.

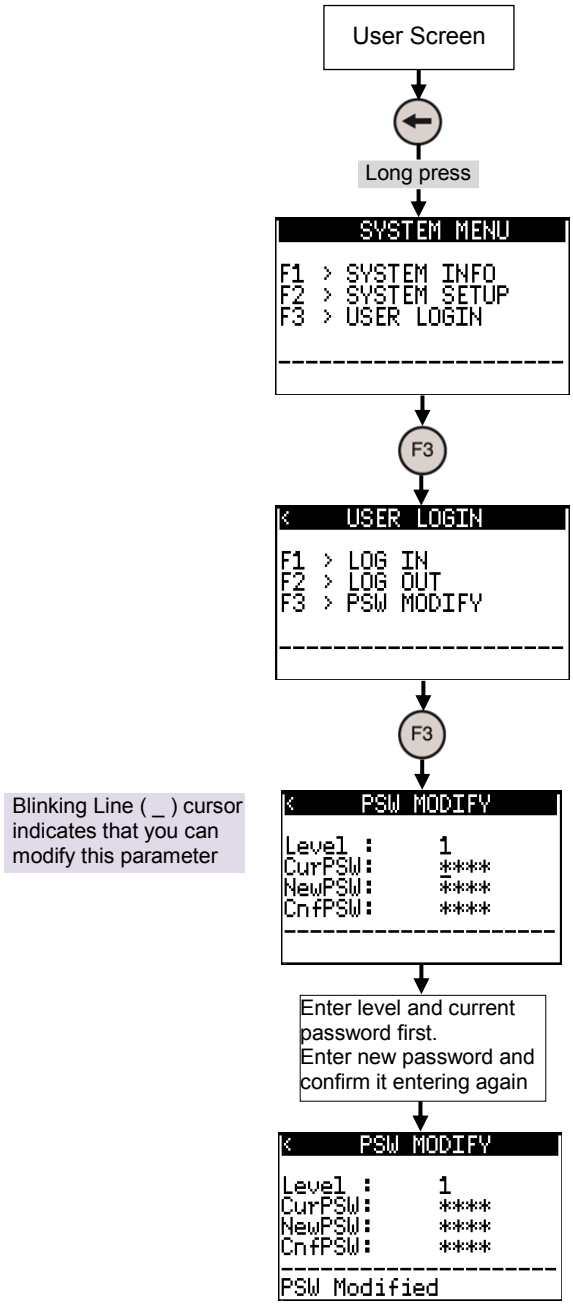


After password login, if user does not carry any key operation for 3 minutes duration, then password is logout automatically.

User can login one password level at a given time. Whenever user login any password level, previously login level is logout.

17.2.3.3 PSW Modify

User can modify password (4 digit numeric value) if old password is known. Follow the steps below to modify password for selected level.



Refer section *Log In* to know the steps to enter password level and enter current password and new password using HMI keys.

User can enter individual digit of password. Selected digit is displayed and all other digits are masked by *****.

If current password for selected level is not matched after entry, message 'CURPSW Invalid' is displayed momentarily.

If new password and confirm password does not match, then message 'Failed' is displayed momentarily.

18 Programming

GOC Toolkit is used for programming of the controller. Installation of GOC toolkit installs following components on your computer

- IEC61131-3 compatible programming software CoDeSys V2.3
- Integrated tool Hardware Configuration Tool
- Integrated tool HMI Configuration Tool
- User documentation

The purpose of integrated tools is to make GOC features easy to use and reduce application program development time. Tools provide user friendly GUI to configure and parameterize various controller features as per the application requirement.

Integrated tool *Hardware Configuration* is used for

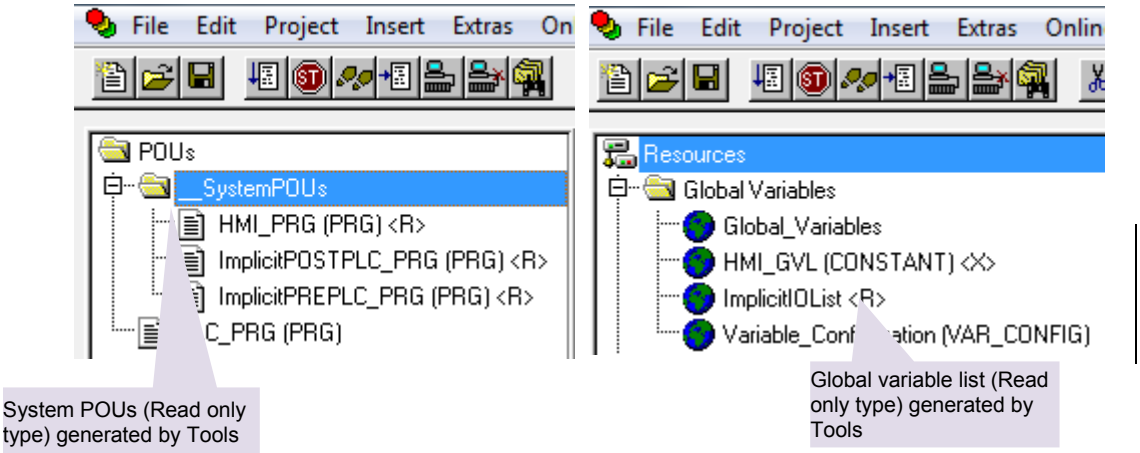
- Configuration and parameterization of Main unit and Extension units
- Special functionality configuration for digital inputs on main unit which includes high speed counter, Encoder, counter compare interrupt, hardware input interrupt, pulse catch etc.
- Serial port parameterization and protocol configuration : Assigning Modbus RTU slave/ master protocol and generating query list for Modbus RTU master
- Analog I/O configuration and handling which includes selecting analog I/O type, filtering and engineering scaling, etc

Integrated tool HMI Configuration is used for

- Fast screen definition with drag n drop objects and selecting object properties
- Assign screen actions (Entry, Display, Exit)
- Define screen chaining
- Configurable operation for function key configuration with local/global scope
- Configurable operation for illuminated keys and control of dual-colour LEDs

Thus, GOC offers single programming platform for programming PLC function and HMI function.

Tools create PROGRAM type of POUs and global variable list in CoDeSys application program and configures relevant tasks as shown below.



Programmer can develop application logic in CoDeSys as per the sequence of operation for machine or process as per application requirement.

Use programming software from 'GOC Toolkit CD'. This CD is available for purchase with ordering code 'GC-TOOLKIT'.
For software installation guidelines refer '*ED-2002-118 Installation and Start-up guideline for GOC Tool Kit*'.

18.1 Set up

Programming is possible through COM extension unit GC-ENET-COM (Ethernet interface) and GC-232-COM (serial RS232 interface) installed in COM1 slot on the backside of the controller. Refer installation Manual *PD-2002-106 GOC Main Unit (GC35MH-□□M□-D□) Installation Manual.*

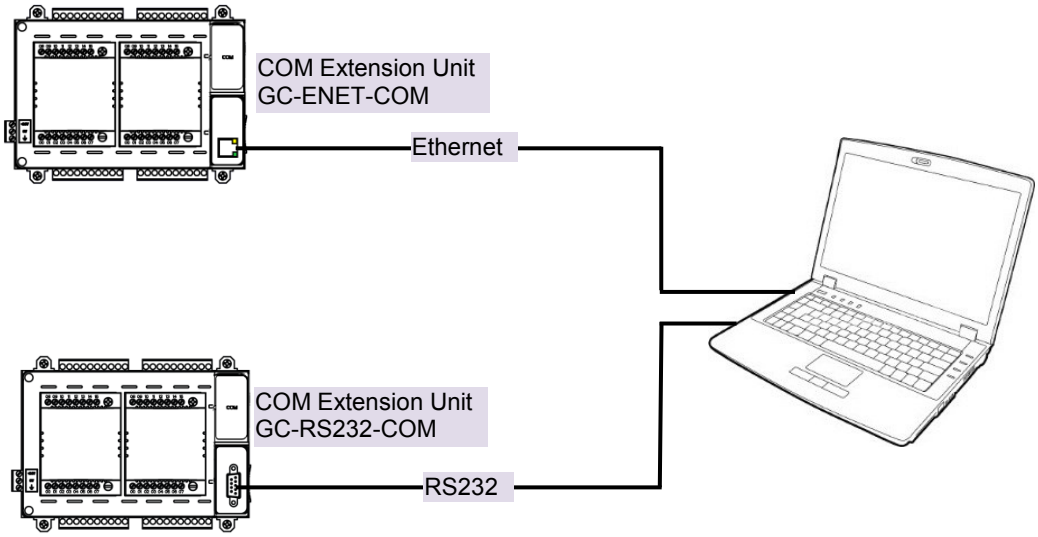
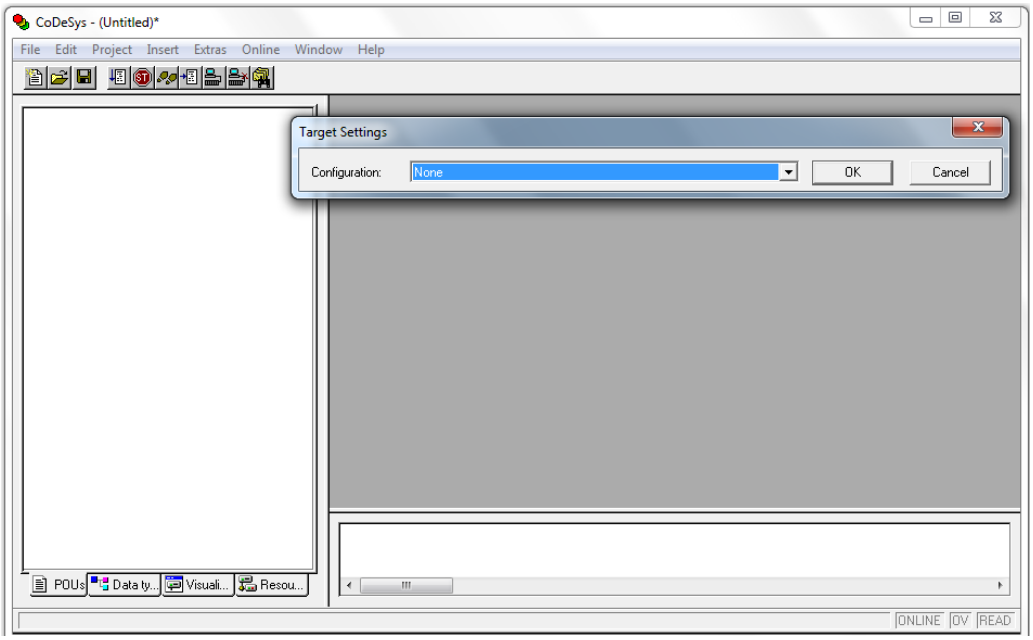


Figure 30: Programming set up

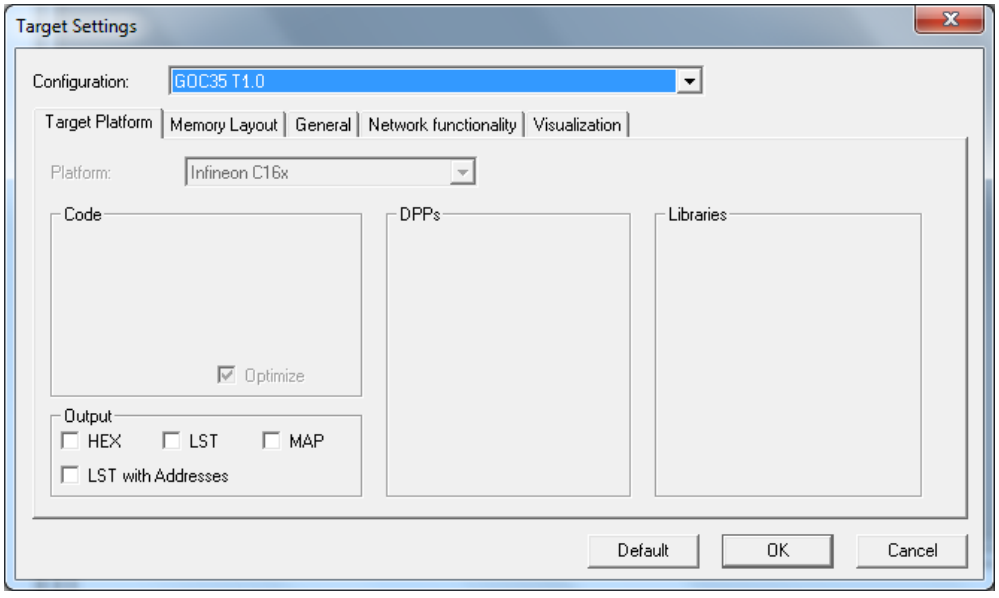
18.2 Quick Start CoDeSys

Run programming software CoDeSys, Start → All Programs → GOC Toolkit →CoDeSys V2.3 → CoDeSys V2.3

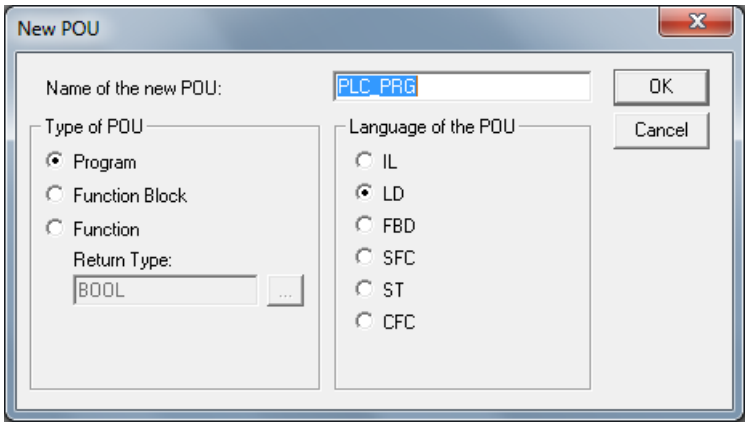
Execute menu command File → New to create new PLC program. 'Target Setting' dialogue box appears as shown below.



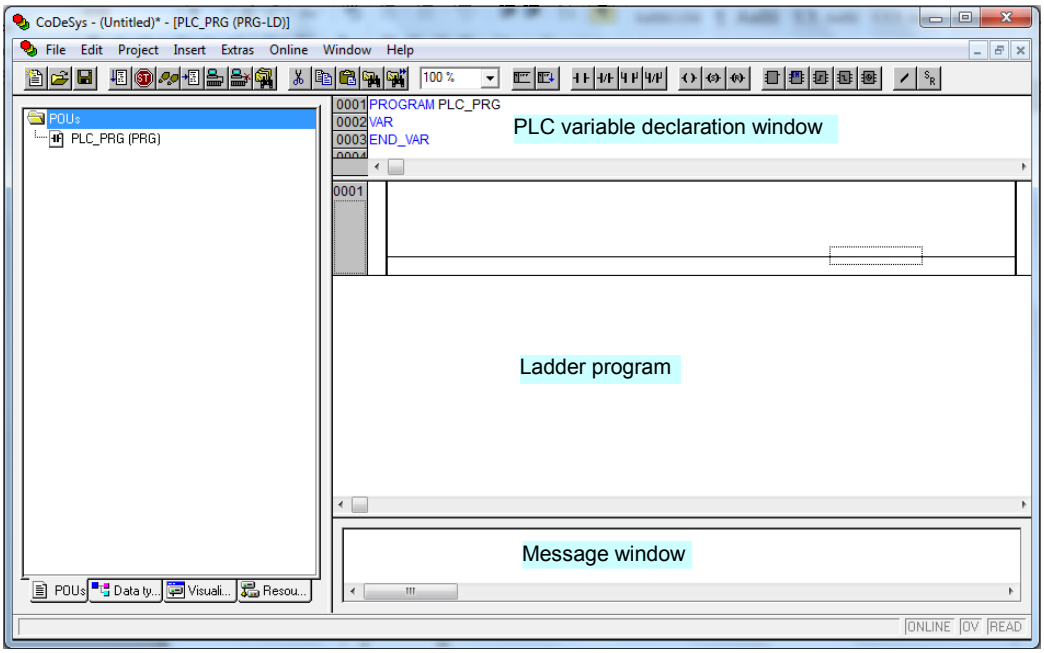
Select 'GOC target from Configuration drop down list and click on OK button as GOC35 Tm.n (Here, 'm' is 1 and 'n' is 0) as shown below.



New POU' window is popped up for program type POU 'PLC_PRG'. Select language of POU and click on OK button.

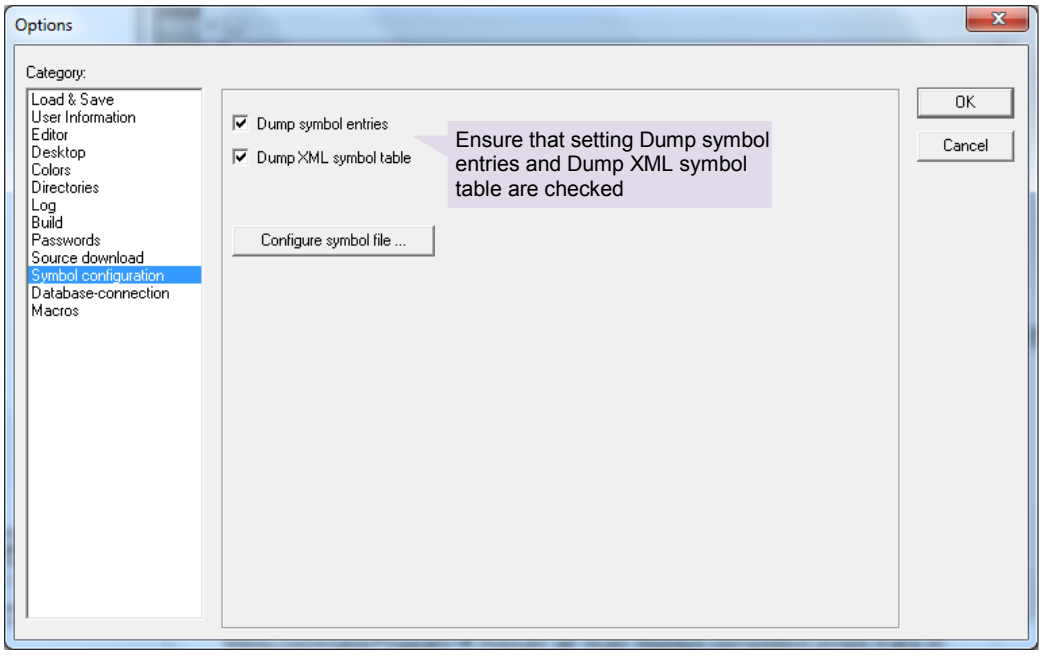


Ladder editor is displayed as shown below



Before using integrated Tool " Hardware Configuration <R> and " HMI Configuration <R>, user has to ensure that following settings are done.

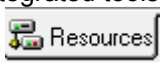
Execute menu command Project → Options →Symbol configuration to pop up window as shown below.



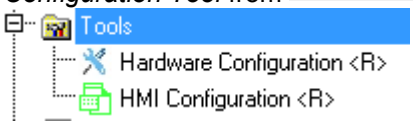
Ensure that setting Dump symbol entries and Dump XML symbol table are checked

Also ensure that Online→ Simulation Mode is not selected.

Save project with suitable name for file.

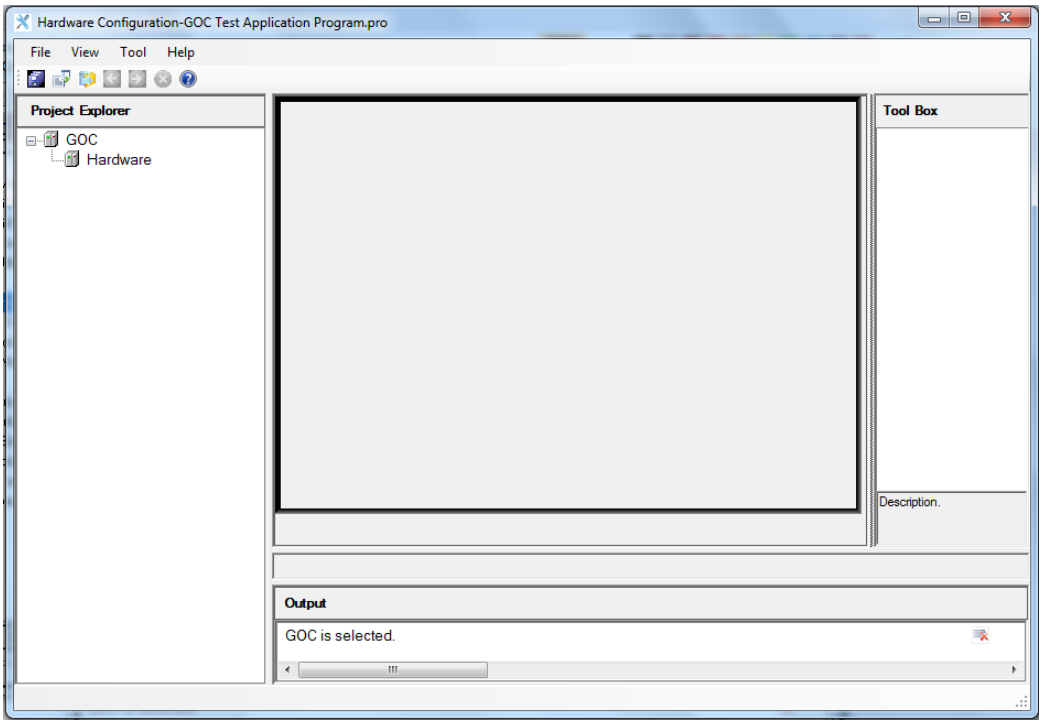
User can then invoke integrated tools like *Hardware Configuration Tool* and *HMI Configuration Tool* from 

tab from Tools folder as shown below

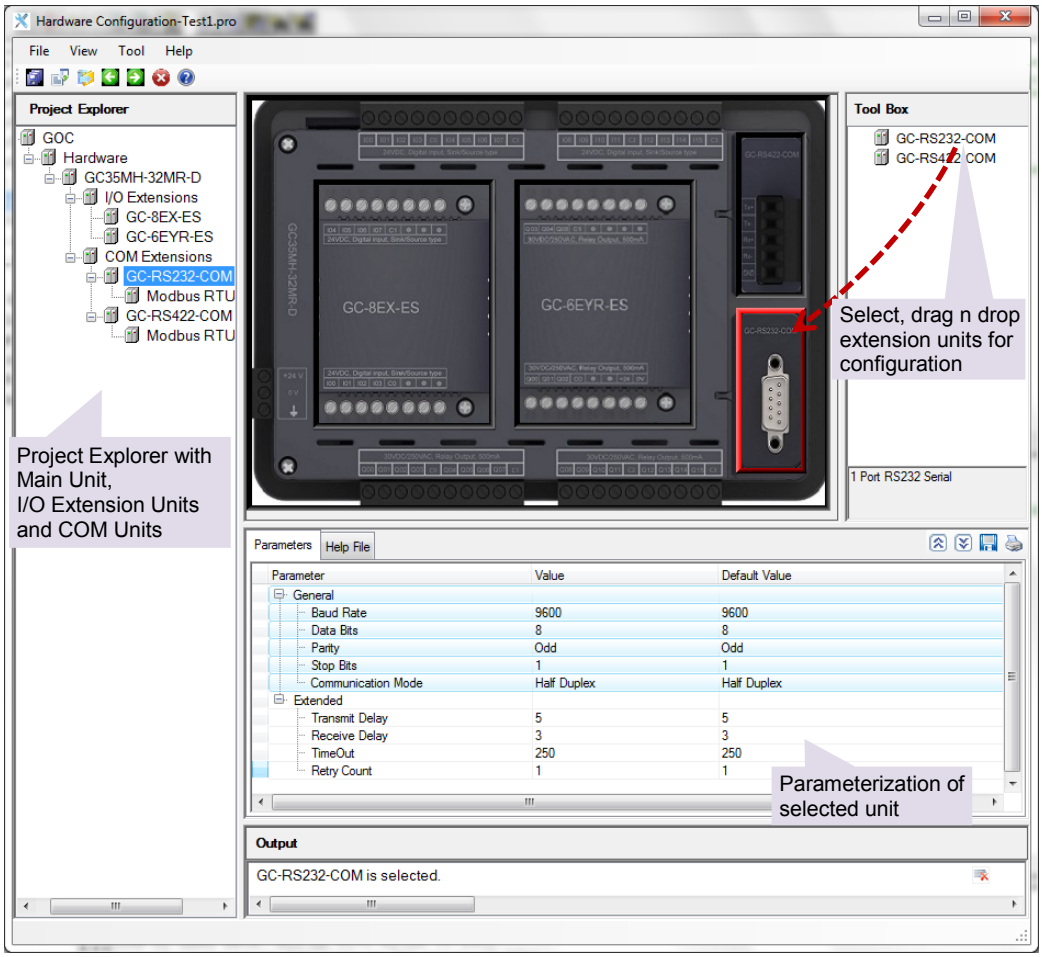


User can configure Hardware like Main unit, I/O extension units and COM extension units.

Click on Hardware Configuration <R> to invoke *Hardware Configuration Tool* as shown below.



Tool box provides necessary units and user can drag n drop units and parameterize it as per application requirement as shown below

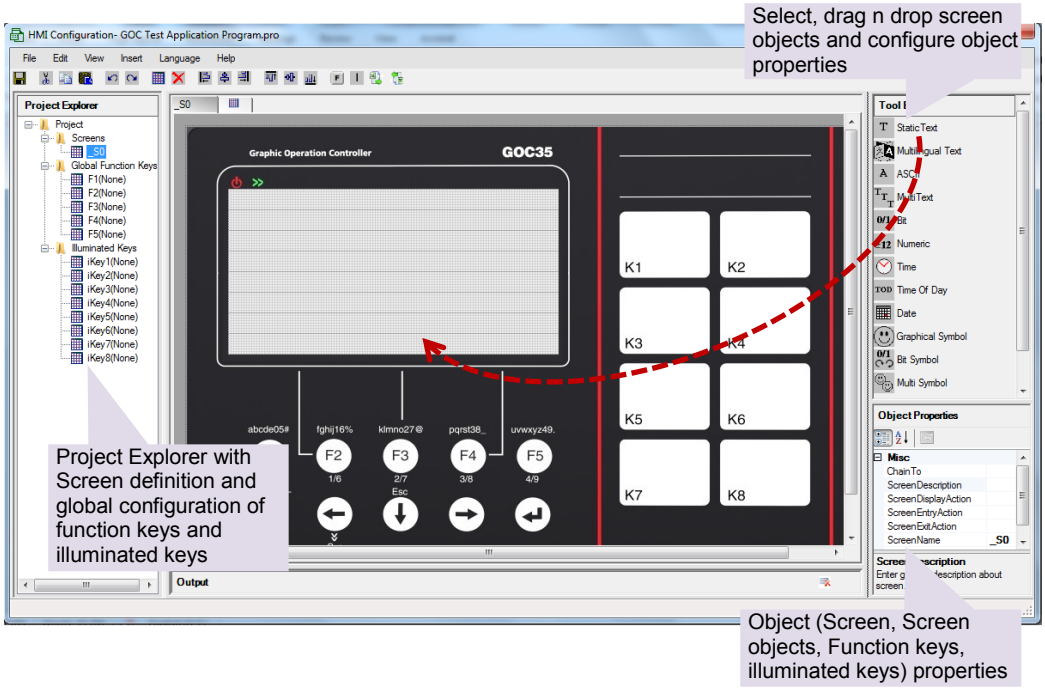


For more details, refer Help associated with Tool.

Once hardware configuration is over, user can develop application program using any IEC language, declaring variables, programming POUs, etc. Refer Help associated with CoDeSys for more details.

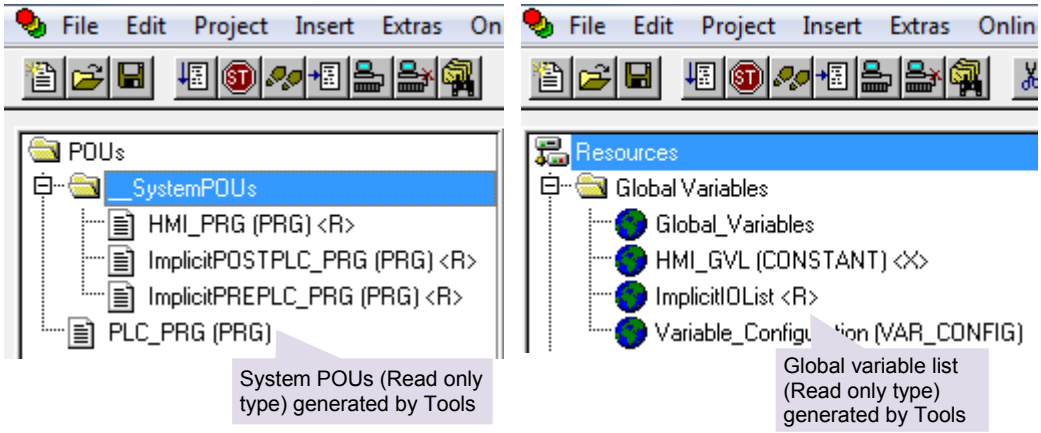
Once basic application program with sequence of operation for machine or process is ready, user can develop screens for LCD display, configure user functionality for HMI keys (F1 to F5) and illuminated keys.

Click on HMI Configuration <R> to invoke *HMI Configuration Tool* as shown below.



CoDeSys editor is deactivated temporarily.

Once configuration is done, respective Tool should be closed. Tool in turn generates system POUs and global variable list as shown below.



So HMI configuration along with hardware configuration is a part of CoDeSys application program only.

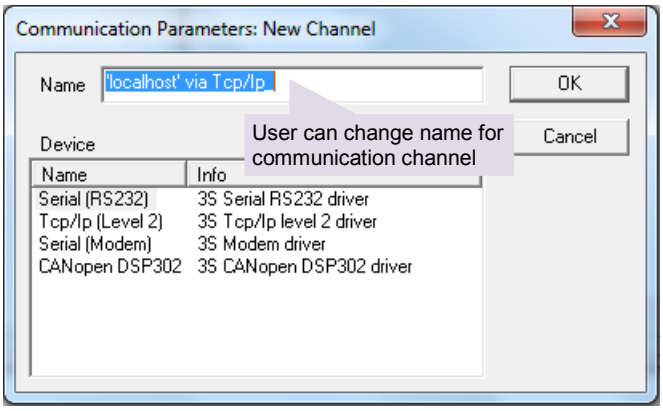
Once configuration and application program development is completed, user can download it to the controller.

Menu command Program → Rebuild all/ Build displays compilation errors if any in Messages window.

Once compilation is successful (0 Error(s)), program can be downloaded to the controller.

To download program, it is necessary to set communication parameters. Click on menu Online → Communication parameters to pop up a window.

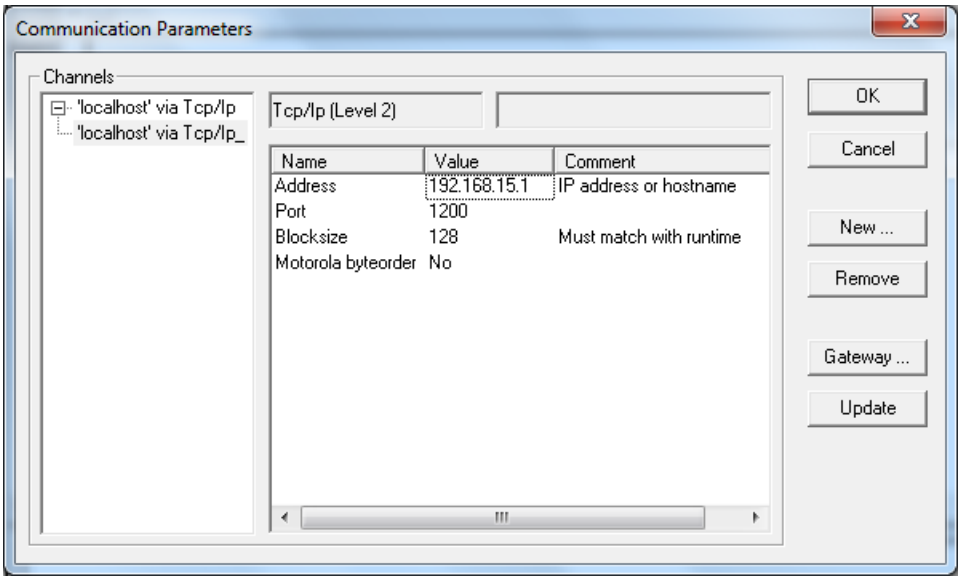
Click on New button to configure a new channel as shown below.



Select Tcp/Ip (Level 2) for Ethernet communication.

Select Serial (RS232) for serial communication.

For Ethernet channel, enter IP address of GOC in value field of Address. Do not change any other setting.

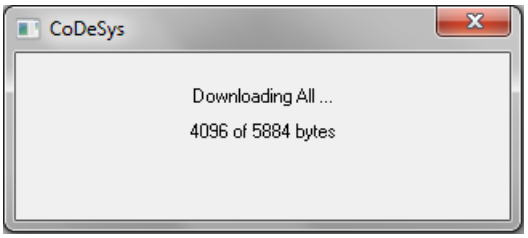


For Ethernet communication, ensure that your PC is in same subnet.

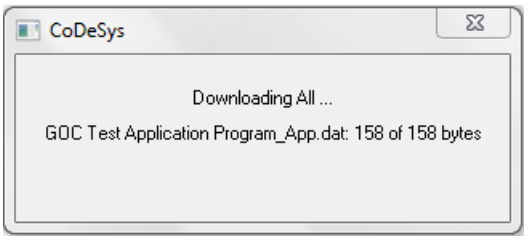
For serial channel, user can select PC COM port.

To download program to controller, execute menu command Online → Login.

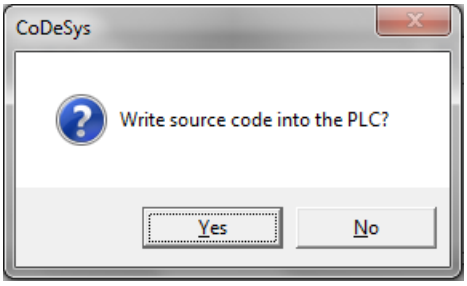
During download, following message box is displayed.



Once application code download is complete, <Project Name>_App. Data file which consists of data like multilingual object data is downloaded.

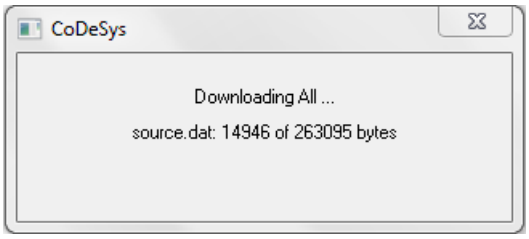


Then following dialogue is pooped up as shown below.



User has to download source code at appropriate time (e.g. application program development is completed up to desired level). Source code is a compressed project file that consists of application project with project information like comments, symbolic names, libraries (optional), etc. If source code is downloaded to the controller, it can be uploaded and retrieved.

Click on Yes button to download source code to the controller that may take considerable time depending upon size of source code. **Downloading of source code is mandatory.**



Once, source code download is over, put controller in RUN mode

Controller starts executing application program, update inputs and outputs and LCD screen.

To upload application program from PLC, execute menu command File → Open → Open project from PLC. But upload is possible only if source code is downloaded earlier.

19 Controller Memory

When programming a PLC and HMI, it is important to understand the different types of memory available and how to access it. In GOC, two types of memory is used; RAM (volatile memory) and Flash (permanent memory). RAM type of memory holds PLC variable data (input, output, marker, data and system variable) whereas Flash type of memory holds application program code, source code and application data. The table below shows utilization of RAM and Flash memory.

Memory	Description	Addressing	Size
RAM	Input process image	Addressable as %I	32 Bytes
	Output process image	Addressable as %Q	32 Bytes
	Marker *	Addressable as %M	4 Kbytes
	Data *	Addressable by user defined symbolic names	24 Kbytes
	System variables	Addressable by predefined symbolic names	512 Bytes
Flash	Application program code	Not addressable	192 Kbytes
	Application program source code	Not addressable	1.5 Mbytes
	Application data	Not addressable	64 Kbytes

* User can declare some part of marker memory and data memory as retentive as per application need. Retentive data is stored in FRAM.

19.1 Input Process Image

The processor scans all the input points from main unit and extension units in input scan and stores the status in input process image. The application program then refers this status in the logic scan. Input process image is addressable global memory. Hence, external devices like HMI and SCADA can access it via protocols like Modbus RTU protocol. In GOC, input memory size is 32 bytes and is addressed from %IB0 to %IB31. Input memory can also be accessed by user defined symbolic names. The input process image is not retained.

During logic scan, instruction `Refresh_In` can read the input status from a particular slot and update input process image if required.

19.2 Output Process Image

The processor updates the status of output points as per the application program in the logic scan and stores the updated status in the output process image. After logic scan, the processor executes output scan. The output scan activates the actual outputs on main unit and extension units as per the output process image. Output image is addressable global memory. Hence, external devices like HMI and SCADA can access it via protocols like Modbus RTU protocol. In GOC, output memory size is 32 bytes and is addressed from %QB0 to %QB31. Output memory can also be accessed by user defined symbolic names. The output process image is not retained.

During logic scan, instruction `Refresh_Out` can activate outputs on a particular I/O slot if required.

19.3 Marker Memory

Marker memory holds the intermediate results in the application program. This is addressable global memory. Hence, external devices like HMI and SCADA can access it via protocols like Modbus RTU protocol. In GOC, marker memory size is 4000 bytes and is addressed from %MB0 to %MB3999. Marker memory can also be accessed by user defined symbolic names.

By default, 300 bytes of marker memory from %MB3700 to %MB3999 is retained.

19.4 Data Memory

Data memory holds the intermediate results and Function Block instance data. This memory is addressed by user defined symbolic name only. The programming software 'CoDeSys' assigns the address to such PLC variable during compilation of the application program. This address is for internal purpose and may change during number of compilations at the time of application program development. So the addresses of such variables are not fixed and external devices like HMI and SCADA cannot access it via protocols like Modbus RTU protocol.

User can retain 700 bytes of data memory.

19.5 I/O Memory Mapping

Digital I/O memory mapping is fixed with respect to main unit and extension units.

This I/O memory mapping is fixed irrespective of type of main and extension unit. If any I/O points are not used, respective I/O memory is redundant. If any extension is not used, respective input/output memory is redundant and input byte holds 0 permanently where as if output byte is modified in application program there is no action.

Unit	Slot number	Digital input address	Digital output address
Main	0	%IB00 to %IB03	%QB00 to %QB03
COM1 Extension	1	%IB04 to %IB05	%QB04 to %QB05
COM2 Extension	2	%IB06 to %IB07	%QB06 to %QB07
Reserved	--	%IB08 to %IB11	%QB08 to %QB11
IO1 Extension	5	%IB12 to %IB13	%QB12 to %QB13
IO2 Extension	6	%IB14 to %IB15	%QB14 to %QB15
Reserved	--	%IB16 to %IB31	%QB16 to %QB31

Instructions `Refresh_In` and `Refresh_Out` can update digital I/Os immediately in logic scan.

Input image of digital inputs in main unit is updated even if inputs are configured for special functionality like high speed counter but it may not be useful in the application.

For COM extension units, input memory provided status of communication and output memory is used for commands and settings.

For IO extension units like analog I/Os, input memory provided status of analog channels and output memory is used for commands and settings.

Addressing I/Q/M memory

The figure below shows the addressing of input, output and marker memory.

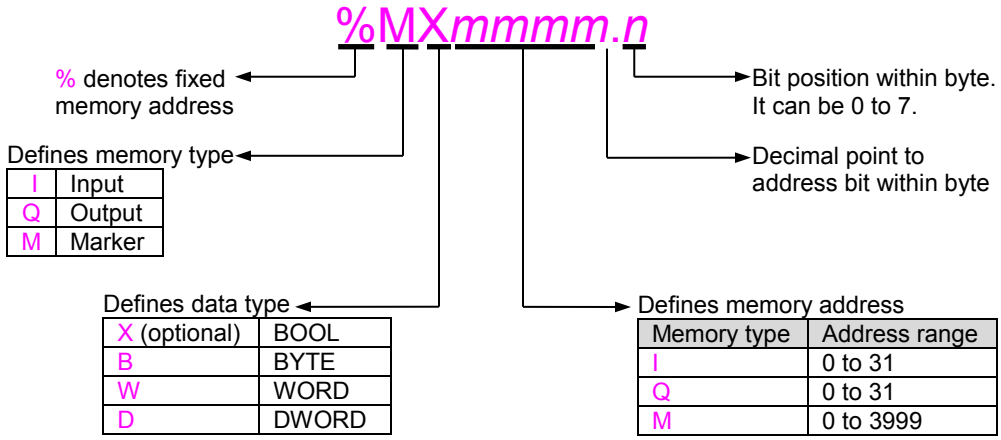


Figure 31: Addressing of PLC variables

The figure below shows memory mapping and significance of BOOL, BYTE, WORD and DWORD data type for marker memory as an example.

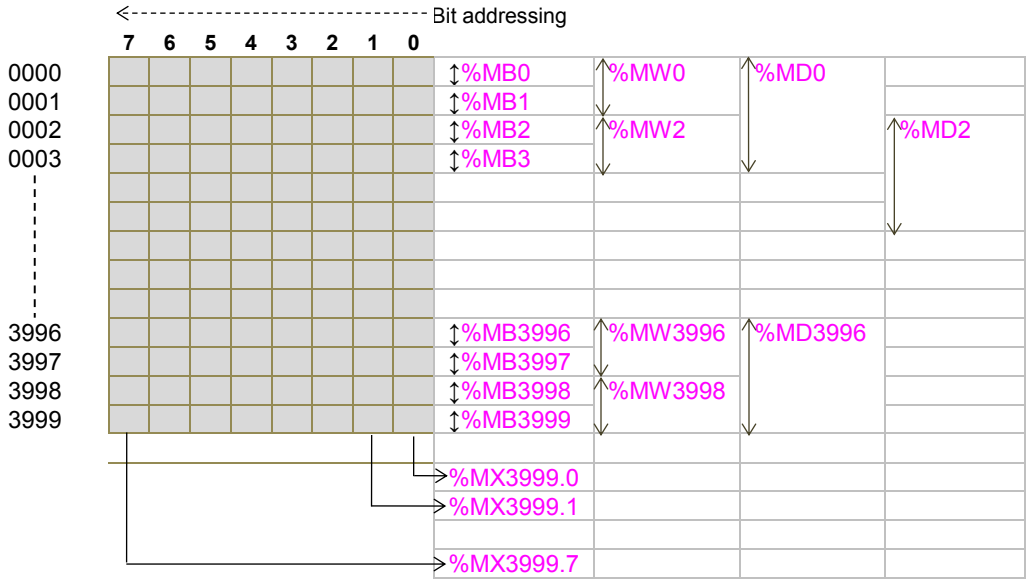


Figure 32: Memory mapping

Points to remember

1. In GOC, memory mapping is byte wise. All the 8 bits (0 to 7) within a byte can be addressed.
2. When memory is addressed as word, then two consecutive bytes are accessed. Byte at start address is lower byte and next byte is higher byte. Memory words can be accessed only with even address e.g. %MW0, %QW2, %MW4 and not %IW1, %MW3.
3. When memory is addressed as double word, then two consecutive words (i.e. four consecutive bytes) are accessed. Word at start address is lower word and next word is higher word. Memory double words can be accessed only with even address e.g. %ID0, %QD2, %MD4, %MD3996 and not %ID1, %QD3, %MD3991.
4. Addressing double words like %MD2, %MD4 in same application program will certainly overlap 2 bytes (%MB4 and %MB6 in this case) and should be avoided.

19.6 Addressing Range

The following table shows addressing range supported by GOC.

PLC Memory	Type	Data Type	Range
Input BOOL	I	X	%IX0.0 to %IX0.7 %IX1.0 to %IX1.7 ⋮ %IX31.0 to %IX31.7
Input BYTE	I	B	%IB0, %IB1, %IB2 to %IB31
Input WORD	I	W	%IW0, %IW2, %IW4 to %IW30
Input DWORD	I	D	%ID0, %ID4, %ID8 to %ID28
Output BOOL	Q	X	%QX0.0 to %QX0.7 %QX1.0 to %QX1.7 ⋮ %QX31.0 to %QX31.7
Output BYTE	Q	B	%QB0, %QB1, %QB2 to %QB31
Output WORD	I	W	%QW0, %QW2, %QW4 to %QW30
Output DWORD	I	D	%QD0, %QD4, %QD8 to %QD28
Marker BOOL	M	X	%MX0.0 to %MX0.7 %MX1.0 to %MX1.7 ⋮ %MX3999.0 to %MX3999.7
Marker BYTE	M	B	%MB0, %MB1, %MB2 to %MB3999
Marker WORD	M	W	%MW0, %MW2, %MW4 to %MW3998
Marker DWORD	M	D	%MD0, %MD2, %MD4 to %MD3996

19.7 Retained Memory

Retentive memory is a memory that is declared by the user to maintain values through a power cycle or warm initialization. GOC allows 1000 bytes of memory to retain. The table below shows memory type and maximum size of memory that can be retained

Data memory	700 bytes
Marker memory	300 bytes (from %MB3700 to %MB3999)

Retained data is stored in FRAM type of memory. Cold initialization resets entire memory (including retentive memory) to 0 or user defined initial value.

Points to remember

1. User can define PLC variable as retentive using keyword VAR RETAIN. Local as well as global variable can be declared as retentive.

   e.g.

   ```
   VAR RETAIN
       Data1: WORD;
   END_VAR
   ```

   Here, variable declared with symbolic name Data1 is retained.

2. Marker memory form %MB3700 to %MB3999 is retained by default if user accesses it directly by marker memory address (and not declared with some symbolic name). Whereas remaining marker memory from %MB0 to %MB3699 is cleared at warm initialization.

3. If any variable with symbolic name is mapped at marker memory address from %MB3700 to %MB3999, it is cleared at warm initialization if declared as shown below

   ```
   VAR
       Data1 AT%MW3700: WORD;
   END_VAR
   ```

   To retain this variable, user has to declare it as retentive as below

   ```
   VAR RETAIN
       Data1 AT%MW3700: WORD;
   END_VAR
   ```

4. Any variable mapped outside specified marker memory is not retained even though declared as retentive.

   e.g.

   ```
   VAR RETAIN
       Data10 AT%MW1000: WORD;
       Op2 AT%QB2: BYTE;
   END_VAR
   ```

   Here, variables Data10 and Op2 will not be retained.

5. If user declares function block instance as retentive, then the complete instance of the function block (all the data of function block instance) is retained.

   e.g.

   ```
   VAR RETAIN
         T1: TON;
   END_VAR
   ```

   If instance T1 of ON delay timer TON is declared as retentive, then 15 bytes of data memory is retained.

6. During application program compilation, programming software 'CoDeSys' checks PLC variables declared by symbolic name (and not mapped at I/Q/M memory) for retained size limit of 700 bytes. If retained size exceeds 700 bytes, it displays compilation error as

   Error 3802: <POU Name>(<line number>): Out of retain memory. Variable '<name>', <number> bytes

   But for PLC variables mapped at marker memory (as well as for input and output memory), it does not check for retained size limit of 300 bytes. It does not display any compilation error for user mistake.

19.8 System Variables

The system variables are implicitly defined global variables. These variables exchange the information between CPU and application program. Each system variable has a unique name which starts with underscore '`_Sysvar`'. These system variables are useful to know the system status and for diagnostics.

For more details, refer chapter *System Variables*.

19.9 Application Program Memory

GOC stores application program in flash memory in form of code and source code.

Application Program Code (Boot Project)

The programming software 'CoDeSys' downloads complied project when

1. Menu command Online→ Download is executed or
2. Menu command Online → Login Alt+F8 is executed and there is mismatch between complied project and existing project in GOC

This complied project is called the application program code or boot project which is executed by the processor.

Maximum application program code size is 192 kbytes

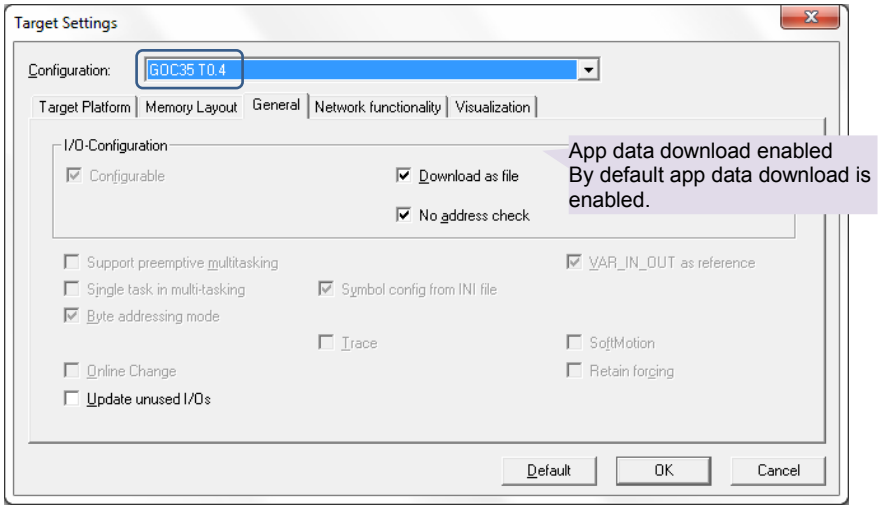
Points to remember

1. Application program code (boot project) is not retrievable i.e. cannot be uploaded as 'CoDeSys' project file.
2. CoDeSys downloads application program code in flash memory. In case of unresolved external POUs (POUs in external library those are not supported by CPU firmware), CoDeSys prompts programmer at the end of download and PLC remains in STOP mode indicating memory error. At this point, programmer must download a valid application program code and put PLC in RUN mode.
   Instead if programmer recycles PLC power, PLC may start executing application code containing unresolved external POUs and may malfunction.

Application Data

64000 bytes of memory is reserved for application specific data that contains information of multilingual data objects and configuration information of data logging using SD memory card. HMI Configuration Tool creates application data file named as '<ProjectName>_App.dat'.
This data is downloaded in continuation with application program download. This data cannot be uploaded back from controller. However app data is recreated by HMI configuration tool once tool is opened from uploaded CoDeSys project.

Download of application data to controller can be enabled / disabled by following setting in target setting



Application Program Source Code

The programming software 'CoDeSys' enables user to develop the application program using various IEC languages. This application program is saved as <Project_Name>.pro file on computer hard disk. The project file contains all project related information e.g. POUs in various IEC languages, program comments, variable declarations with symbolic name and comments, password, libraries (optional), etc. It is necessary to store all this information called as Source code in GOC in the format defined by user.

Menu command Online→ Sourcecode download downloads application program source code to flash memory. Source code download is possible when CoDeSys is in Online monitoring mode.

Menu command File→ Open Ctrl+O → Open project from PLC uploads project file from GOC and present it in **.pro** file format.

Maximum source code size is 1.5 Mbytes.

Points to remember

1. Download source code is mandatory so that entire application project remains with GOC main unit and can be retrieved later on whenever required.
2. Programmer can restrict unauthorized uploading of source code by programming read protection password and write protection password.
3. During source code download / upload, HMI LCD screen is not updated.
4. During Update OS or downloading New OS using PC based Bootstrap utility, application program code (User Ladder Code) can be erased as per choice but source code cannot be erased.

The block diagram below shows CoDeSys menu commands to download and upload application program.

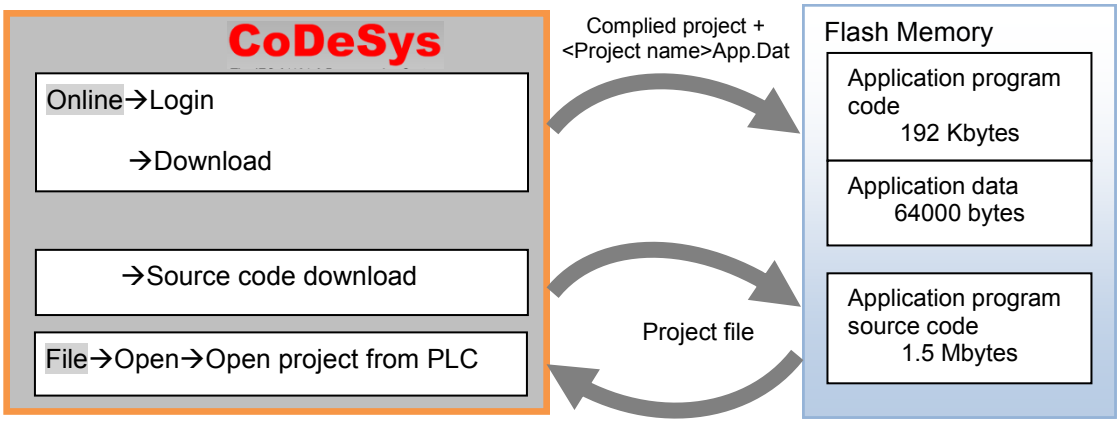


Figure 33: CoDeSys menu commands

20 Real Time Clock

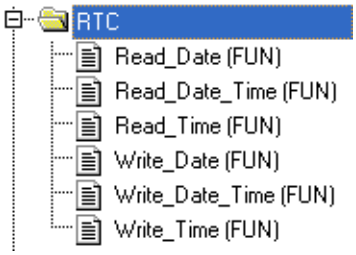
GOC has a built-in real time clock (RTC).Clock time resolution is 1 sec.Clock time updation rate is 500 ms. RTC is backed up by super capacitor and back up time is 2 weeks nominal at 25ºC ambient.RTC error is ±2 Secs maximum per day.

If RTC is currupted due to loss of back up or due to any reason, then it is initialized to date of January 1, 1970 and time of 00H:00M:00S:00MS. In such case user has to ensure that super capacitor back up is restored and proper value of RTC is set again. In case of RTC error, system variable bit `_SysvarRTC.BRTCInvalid` is set and RTC Status is displayed in system info screen from system menu. Also, system variable `_SysvarCPU.WSTATUS.4` is set to indicate RTC error status.

Current clock time is available in system variables structure `_SysvarRTC` as shown in the table below and can be referred in the application program.

System Variable	Data Type	Description
`_SysvarRTC.`		
`BRTCINVALID`	BYTE	TRUE, if RTC is corrupted or invalid
`BREADSECS`	BYTE	Holds current “Seconds” value of RTC
`BREADMINS`	BYTE	Holds current “Minutes” value of RTC
`BREADHRS`	BYTE	Holds current “Hours” value of RTC
`BREADDATE`	BYTE	Holds current “Date” value of RTC
`BREADMONTH`	BYTE	Holds current “Month” value of RTC
`BREADYEARL`	BYTE	Holds current “Year” value (lower byte) of RTC
`BREADYEARH`	BYTE	Holds current “Year” value (higher byte) of RTC

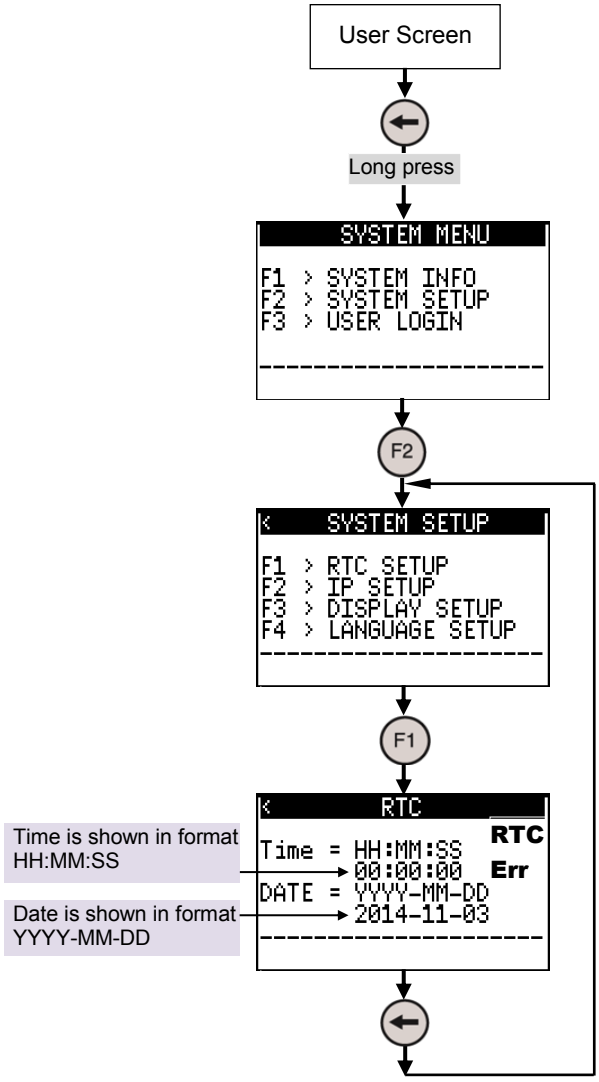
GOC35 library supports various functions to read from and write to Real Time Clock through application program as shown below.



Description
Returns RTC Date in IEC data type DATE format
Returns RTC Date and TOD in IEC data type format
Returns RTC Time in IEC data type TOD format
Writes new Date to RTC
Writes new Date and TOD to RTC
Writes new TOD to RTC

Refer *ED-2002-540 GOC35 library manual* to know more about usage of such functions in the application program.

Additionally, user can set and monitor Real Time Clock on HMI LCD. System menu provides the screen as shown below.



Refer chapter *Status and Diagnostics* for more details related to system menu which displays system status and dignostic information and allows settings of few system parameters.

In case of RTC error due to loss of super capacitor back up, below screen is dispalyed for 5 Sec duration after power ON.

21 High Speed Counter

GOC main unit and I/O extension unit GC-4HSXPTY provides high speed inputs. By default, these inputs function as general purpose digital inputs. These inputs can be configured for different modes of counter operations. The numbering of counters and different modes of operation with maximum frequency allowed as well as necessary program blocks is explained in the table below.

For Main Unit (channel numbers 0 and 3):

Inputs / Description	I00	I01	I02	I03	I04	I05
Single phase counter with software direction control	HSC0	-	-	HSC3	-	-
	20 KHz	-	-	20 KHz	-	-
Quadrature encoder A phase, B phase	ENCAB0		-	ENCAB3		-
	10 KHz		-	10 KHz		-
Quadrature encoder A phase, B phase with Z pulse	ENCABZ0			ENCABZ3		
	10 KHz			10 KHz		

For I/O Extension Unit GC-4HSXPTY (channel numbers 10 and 13):

Inputs / Description	I00	I01	I02	I03	I04	I05
Single phase counter with software direction control	HSC10	-	-	HSC13	-	-
	100 KHz	-	-	100 KHz	-	-
Single phase counter with Hardware direction control	HSC10		-	HSC13		-
	100 KHz		-	100KHz		-
Quadrature encoder A phase, B phase	ENCAB10		-	ENCAB13		-
	50 KHz		-	50 KHz		-
Quadrature encoder A phase, B phase with Z pulse	ENCABZ10			ENCABZ13		
	50 KHz			50 KHz		

All the counters are 32-bit bi-directional counters.

21.1 Single Phase Counter with Software Direction

Main unit and I/O extension unit GC-4HSXPTY provides up to 2 high speed inputs which can be configured for single phase counter operation (32-bit bi-directional) and counting direction can be changed through the application program.

Inputs I00 (HSC0/ HSC10) and I03 (HSC3/ HSC13) are single phase counters.

The figure below shows action of Reset and Direction control on single phase counter.

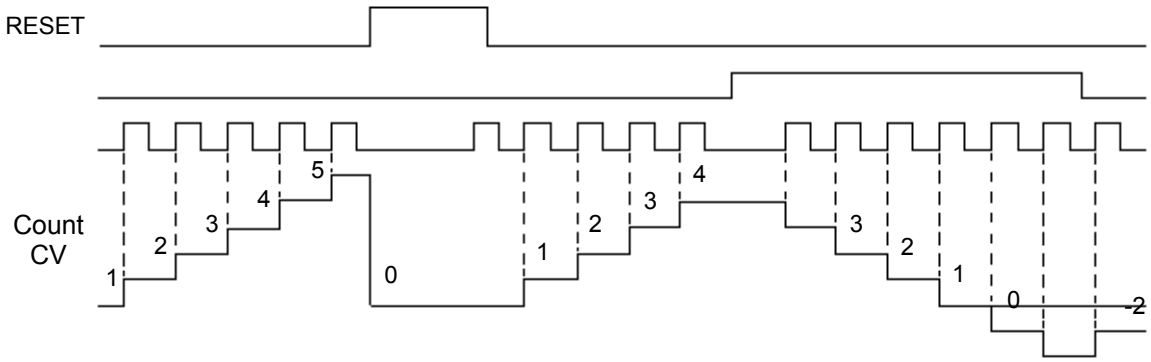


Figure 34: Functioning of single phase counter with software direction

At rising edge (OFF to ON) at input, count increments by 1 if counting direction set is FALSE. At rising edge (OFF to ON) at input, count decrements by 1 if counting direction set is TRUE.

Counter current value is reset to 0 as long as Reset is TRUE.

21.2 Single Phase Counter with Hardware Direction

I/O extension unit GC-4HSXPTY provides up to 2 high speed inputs which can be configured for single phase counter operation (32-bit bi-directional) and counting direction can be changed through digital input.

Inputs I00 (HSC10) and I03 (HSC13) are single phase counters.Inputs I01 and I04 decide direction of counting respectively.

The figure below shows action of Reset and Direction control on single phase counter.

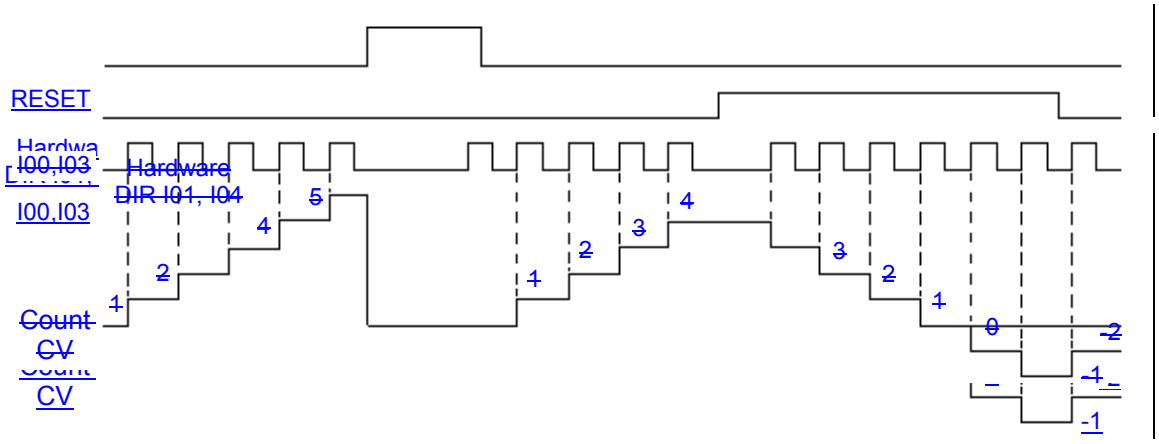


Figure 35: Functioning of single phase counter with hardware direction

21.3 Encoder A B Phase

Main unit and I/O extension unit GC-4HSXPTY provides up to 2 encoder interfaces. Counter provides 32 bit bi-directional count.

Two inputs I00 (A phase) and I01 (B phase) along with common terminal C0 provide one encoder interface as ENCAB0/ ENCAB10.

Two inputs I03 (A phase) and I04 (B phase) along with common terminals C0 and C1 provide another encoder interface as ENCAB3/ ENCAB13.

The figure below shows action of Reset and direction control depending upon phase shift beteen A phase and B phase.

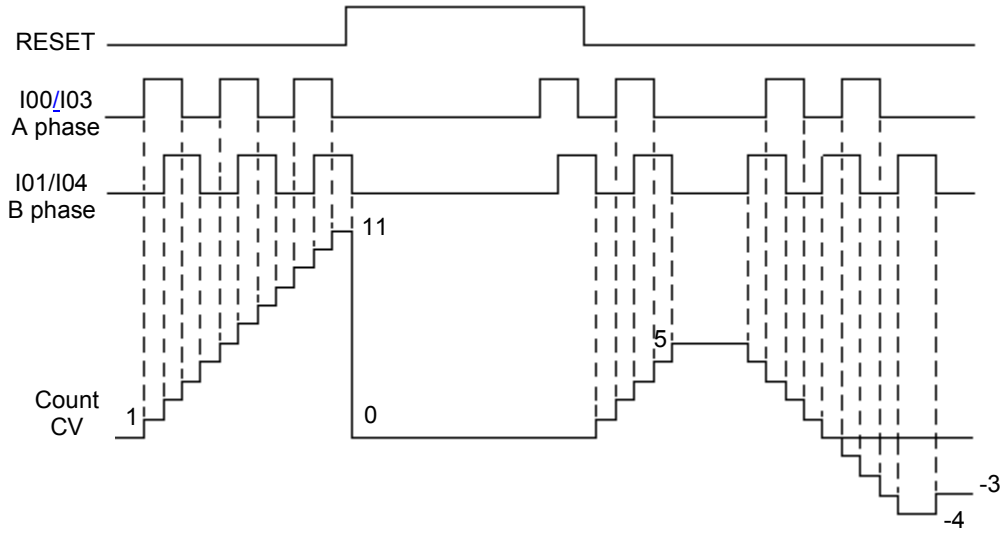


Figure 36: Functioning of AB encoder

At rising edge (OFF to ON) as well as falling edge (ON to OFF) at A phase and at rising edge (OFF to ON) as well as falling edge (ON to OFF) at B phase, count increments by 1 if A phase is leading B phase.

At rising edge (OFF to ON) as well as falling edge (ON to OFF) at A phase and at rising edge (OFF to ON) as well as falling edge (ON to OFF) at B phase,count decrements by 1 if A phase is lagging B phase.

Counter current value is reset to 0 as long as Reset is TRUE.

21.4 Encoder A B Phase with Z Pulse

Main unit and I/O extension unit GC-4HSXPTY provides up to 2 encoder interfaces. Counter provides 32-bit bi-directional count.

Three inputs I00 (A phase), I01 (B phase), I02 (Z marker pulse) along with common terminal C0 provide one encoder interface as ENCABZ0/ ENCABZ10.

Three inputs I03 (A phase), I04 (B phase), I05 (Z marker pulse) along with common terminals C0 and C1 provide one encoder interface as ENCABZ3/ ENCABZ13.

User can program Z input to reset counter current value on occurance.The figure below shows action of Reset and Z input on encoder count.

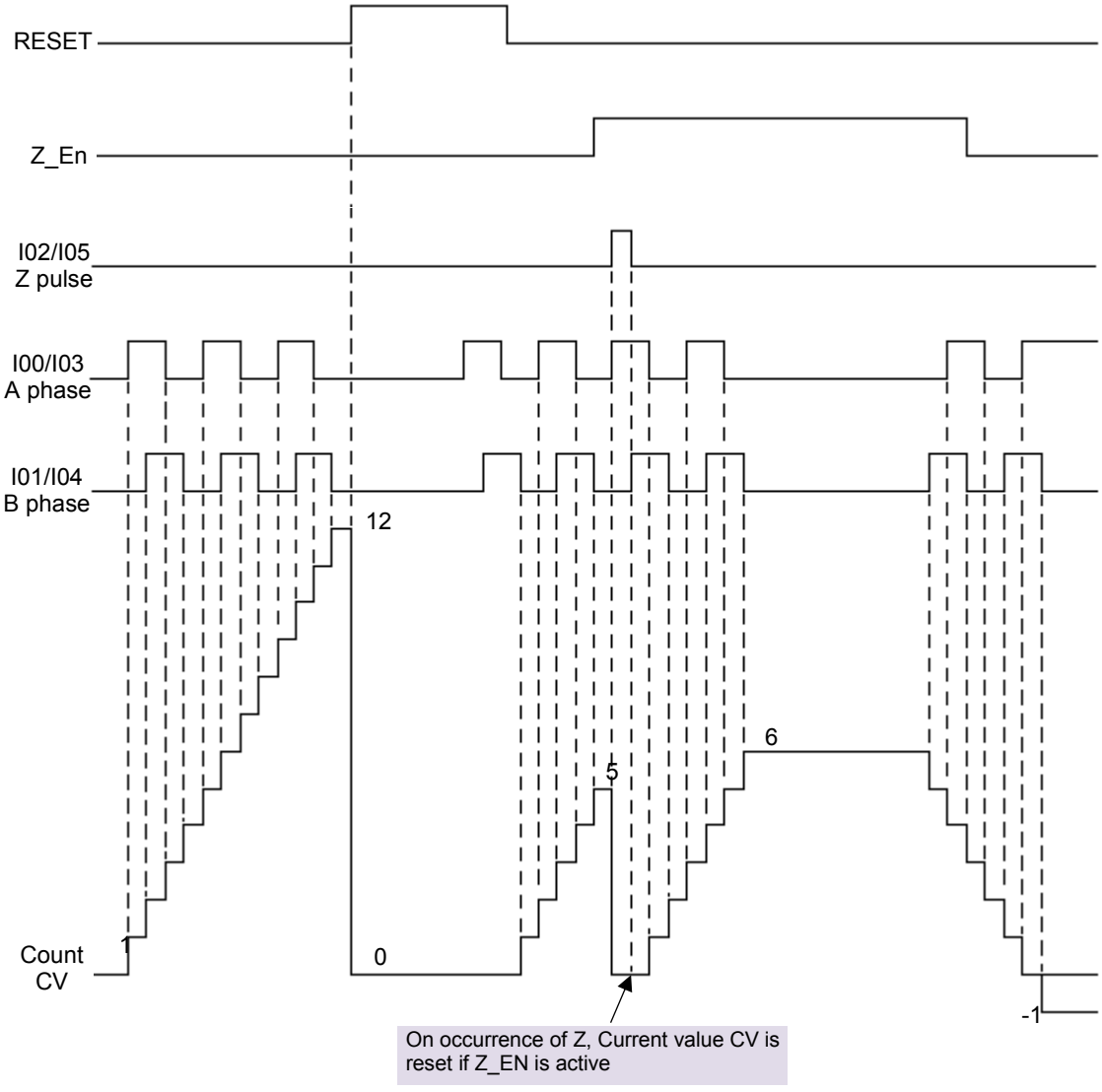


Figure 37: Functioning of ABZ encoder

If RUN is TRUE, counter starts counting. If RUN is FALSE, counter does not count and counter current value CV holds last value.

At rising edge (OFF to ON) as well as falling edge (ON to OFF) at A phase and at rising edge (OFF to ON) as well as falling edge (ON to OFF) at iB phase, count increments by1 if A phase is leading B phase.At rising edge (OFF to ON) as well as falling edge (ON to OFF) at A phase and at rising edge (OFF to ON) as well as falling edge (ON to OFF) at B phase,count decrements by 1 if A phase is lagging B phase.Counter current value is reset to 0 as long as Reset is TRUE.

If user enables Z action then counter current value gets reset to 0 on occurrence of Z pulse. It remains 0 as long as Z marker pulse in ON.

User can modify counter current value at any time and counter starts counting from modified value afterwards.

21.5 HSC Configuration and Programming

Refer section *High Speed Input Configuration* to understand configuration and programming of High speed counter functionality from Main unit.

Refer section *High Speed Input Configuration* to understand configuration and programming of High speed counter functionality from I/O Extension unit GC-4HSXPTY.

22 Interrupt Events

In the design of a real time system, the designer requires flexible and configurable control over the execution of different Program Organization Units (POUs) of the application program. Program type POU PLC_PRG is executed in freewheeling mode. Some POUs may get executed cyclically after predefined time interval. For some POUs, execution can be in response to the particular high priority interrupt event which is non-cyclic. Interrupt event can be periodic timer interval elapse, hardware input changing its state, high speed counter value matching the preset value, etc. Interrupt events require immediate action. It stops normal freewheeling execution called as PLC scan, performs specific POU execution, returns back to suspended point and resumes from the point where normal freewheeling execution was interrupted.

GOC supports several interrupt events through Main unit or I/O extension unit GC-4HSXPTY. In case of multiple interrupt events, execution needs to be prioritized. Such interrupt events are queued and executed one by one depending upon the priority.

22.1 Handling Interrupt Events

The interrupt events are configured, activated and controlled dynamically by various functions provided in GOC35 library. Each interrupt event has following configuration information associated with it:

1. PROGRAM type POU that is executed on the occurrence of the event.
2. Interrupt event priority.
3. Event related additional parameters in the system variables.
4. Other functions and function blocks related to interrupt event.

Further following control on execution of various events is possible via the application programs:

1. Hold the execution of any other event execution while handling critical tasks or accessing shared memory.
2. Dynamically attaching and detaching interrupt event.
3. Dynamically change the attached POU to change the control and sequence over the application as needed.

Following are the important points to remember while using interrupt events:

If any interrupt event is occurred, it is queued and kept pending in the following cases:

1. If high priority interrupt is under execution.
2. If maximum nesting depth (which is 4) of interrupt events is crossed. This is possible if more than 4 higher priority interrupt events are occurred one after another before completing execution of earliest interrupt event.
3. If interrupt events are hold in the application program.

The interrupt event occurred is lost and not executed in the following cases:

1. If pending interrupt events are more than 32.
2. If multiple interrupt events are generated from the same source before the execution.

The interrupt event execution is hold by the system in following cases:

1. During input and output scan.

2. During execution of function Refresh_In and Refresh_Out.
3. While accessing expansion units by executing functions like IM_Read, IM_Write, Read_Diagnostic, etc.

Hence, all such functions should be handled carefully in the application program to get fast response to the interrupt events. The subsequent sections discuss different types of interrupt events supported as below

- Periodic
- Counter compare
- Hardware input

22.2 Periodic interrupt event

GOC supports one periodic interrupt event; EVENT_PERIODIC. PLC automatically interrupts the currently executed program (normally program PLC_PRG) after every preset time interval (defined by _EVENT_WPERIOD) and jumps to execute assigned POU. After executing this POU, it returns back to the point previously left from.

Details are listed below in tabular form.

<table>
<tr><td rowspan="3">No. of periodic interrupt events</td><td colspan="2">1</td></tr>
<tr><td>Event number</td><td>ENUM</td></tr>
<tr><td>1</td><td>EVENT_PERIODIC</td></tr>
<tr><td rowspan="3">Time interval resolution</td><td>Resolution</td><td>_EVENT_BPERIODIC_RESOLUTION</td></tr>
<tr><td>1 ms</td><td>FALSE</td></tr>
<tr><td>0.5 ms</td><td>TRUE</td></tr>
<tr><td>Minimum time interval</td><td colspan="2">1 / 0.5 ms</td></tr>
<tr><td>Maximum time interval</td><td colspan="2">2500 / 1250 ms</td></tr>
<tr><td rowspan="2">Associated system variables</td><td colspan="2">_EVENT_WPERIOD</td></tr>
<tr><td colspan="2">_EVENT_BPERIODIC_RESOLUTION</td></tr>
</table>

Timing diagram below explains behaviour of periodic interrupt event.

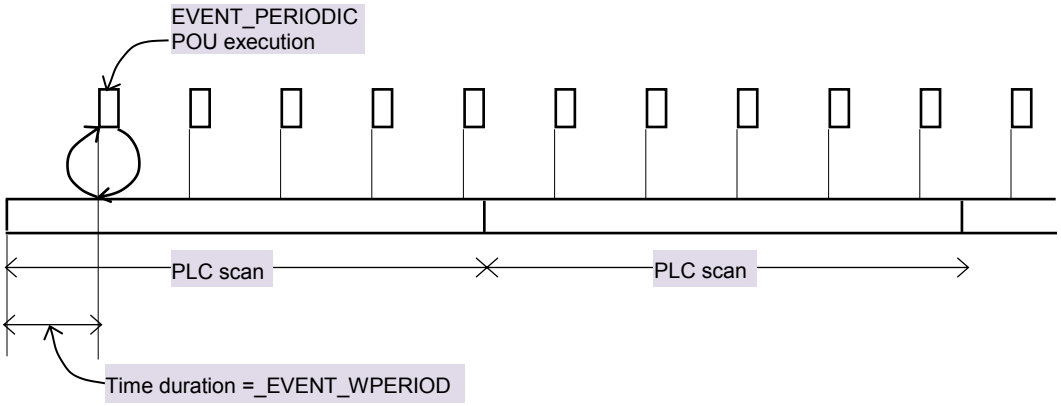


Figure 38: Periodic Interrupt event execution

When EVENT_PERIODIC is attached by executing function EVENT_ATTACH, preset is loaded with the value of _EVENT_WPERIOD and timer starts from 0. When time is equal to (_EVENT_WPERIOD x Resolution) msec, associated POU is executed. At the same time, time restarts from 0. This cycle continues as long as periodic interrupt event is not detached. If interrupt events are hold, then EVENT_PERIODIC is not executed but is queued. During hold, if multiple interrupt events of same source are occurred, only one occurrence is queued and others are lost. If the queue (of 32 events) is full, further interrupt events are lost.

22.3 Counter compare interrupt event

GOC supports 2 high speed counter / encoder interfaces either on Main unit or I/O extension unit GC-4HSXPTY. So it supports 2 counter compare interrupts. PLC interrupts the currently executed program (normally program PLC_PRG) whenever counter current value equals to the predefined value and jumps to execute assigned POU. After executing this POU, it returns back to the point previously left from.

Counter compare value can be changed at any time. It is necessary to ensure that the counter current value is not crossed the new comparison value before modifying it. Here, input pulse frequency, priority of HSC compare interrupt event, execution time of POU associated with that HSC compare event plays important role.

Details are listed below in tabular form.

<table>
<tr><td rowspan="8">No. of HSC compare interrupt events</td><td colspan="2">2</td></tr>
<tr><td colspan="2">Main unit</td></tr>
<tr><td>EVENT_HSC0_CMP</td><td>Applicable for counter/ encoder configured for
inputs I00, I01, I02</td></tr>
<tr><td>EVENT_HSC3_CMP</td><td>Applicable for counter/ encoder configured for
inputs I03, I04, I05</td></tr>
<tr><td colspan="2">I/O Extension unit GC-4HSXPTY</td></tr>
<tr><td>EVENT_HSC10_CMP</td><td>Applicable for counter/ encoder configured for
inputs I00, I01, I02</td></tr>
<tr><td>EVENT_HSC13_CMP</td><td>Applicable for counter/ encoder configured for
inputs I03, I04, I05</td></tr>
<tr><td>Time interval between successive interrupt events</td><td colspan="2">1 ms minimum (additionally depends upon assigned POU execution time)</td></tr>
</table>

22.4 Hardware input interrupt event

PLC interrupts the currently executed program (normally program PLC_PRG) whenever there is rising edge and/ or falling edge at high speed inputs I02 and I05 from Main unit or I/O extension unit GC-4HSXPTY and jumps to execute assigned POU. After executing this POU, it returns back to the point previously left from.

No. of hardware input interrupts (Rising edge and Falling edge)	4	
	Main unit	
	EVENT_IX2_R_TRIG	Interrupt event at rising edge at input I02
	EVENT_IX2_F_TRIG	Interrupt event at falling edge at input I02
	EVENT_IX5_R_TRIG	Interrupt event at rising edge at input I05
	EVENT_IX5_F_TRIG	Interrupt event at falling edge at input I05
	I/O Extension unit GC-4HSXPTY	
	EVENT_IX12_R_TRIG	Interrupt event at rising edge at input I02
	EVENT_IX12_F_TRIG	Interrupt event at falling edge at input I02
	EVENT_IX15_R_TRIG	Interrupt event at rising edge at input I05
	EVENT_IX15_F_TRIG	Interrupt event at falling edge at input I05
Pulse ON time	100 µs minimum for rising edge interrupt event	
	50 µs minimum for falling edge interrupt event	
Pulse OFF time	2 ms minimum	

User can have HSC functionality and Hardware input rising/ falling edge interrupt functionality either from Main unit (channel numbers 0/3) or I/O Extension unit (channel numbers 10/13) at a time.

When any functionality is configured on Main unit, user should not fix I/O Extension unit GC-4HSXPTY in IO1 slot. User is informed and care is taken during configuration using Hardware Configuration Tool.

22.5 Interrupt Event Configuration

Refer section *Interrupt Configuration* to understand configuration and programming of Interrupt functionality from Main unit.

Refer section *Interrupt Configuration* to understand configuration and programming of Interrupt functionality from I/O Extension unit GC-4HSXPTY.

23 microSD Memory

All the Main units are equipped with a microSD card slot on its right side wall (i.e. on the side of COM extension slot) when looking from the backside. Refer section 'Data Logging' for more information on usage of SD memory card.

23.1 SD Card

microSD cards with following specifications can be used

Item	Description
Type	Micro SD card
SD Card Standard	SDHC
Speed Class Supported	Class 4 (4MB/S) , Class 6 (6MB/S) , Class 10 (10MB/S)
Memory Capacity	4GB to 32GB *1
File System	FAT32 *2
SD card Dimentions	11 x 15 x 1.0 mm

Table 1 : SD card specifications

*1 microSD cards with memory capacity 4GB, 8GB, 16GB and 32GB and speed class 4, Class 6 and Class10 are tested successfully with GOC. GOC may not work with SD cards with other memory sizes. Do not use UHSI or UHS3 cards.

*2 microSD cards formatted to FAT32 using windows 7 (or above) should be used. SD cards with other formats may not work.

For optimum performance, make sure that SD card installed is not more than 80% full. Regularly check free space on SD card to avoid data loss. Make sure that entire SD card memory is available for GOC use and no unnecessary files are present on card.

Do not remove the microSD card before manually un-mounting SD card when Main unit is power is ON. Accidental removal of SD card may lead to corruption of files. Refer section 'System Menu' for un-mounting procedure.

Status and diagnostic information for microSD card is provided in 'System Variables' and in 'System Menu'. Refer respective chapters for more details.

In few situations, after accessing SD from PC, card may take few minutes time for mounting on first insertion.

Status and diagnostic information for microSD card is provided in 'System Variables' and in 'System Menu'. Refer respective chapters for more details.

While GOC is accessing to microSD card, do not remove the microSD card or power off the GOC. Failure to do so may cause microSD card failures or malfunctions. The file system on SD card may get corrupt in such cases.

Status of SD card access is provided in card status in system variable '`_SysvarSDCard.bCardStatus`'. Refer section 'System Variables' for more details.

It is advised to disable data logging before GOC power is switched off.

23.2 SD Card Folder Structure

All the data log files are stored in 'DataLogs' folder. If folder is not present, it is created automatically before writing log files. Folder is created only if data logging is configured in the application program.

If user stores other files, those will be kept as it is, however it is advised not to keep other files in SD card for faster and efficient access. PC based GOC File Explorer Tool is used to access data log files from microSD card.

23.3 Access to SD Card Data

There are two ways to access data logged in SD card

- Removing SD card from slot

  User can access SD card data by removing SD card. User should un-mount SD card using system menu before removal.

- Using PC based 'GOC File Explorer' tool

  GOC File Explorer' tool is provided in GOCToolkit to access files stored on SD card. Refer Appendix 1 for more details

Do not try to edit folders/ files created while accessing files from PC using card reader.

While retrieving files from SD card, copy them from SD card and paste them to local drive of your computer.

In few situations SD card may take several minutes of time for mounting on first insertion after formatting.

24 Data Logging

Data logging feature allows user to log PLC application variable values with time stamp in to microSD card in a CSV file format. User can retrieve these logged records by reading .CSV files by removing card and reading files on PC or using PC based software tool (GOC File Explorer) to upload files via communication interface. The logged data from CSV files can be further used for analysis, records and diagnostic purpose by user.

24.1 Specifications

Following table lists the functional specifications for data logging feature.

Item	Description
Number of data log groups	1
Number of data elements	32
Data Types	Bit, WORD, DWORD, INT, DINT, REAL, String (up to 80 characters)
Maximum size of log record	512 Bytes
Sampling method	• **Time Trigger** : Periodic time sampling • **Condition Trigger** : Application trigger based sampling
Minimum sampling time	• 500ms for log record size less than 256 bytes • 1 sec for log record size more than 256 bytes and less than 512 bytes
File format	.CSV
File switching	• Based on date (New file every day) • Based on number of log records • From PLC application
Maximum number of files	1000
Maximum file size	100MB Maximum 32000 log records in file.
File access	• By removing card from GOC and reading .CSV files directly in PC • By using PC based software tool (GOC File Explorer) to upload .CSV files via serial/ Ethernet communication interface without removing card from GOC. Refer Appendix1 for more details
Configuration	Using *HMI Configuration Tool*[*2]

[*1] Though user can generate condition trigger in alternate PLC scan, practically minimum sampling time depends on number of data elements, size of log record and PLC scan time. Writing to microSD card is relatively slow process compared to application program execution. Always make sure that you are not triggering data log records too frequently i.e. faster than minimum sampling time.

For data logging, internal buffer of 1024 bytes is reserved. Log records are stored in this buffer before writing to microSD card. Once buffer reaches 512 bytes, 512 bytes of data records are written to microSD card. This helps in reducing number of write cycles to microSD card. If internal buffer is filled with log records faster than SD card writing time, data loss may occur. This situation is indicated by bit 5 of system variable `'_SysVarSDCard.bCardStatus'`. Refer section 'System Variables' for more details.

[*2] Data logging configuration data is stored in app data memory along with multilingual objects data. Make sure that app data download is enabled. For more details refer section 'Application Program Memory'.

24.2 Before Configuration

User can configure data logging functionality using *Hardware Configuration Tool*. This section contains information related to data logging which should be known before starting configuration.

- Format of data log file
- Data log elements
- Data log trigger sources
- Naming format for data log file
- File switching conditions

Format of data log file

CSV file content has following format.

(1)	Date Time	,	Data*1* Header	,	Data*2* Header	,	…	,	Data*n* Header	CR	LF
	Year/Month/Day HH:MM:SS	,	Data	,	Data	,	…	,	Data	CR	LF
	Year/Month/Day HH:MM:SS	,	Data	,	Data	,	…	,	Data	CR	LF
	Year/Month/Day HH:MM:SS	,	Data	,	Data	,	…	,	Data	CR	LF
	Year/Month/Day HH:MM:SS	,	Data	,	Data	,	…	,	Data	CR	LF
		(2)		(3)	(4)					(5)	

No	Content	Description
1	Header row	This is header for all the columns in a file. User defined data names of the data elements logged in particular column are added in this row. Data header is added to log file at every program download and new file creation. User can enable/ disable inclusion of data header as per the application requirement.
2	Time stamp	Contains time stamp value. One of the following format can be selected. **None :** Null (No time stamp added) **YYYY/MM/DD HH:MM:SS :** 19 characters **DD/MM/YYYY HH:MM:SS :** 19 characters **HH:MM:SS :** 8 characters
3	**,**	Separator character. ‘**,**’ (coma) is a fixed as separator character. User cannot change it.

Format of data log file…

No	Content	Description
4	Data	Contains value of data element (PLC variable) at the time of log. **Bit :** 1 character (0 or 1) **WORD :** 5 characters max **DWORD :** 10 characters max **INT :** 6 characters max **DINT :** 11 characters max **REAL:** 14 characters max **STRING:** Depend on string size, up to 80 characters allowed *Note :* 1. Leading 0s are not included for numerals 2. '+' sign not included for positive integer values. 3. String is not added with null character.
5	CR LF	Return code at the line end

One line in data log file with set of time stamp, values of all the log elements and separator characters is referred as 'Log Record'. Size of single log record is limited to 512 bytes. *Hardware Configuration Tool* restricts user from exceeding the limit.

Data log elements

User can select any PLC variable of supported data type from application program as data log element.

Maximum 32 data log elements can be selected for a log record. Apart from variables declared by symbolic name in PLC application, user can set direct address from marker memory (%M), input memory (%I) and output memory (%Q) as data log element. Direct addressing is supported for data types BOOL, WORD and DWORD.

Table below shows supported data types, their display formats and maximum number of characters consumed.

Sr.No.	Data Type	Available Characters	Display Format	Max. Number of Characters
1	BOOL	0,1	#	1
2	WORD	0 to 9	#####	5
3	DWORD	0 to 9	##########	10
4	INT	-, 0 to 9	##### -#####	6
5	DINT	-, 0 to 9	########## -##########	11
6	REAL	0 to 9, E, +,-	#.#######E+## #.#######E-## -#.#######E+## -#.#######E-##	14
7	STRING	ASCII characters	Random ASCII string	1-80

Data log trigger sources

Two log trigger conditions are supported.

- **Time Trigger**
  - Triggers data log sampling at specified time interval.
  - Sampling time can be set between 500 ms to 32000 msec in steps of 100 msec or 1sec to 86400 sec in steps of 1sec.
- **Condition Trigger**
  - Triggers data log sampling if particular condition is met in user application.
  - User can trigger sampling of log data through system variable '_SysvarDataLog.bLogTrigger'.

User can enable both log trigger sources at a time. i.e. data is logged after specified time regularly and occasionally when specific condition is met.

Note that data logging functionality is executed by GOC at the end of every PLC scan. Hence there may be delay of maximum of one scan time .in processing of log trigger.

Writing to microSD card is relatively slow process compared to application program execution. Always make sure that you are not triggering data log records too fast i.e. faster than minimum sampling time specified. Triggering faster data logs may lead to data loss.

Naming formats for data log file

Data log files created are named as 'LOGNNNNN.CSV'. LOG is fixed prefix and NNNNN is file number from 00001 to 01000.

System variable '_SYSVARDATALOG.wLogFileNumber' holds file number and it is read/write type of variable. When condition for creating new log file is detected, file number is incremented by 1 and file is generated with incremented file number value. After reaching 1000 file number reset to 1.

User can modify file number to required value by directly writing to the system variable from the application program when data log is disabled. Writing of log file number when data logging is enabled is not allowed.

File switching conditions

User can configure condition for switching of log file as per the application requirement. User can enable one or more conditions from following 3 options:

- Switching of log file on day change i.e. midnight 00:00 AM.
- Switching of log file triggered from application program.

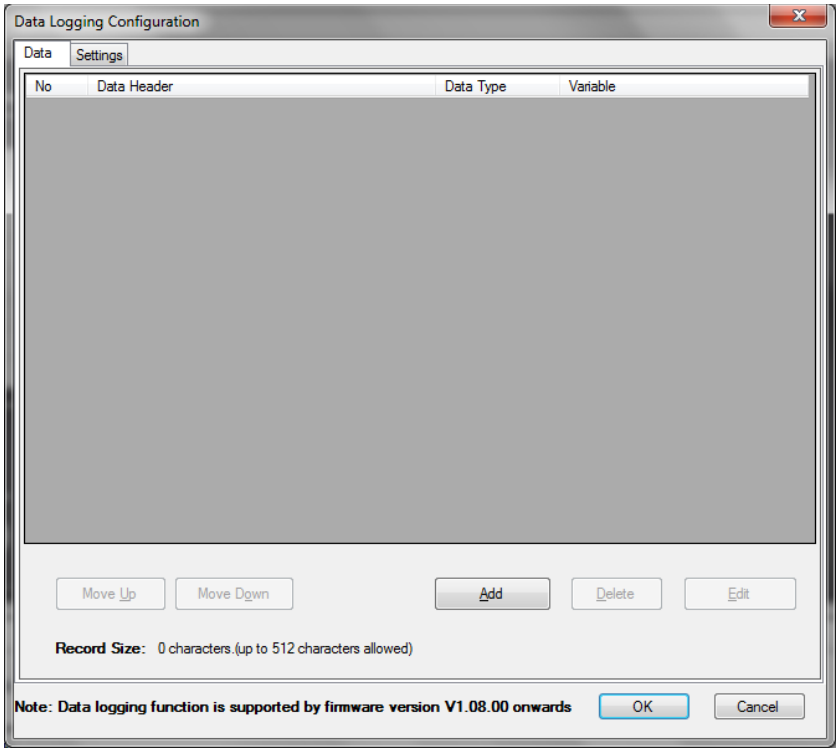
  Log file is switched when file switching trigger is generated from the application program via system variable '_SYSVARDATALOG.bCreateNewFile'.
- When number of records in a file reaches to maximum number of log records in file configured by user. By default this file switching condition is enabled with maximum number of log records equal to 32000. User cannot disable this file switching condition. However if required user can reduce number of maximum log records from 1000 to 32000.

After receiving file switching condition, new file is generated with incremented file number. If file with same name is already available on SD card, it will get deleted.

24.3 Configuration

HMI Configuration Tool allows user to configure data logging functionality as per the application requirement.

Menu command 'Tools → Data Logging' facilitates user configuration for data logging. Following dialogue appears for data logging configuration.

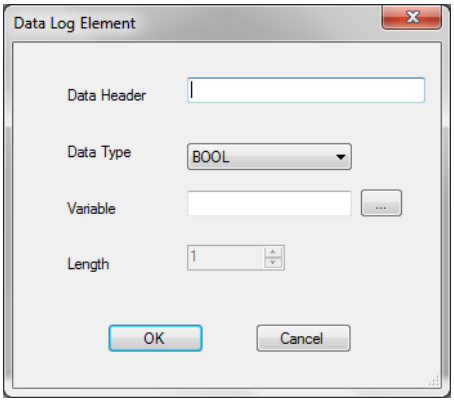


There are two tabs available in configuration dialogue; 'Data' and 'Settings'.

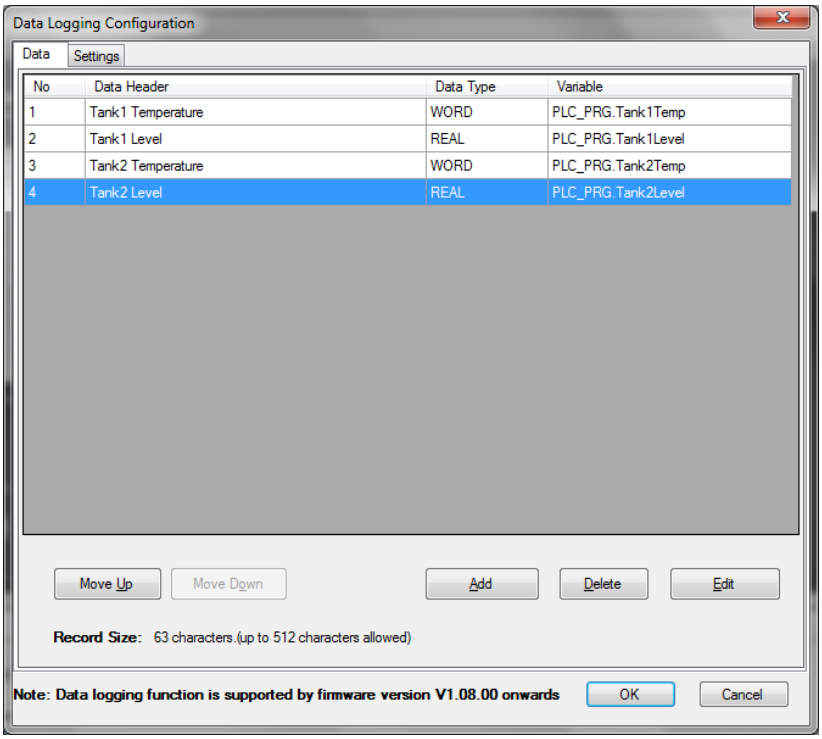
Data tab is used to manage log elements and 'Settings' tab allows user to set various configuration parameters related to log file content , log trigger source and file switching.

Data tab

Following buttons are provided to manage log element list

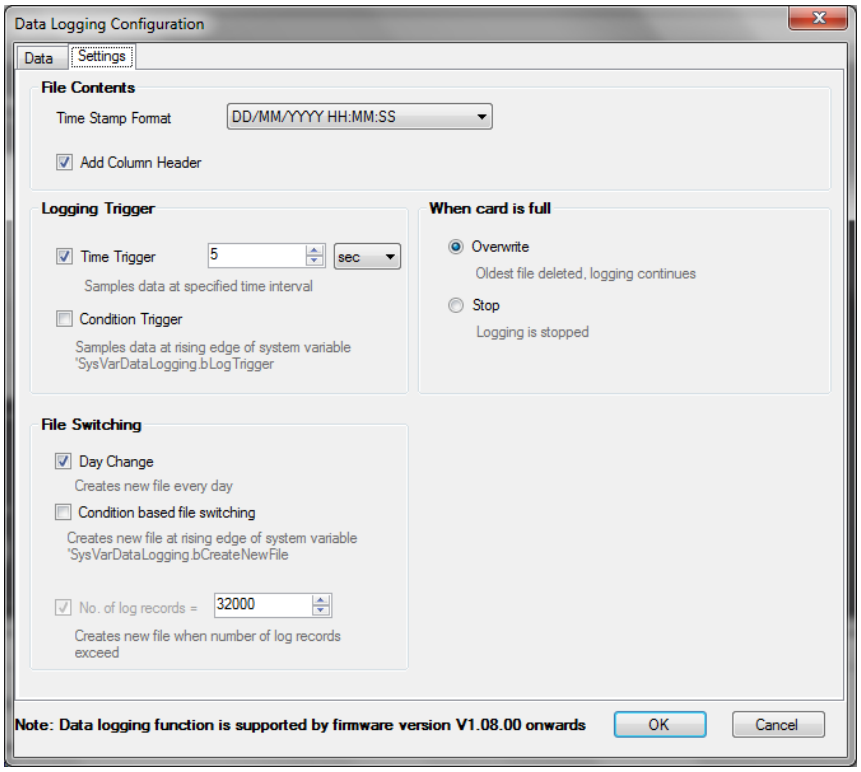
Button Name	Details
Add	Opens dialogue to add new data log element.  Define data header up to 32 character size. Select data type of log element variable from BOOL, WORD, DWORD, INT, DINT, REAL and STRING. Define associated controller variable. User can either select a variable (with symbolic name) from controller application program or use direct addressing e.g. %MX10.0 for BOOL type variable, %MW0 for WORD type variable, etc Length property defines number of characters consumed by log element. For the data types other than STRING, length is automatically set to maximum characters consumed by selected data type. For STRING data type, user can set length from 1 to 80 characters. Maximum 32 entries are allowed in the list.
Delete	Deletes selected entry from list.
Edit	Opens data log element dialogue for modification.
Up	Moves up selected entry in the list.
Down	Moves down selected entry in the list.

Following is the view of Data tab after adding log elements.



Settings tab

Following is the view of setting tab



Group Name	Setting	Details
File Contents	Time Stamp Format	Allows selection of time stamp in following formats None : No time stamp YYYY/MM/DD HH:MM:SS DD/MM/YYYY HH:MM:SS HH:MM:SS
	Add Column Header	Allows enabling / disabling inclusion of column header row in a log file.
Logging Trigger	Time Trigger	Enables periodic logging of data log elements after a predefined time. Time can be set from 500 ms to 32000msec in steps of 100msec. 1 Sec to 86400 Sec in steps of 1sec. *HMI Configuration Tool* will not allow setting of logging time less than 1sec when log record size is more than 256 character.
	Condition Trigger	Enables accepting data log trigger from application program. If Enabled, data is logged in on rising edge of system variable '`_SysVarDataLogging.bLogTrigger`'.

Setting tab

Group Name	Setting	Details
File Switching	Day Change	New file is created when day changes.
	Condition based file switching	New file is created on rising edge of a system variable '_SysvarDataLogging.bCreateNewFile'
	Number of log records	New file is created if number of log records in a file exceeds defined number.
When Card is full	Overwrite	When card full is detected, the oldest file on microSD card is deleted before creating a new file.
	Stop	Data logging is stopped when card full is detected.

24.4 Conditions of Loss of Logged Data

Following are few situations when there is possibility of loss of logged data.

Power off

When data logging in progress, GOC periodically performs file save operation after every 1 min. In case of power fail log data of last 1 minute before power fail may loss.

Log file is also saved (and closed) when data logging is disabled by user. If required log data file save can be forced by disabling data logging from application program.

Buffer full condition

To avoid frequent writing to SD card, logged data is stored in internal buffer of 1024 bytes size. Writing to SD card is relatively slow process. If log data generation is faster than SD card writing, then there is possibility of loss of log data. This situation occurs when very fast (faster than minimum sampling time) log records are generated using condition trigger from the application program. This condition is indicated using bit number 6 of system variable '_SysVarSDCard.bCardStatus'.

Source code download / upload

Data logging function will not work when source code upload or download activity in progress. Logging data will be lost.

While GOC accessing to microSD card, do not remove the microSD card or power off the GOC. Failure to do so may cause microSD card failures or malfunctions. The file system on SD card may get corrupt in such cases.

Status of SD card access is provided in card status in system variable '_SysvarSDCard.bCardStatus'. Refer section 'System Variables' for more details.

It is advised to disable data logging before GOC power off.

24.5 Effect on Scan Time

Using data logging

Data logging function execution time depends on the data logging configuration done by user. With typical data logging configuration of 15-20 log elements and sampling time of 5 sec, PLC scan time may increase by 7ms maximum.

File transfer

There will be additional increase in PLC scan time during file upload using 'GOC File Explorer' tool.

.

25 System Variables

The system variables are predefined global variables, which exchange the information between the CPU and the application program. Each system variable has a unique Name, which starts with underscore '_Sysvar'. System variables are categorized depending on functionality and presented in a predefined structure format.

The table below explains the significance of system variable structures based on functionality –

System Variable Structures	Description
_SysvarVersionInfo	Provides version of firmware and hardware of main unit
_SysvarCPU	Provides CPU specific status and diagnostic information
_SysvarRTC	Provides RTC data and status
_SysvarMemPtr	Provides start address and size of different types of memory blocks in the controller.
_SysvarHMI	Provides HMI function specific system variables
_SysvarInterrupt	Provides interrupt event specific system variables
_SysvarSDCard	Provides SD card status, card size , card free size etc.
_SysvarDataLog	Provides variables to monitor and control data logging function.

Individual structure variables can be accessed using dot (.) operator e.g. _SysvarVersionInfo.SRTSVERSION which holds firmware version.

The table below explains the significance of system variables category wise –

Name of System Variable	Data Type	Access	Description	
_SysvarVersionInfo				
.SRTSVERSION	STRING	Read only	String holds CPU firmware version.	
.SHWVERSION	STRING	Read only	String holds CPU (main unit) hardware version.	
_SysvarCPU				
.SYSTEMBITS	WORD	Read only	These are special bits available to the application program. Bit number and details are provided as below.	
			Bit	**Details**
			0	Bit remains on always.
			1	Bit is on for the first scan cycle. This bit can be used to call an initialization subroutine.
			2	Bit is ON in case of warm start initialization. It remains ON for 1 scan only.
			3	Bit is ON in case of cold start initialization. It remains ON for 1 scan only.
			4	Bit provides a clock pulse with ON OFF duration of 5 ms
			5	Bit provides a clock pulse with ON OFF duration of 50 ms.
			6	Bit provides a clock pulse with ON OFF duration of 500 ms.
			7	Bit provides a clock pulse with ON OFF duration of 30 sec.
			8	Bit toggles its status in consecutive scans.
			9 - 15	Reserved

System variables…

<table>
<tr><th>Name of System Variable</th><th>Data Type</th><th>Access</th><th colspan="3">Description</th></tr>
<tr><td colspan="6">_SysvarCPU</td></tr>
<tr><td rowspan="16">.WSTATUS</td><td rowspan="16">WORD</td><td rowspan="16">Read only</td><td colspan="3">Holds system status and the significance of individual bit is as explained below.</td></tr>
<tr><td>Bit</td><td>Status</td><td>Significance</td></tr>
<tr><td rowspan="2">0</td><td>TRUE</td><td>CPU in RUN mode</td></tr>
<tr><td>FALSE</td><td>CPU in STOP mode</td></tr>
<tr><td>1</td><td>--</td><td>Reserved</td></tr>
<tr><td rowspan="2">2</td><td>TRUE</td><td>Forcing of any PLC variable through programming software CoDeSys is active.</td></tr>
<tr><td>FALSE</td><td>No forcing of any PLC variable is carried out.</td></tr>
<tr><td rowspan="2">3</td><td>TRUE</td><td>I/O error.
One or more extension units registered at power ON is not communicating.
Probable cause for this fault is removal or hardware fault of extension unit or change in extension unit type after power on.</td></tr>
<tr><td>FALSE</td><td>No I/O error</td></tr>
<tr><td rowspan="2">4</td><td>TRUE</td><td>RTC Error. User should set RTC again.</td></tr>
<tr><td>FALSE</td><td>RTC value is valid as RTC back-up is healthy</td></tr>
<tr><td>5</td><td>--</td><td>Reserved</td></tr>
<tr><td rowspan="2">6</td><td>TRUE</td><td>One or more illuminated keys found pressed at power on. It may indicate fault in illuminated key hardware section.</td></tr>
<tr><td>FALSE</td><td>Any illuminated key doesn’t found pressed at power on.</td></tr>
<tr><td rowspan="2">7</td><td>TRUE</td><td>One or more HMI keys found pressed at power on. It may indicate fault in HMI key hardware section.</td></tr>
<tr><td>FALSE</td><td>Any HMI key doesn’t found pressed at power on.</td></tr>
<tr><td>8 - 15</td><td>--</td><td>Reserved</td></tr>
</table>

System variables…

<table>
<tr><th>Name of System Variable</th><th>Data Type</th><th>Access</th><th colspan="4">Description</th></tr>
<tr><td colspan="7">_SysvarCPU</td></tr>
<tr><td>.BCPUSTOPCAUSE</td><td>BYTE</td><td>Read only</td><td colspan="4">Indicates the reason for CPU to go in STOP mode. It is cleared when CPU goes to 'RUN' mode.</td></tr>
<tr><td></td><td></td><td></td><td>Bit</td><td>Significance</td><td>Details</td><td>Corrective Action</td></tr>
<tr><td></td><td></td><td></td><td>1</td><td>Memory Error</td><td>Invalid application program.
It is also indicated by LED indications.
Refer section ’13.1 LED Indications’ for more details.</td><td>Download a valid application program</td></tr>
<tr><td></td><td></td><td></td><td>2</td><td>Scan Error</td><td>Scan time in ms exceeds the value of _SYSVARCPU.WLIMITMAXSCAN.
It is also indicated by LED indications.
Refer section ’13.1 LED Indications’ for more details.</td><td>Find out cause of scan error (e.g. infinite loop) in application program and download a valid and corrected application program.</td></tr>
<tr><td></td><td></td><td></td><td>3</td><td>PFNMI Error</td><td>Low input power (< 18 VDC) to the controller main unit.
It is also indicated by LED indications.
Refer section ’13.1 LED Indications’ for more details.
In this case, controller doesn’t communicate with programming software CoDeSys and System Menu on LCD in not active.</td><td>Switch off the controller power and restore it again such that input power is > 18 VDC.</td></tr>
<tr><td></td><td></td><td></td><td>4</td><td>User Stop</td><td>User initiated STOP mode command through programming software CoDeSys.
It is also indicated by LED indications.
Refer section ’13.1 LED Indications’ for more details.</td><td>User should put CPU in RUN mode through programming software</td></tr>
<tr><td></td><td></td><td></td><td>5</td><td>Address error</td><td>Odd address is assigned to data type other than BOOL and BYTE.
It is also indicated by LED indications.
Refer section ’13.1 LED Indications’ for more details.</td><td>Find out the cause of invalid address assignment in application program and download a valid and corrected application program</td></tr>
</table>

System variables…

Name of System Variable	Data Type	Access	Description
_SysvarCPU			
.BINITSTATUS	BYTE	Read only	Holds the status of CPU initialization. This byte is updated whenever related action is executed.
			Hot Initialization _SysvarCPU.BINITSTATUS holds 1, if system detects a power break for less than 15 ms but greater than 10 ms. In this case controller functioning is normal as if there is no power disturbance.
			Warm Initialization _SysvarCPU.BINITSTATUS holds 2 on healthy power ON, if system detects a power break for more than 15 ms. It results resetting of data, which is not retained. CoDeSys menu command Online → Reset causes warm initialization
			Cold Initialization _SysvarCPU.BINITSTATUS holds 3, 1. If system detects any change in the application program. If a new application program is downloaded, cold start is observed. 2. Destroying of retentive data because of hardware fault. 3. Any related fault in power supply section 4. CoDeSys menu command Online →Reset Cold
.WCOLDSTARTCOUNTER	WORD	Read Only	Holds number of cold start initialization occurrences. This variable is persistent.
.WWARMSTARTCOUNTER	WORD	Read Only	Holds number of warm start initialization occurrences. This variable is persistent.
.WCURSCANTIME	WORD	Read Only	Holds scan time of last scan (in ms). The value is updated at the end of each scan.
.WMINSCANTIME	WORD	Read Only	Holds minimum scan time (in ms) in all previous PLC scans after power ON or warm or cold initialization. The value is updated at the end of each scan.
.WMAXSCANTIME	WORD	Read Only	Holds maximum scan time (in ms) in all previous PLC scans after power ON or warm or cold initialization with 1ms resolution.
.WLIMITMAXSCAN	WORD	Read Write	This is the maximum limit for scan time. If current scan exceeds this limit, CPU is put in STOP mode by declaring 'scan error'. It is also indicated by LED indications. Refer section '13.1 LED Indications' for more details. Default value is 250 ms and it is loaded before logic-scan. If user needs change in scan time limit, he can modify the value at the beginning of logic scan. Maximum scan time limit allowed is 1 sec. The new value is applicable for that scan only. Next scan is checked for default value unless this system variable is modified again to a new limit. If you want to change scan limit for every scan then you must write this variable in every scan.

System variables…

Name of System Variable	Data Type	Access	Description	
_SysvarCPU				
.W_REG_STATUS	WORD	Read only	Holds extension unit registration status as per the slots. The table below explains significance of bit depending upon hardware units and slot numbers.	
			Bit	**Details**
			0	Bit becomes TRUE, if CPU detects presence of main unit
			1	Bit becomes TRUE, if CPU detects presence of COM extension unit in COM1 slot
			2	Bit becomes TRUE, if CPU detects presence of COM extension unit in COM2 slot
			3,4	Reserved
			5	Bit becomes TRUE, if CPU detects presence of IO extension unit in IO1 slot
			6	Bit becomes TRUE, if CPU detects presence of IO extension unit in IO2 slot
			7 to 15	Reserved
.W_IOERR	WORD	Read only	Holds hardware units error status as per the slots. After power ON during initialization, CPU detects hardware units and starts I/O data exchange. Later on if CPU does not get response from extension unit for 2 sec duration due to physical disconnection/loose connection or hardware fault etc, then IO error is declared for that slot. Once IO error is declared it will not be cleared even if hardware is restored. I/O error is declared only for the hardware unit which is registered during controller power ON. IO Error is also indicated by LED indications. Refer section ’13.1 LED Indications’ for more details.	
			Bit	**Details**
			0	Reserved
			1	Bit becomes TRUE, if CPU detects IO error for COM extension slot COM1.
			2	Bit becomes TRUE, if CPU detects IO error for COM extension slot COM2.
			3,4	Reserved
			5	Bit becomes TRUE, if CPU detects IO error for IO extension slot IO1.
			6	Bit becomes TRUE, if CPU detects IO error for IO extension slot IO2.
			7 to 15	Reserved

System variables…

Name of System Variable	Data Type	Access	Description
_SysvarCPU			
.AMODULEORDERINGCODE	ARRAY [0..15] OF STRING	Read Only	This array holds ordering code of hardware units detected. _SysvarCPU.AMODULEORDERINGCODE[0]:Ordering code of Main unit _SysvarCPU.AMODULEORDERINGCODE[1]:Ordering code of COM Extension unit fixed in COM1 slot _SysvarCPU.AMODULEORDERINGCODE[2]:Ordering code of COM Extension unit fixed in COM2 slot _SysvarCPU.AMODULEORDERINGCODE[3]:Reserved _SysvarCPU.AMODULEORDERINGCODE[4]:Reserved _SysvarCPU.AMODULEORDERINGCODE[5]:Ordering code of IO Extension unit fixed in IO1 slot _SysvarCPU.AMODULEORDERINGCODE[6]:Ordering code of IO Extension unit fixed in IO2 slot _SysvarCPU.AMODULEORDERINGCODE[7]to _AMODULEORDERINGCODE[15]:Reserved
_SysvarRTC			
.BRTCINVALID	BYTE	Read only	RTC is backed up by super capacitor and back up time is 2 weeks nominal. This bit is TRUE if RTC is corrupted due to loss of back up due to any reason. RTC is then initialized to date of January 1, 1970 and time of 00H:00M:00S:00MS. In such case user has to ensure that super capacitor back up is restored and set proper value of RTC again. If user wants to use RTC data in application, programmer should consider this bit to check validity of RTC and inform user to take necessary action.
.BREADSECS	BYTE	Read only	Holds current “Seconds” value of RTC
.BREADMINS	BYTE	Read only	Holds current “Minutes” value of RTC
.BREADHRS	BYTE	Read only	Holds current “Hours” value of RTC
.BREADDATE	BYTE	Read only	Holds current “Date” value of RTC
.BREADMONTH	BYTE	Read only	Holds current “Month” value of RTC
.BREADYEARL	BYTE	Read only	Holds current “Year” value (lower byte)
.BREADYEARH	BYTE	Read only	Holds current “Year” value (higher byte)

System variables…

Name of System Variable	Data Type	Access	Description
_SysvarMemPtr			
.ASEGMENTPTR	ARRAY [0..5] OF DWORD	Read only	Holds start address of various memory blocks – _SysvarMemPtr.ASEGMENTPTR[0]: Start address of input memory _SysvarMemPtr.ASEGMENTPTR[1]: Start address of output memory _SysvarMemPtr.ASEGMENTPTR[2]: Start address of marker memory _SysvarMemPtr.ASEGMENTPTR[3]: Start address of data memory _SysvarMemPtr.ASEGMENTPTR[4]: Reserved _SysvarMemPtr.ASEGMENTPTR[5]: Start address of LCD data buffer
.ASEGMENTLEN	ARRAY [0..5] OF WORD	Read only	Holds size of memory blocks – _SysvarMemPtr.ASEGMENTLEN[0]: Size of input memory _SysvarMemPtr.ASEGMENTLEN[1]: Size of output memory _SysvarMemPtr.ASEGMENTLEN[2]: Size of marker memory _SysvarMemPtr.ASEGMENTLEN[3]: Size of data memory _SysvarMemPtr.ASEGMENTLEN[4]: Reserved _SysvarMemPtr.ASEGMENTLEN[5]: Size of LCD data buffer
.ADRRXPORT1	DWORD	Read only	This variable holds start address of receive buffer of COM1 slot. Buffer size is 256 bytes.
.ADRTXPORT1	DWORD	Read only	This variable holds start address of transmit buffer of COM1 slot. Buffer size is 256 bytes.
.ADRRXPORT2	DWORD	Read only	This variable holds start address of receive buffer of COM2 slot. Buffer size is 256 bytes.
.ADRTXPORT2	DWORD	Read only	This variable holds start address of transmit buffer of COM2 slot. Buffer size is 256 bytes.
_SysvarHMI			
.STW	WORD	Read write	Screen Trigger Word. _SysvarHMI.STW holds screen number to be displayed on LCD.
.PRESENTSCREENNUM	WORD	Read only	Read only variable. Reserved for internal use only.
.NEWSCREENNUM	WORD	Read only	
.SCREENSTATE	BYTE	Read only	
.REFRESHSCREEN	BOOL	Read only	

System variables…

<table>
<tr><th>Name of System Variable</th><th>Data Type</th><th>Access</th><th>Description</th></tr>
<tr><td colspan="4">_SysvarHMI</td></tr>
<tr><td>.WKEYSTATUS</td><td>WORD</td><td>Read only</td><td>This variable holds status of HMI keys. Bit becomes TRUE is respective key is pressed. It becomes FALSE if key is not pressed or key is released. HMI key status is updated in every input scan.
<table>
<tr><th>Bit</th><th>HMI Key</th><th>Bit</th><th>HMI Keys</th></tr>
<tr><td>0</td><td>Enter</td><td>8</td><td>F5</td></tr>
<tr><td>1</td><td>Next</td><td>9</td><td>F4</td></tr>
<tr><td>2</td><td>Esc</td><td>10</td><td>F3</td></tr>
<tr><td>3</td><td>Back</td><td>11</td><td>F2</td></tr>
<tr><td>4</td><td>Home/Shift</td><td>12</td><td>F1</td></tr>
<tr><td>5 to 7</td><td>Reserved</td><td>13 to 15</td><td>Reserved</td></tr>
</table></td></tr>
<tr><td>.WIKEYSTATUS</td><td>WORD</td><td>Read only</td><td>This variable holds status of illuminated keys. Bit becomes TRUE is respective illuminated key is pressed. It becomes FALSE if key is not pressed or key is released. Illuminated key status is updated in every input scan.
<table>
<tr><th>Bit</th><th>Illuminated Key</th><th>Bit</th><th>Illuminated Keys</th></tr>
<tr><td>0</td><td>K1</td><td>4</td><td>K5</td></tr>
<tr><td>1</td><td>K2</td><td>5</td><td>K6</td></tr>
<tr><td>2</td><td>K3</td><td>6</td><td>K7</td></tr>
<tr><td>3</td><td>K4</td><td>7</td><td>K8</td></tr>
<tr><td></td><td></td><td>8 to 15</td><td>Reserved</td></tr>
</table></td></tr>
</table>

System variables…

<table>
<tr><th>Name of System Variable</th><th>Data Type</th><th>Access</th><th>Description</th></tr>
<tr><td colspan="4">_SysvarHMI</td></tr>
<tr><td>.WIKEYLEDGREEN</td><td>WORD</td><td>Read only</td><td>Variable holds status of green LEDs provided in illuminated keys. Respective bit is TRUE when green LED glows and it becomes OFF when LED is off. Variable is updated by firmware based on value of LED control word defined in illuminated key configuration in ‘HMI Configuration Tool’
<table>
<tr><th>Bit</th><th>Green LED</th></tr>
<tr><td>0</td><td>K1</td></tr>
<tr><td>1</td><td>K2</td></tr>
<tr><td>2</td><td>K3</td></tr>
<tr><td>3</td><td>K4</td></tr>
</table>
<table>
<tr><th>Bit</th><th>Green LED</th></tr>
<tr><td>4</td><td>K5</td></tr>
<tr><td>5</td><td>K6</td></tr>
<tr><td>6</td><td>K7</td></tr>
<tr><td>7</td><td>K8</td></tr>
<tr><td>8 to 15</td><td>Reserved</td></tr>
</table></td></tr>
<tr><td>.WIKEYLEDRED</td><td>WORD</td><td>Read only</td><td>Variable holds status of red LEDs provided in illuminated keys. Respective bit is TRUE when red LED glows and it becomes OFF when LED is off. Variable is updated by firmware based on value of LED control word defined in illuminated key configuration in ‘HMI Configuration Tool’
<table>
<tr><th>Bit</th><th>Red LED</th></tr>
<tr><td>0</td><td>K1</td></tr>
<tr><td>1</td><td>K2</td></tr>
<tr><td>2</td><td>K3</td></tr>
<tr><td>3</td><td>K4</td></tr>
</table>
<table>
<tr><th>Bit</th><th>Red LED</th></tr>
<tr><td>4</td><td>K5</td></tr>
<tr><td>5</td><td>K6</td></tr>
<tr><td>6</td><td>K7</td></tr>
<tr><td>7</td><td>K8</td></tr>
<tr><td>8 to 15</td><td>Reserved</td></tr>
</table></td></tr>
</table>

Note: GOC front panel provided 8 keys with dual LEDs (Green and Red) behind it. When both LEDs glow, key is illuminated with yellow colored light.

System variables…

Name of System Variable	Data Type	Access	Description
_SysvarHMI			
.WKEYSSIMULATED	WORD	Read write	This variable is used to simulate the HMI keys through application program. User can modify status of HMI key to ON/OFF through application program without actually pressing it. Refer SysvarHMI.WKEYSTATUS description for bit to HMI key relation.
.BACKLITTIMEOUT	BYTE	Read write	Defines LCD backlit timeout in minutes. Default value is 30 minutes User can change value either between 20 to 40 or 0. If timeout set is 0, LCD backlit is on permanently.
.LCDCONTRAST	BYTE	Read write	Defines LCD contrast value. Default value is 75. Value should be in range 0 -100; 0: Minimum contrast, 100: Maximum contrast
.PSWLOGINLEVEL	BYTE	Read only	Holds user login password Level as 1 to 4. If user has not login any password level, then it holds 0.
.BUPPERCASE	BOOL	Read only	Read Only variable. Hold status of 'Λ' LED near home key.
_SysvarInterrupt			
.EVENT_WPERIOD	WORD	Read write	Holds time interval of periodic interrupt event. Resolution can be 1 ms or 0.5 ms. The value should be from 1 to 2500 corresponding to 1/0.5 ms to 2500/1250 ms.
.EVENT_BPERIODICRESOLUTION	BYTE	Read write	Decides resolution of time interval of periodic interrupt event. If FALSE, resolution is 1 ms. If TRUE, resolution is 0.5 ms
.EVENT_BCUR_EVENT	BYTE	Read only	Holds event number under execution.
.EVENT_BLOG_INDEX	BYTE	Read only	This byte holds index of array _SysvarInterrupt.EVENT_ALOG which is the event number recently executed.
.EVENT_ALOG	ARRAY [0..31] OF BYTE	Read only	This is array of 32 bytes which holds event number executed sequentially. The event number of executed events is logged in the array starting from _SysvarInterrupt.EVENT_ALOG[0] to_SysvarInterrup.EVENT_ALOG[31]. If array is full, executed events are logged again from _SysvarInterrupt.EVENT_ALOG[0]. The last event executed can be found using array index provided by system variable _SysvarInterrupt.EVENT_BLOG_INDEX
.EVENT_BLOST_COUNTER	ARRAY [0..255] OF BYTE	Read only	This is array of 256 bytes which corresponds to the interrupt event numbers from 0 to 255. Individual byte holds number of interrupt events lost. e.g. _SysvarInterrupt.EVENT_BLOST_COUNTER[1] holds number of periodic events lost.
.EVENT_PSYS	DWORD	Read only	For internal use.

System variables…

Name of System Variable	Data Type	Access	Description
_SysVarSDCard			
.BCARDREADY	BOOL	Read only	This bit provides card healthy status. **TRUE** : SD card is present, mounted, correctly formatted, not full and ready for operation **FALSE**: Any of above condition is not met.
.BCARDFULL	BOOL	Read only	Bit becomes TRUE if SD card memory is 90% full. If this bit is TRUE, as per user configuration either data logging will be stopped or GOC will over write older files.
.BCARDSTATUS	BYTE	Read only	Displays bitwise SD card status information

Bit	Details
0	**SD card detected.** Bit is TRUE when microSD card is detected.
1	**Mount / Unmount status** Bit is TRUE when microSD card is mounted.
2	**Invalid format** Bit is TRUE when format of microSD card is unknown. (other than FAT32)
3	TRUE : File Delete Operation In progress
4	**SD card busy** Bit is TRUE when Read /Write operation is in progress.
5	**SD card interface buffer full. Data loss may occur.** Since writing to microSD card is relatively slow operation, data log records are written to the internal buffer, before writing to microSD card. If buffer gets full, further data is lost. Application should monitor this bit for possible data loss while writing to microSD card. In such case reducing data logging frequency or number of log elements could be possible remedy.
6	Bit is TRUE if SD card mounting is in progress.
7	**SD card is 80% full** Bit is TRUE when microSD card memory is 80% full. This bit can be used in application program to generate prior intimation for operator before card is 100% full.

.WSIZE	WORD	Read only	Displays microSD card memory size in MB. e.g. for 4GB card, it displays 4096
.BFREESIZE	BYTE	Read only	Displays % of free memory on micoSD card.
.Unmount	BOOL	Read write	Unmounts SD card on rising edge of this BOOLean variable. After unmounting, SD card is mounted after power recycle or reinsertion of card.

System variables…

<table>
<tr><th>Name of System Variable</th><th>Data Type</th><th>Access</th><th colspan="2">Description</th></tr>
<tr><td colspan="5">_SysvarDataLog</td></tr>
<tr><td>.BLOGSTATUS</td><td>BYTE</td><td>Read only</td><td colspan="2">Byte variable displays status of data logging as</td></tr>
<tr><td></td><td></td><td></td><td>Value</td><td>Details</td></tr>
<tr><td></td><td></td><td></td><td>0</td><td>Data logging not configured by user.</td></tr>
<tr><td></td><td></td><td></td><td>1</td><td>Storage media not ready for writing. Refer _SysVarSDCard.BCARDSTATUS for the details of error. In this case, logging data is lost.</td></tr>
<tr><td></td><td></td><td></td><td>2</td><td>Invalid data logging configuration found. Data logging will not start in this case. Invalid data logging configuration in <CoDeSys project name>_APP.dat file.</td></tr>
<tr><td></td><td></td><td></td><td>3</td><td>Maximum size (100MB) of logging file reached. Logging stopped.
Logging stops in case of Custom file name format, if log file size reaches 100MB.</td></tr>
<tr><td></td><td></td><td></td><td>100</td><td>Logging is in progress without any error.</td></tr>
<tr><td>.BENABLELOGGING</td><td>BOOL</td><td>Read write</td><td colspan="2">This BOOL variable allows enabling of data logging from application program.
By default, this variable is set to TRUE at power ON. User can change value in application program afterwards for conditional Enable /Disable of data logging.</td></tr>
<tr><td>.BLOGTRIGGER</td><td>BOOL</td><td>Read write</td><td colspan="2">This BOOL variable accepts logging trigger from application program. On rising edge of this variable, data record is logged in file. Logging trigger is accepted only if ‘Condition Trigger’ is enabled.</td></tr>
<tr><td>.BCREATENEWFILE</td><td>BOOL</td><td>Read write</td><td colspan="2">This BOOL variable accepts file switching trigger from application program.
If file switching from application is enabled from configuration, then on rising edge of this variable, new log file is created in SD card. Name of new file depends on file name format selected by user as…</td></tr>
<tr><td></td><td></td><td></td><td>File Name format</td><td>New File Name</td></tr>
<tr><td></td><td></td><td></td><td>LOGNNNNN.csv</td><td>New file is created with incremented file number. e.g. If current file number is 100, then on receiving rising edge at ‘SysvarDataLog.BCREATENEWFILE’, new file is created with file name LOG00101.csv</td></tr>
<tr><td colspan="5">_SysvarDataLog</td></tr>
<tr><td>.DWLOGFILENUMBER</td><td>DWORD</td><td>Read write</td><td colspan="2">Log file number used to generate log file name when ‘LogNNNNN’ file name format is selected by user. File number is incremented by 1 every time new file is created.</td></tr>
<tr><td>.DWLOGRECNO</td><td>DWORD</td><td>Read only</td><td colspan="2">Holds count of number of records written in current file.</td></tr>
</table>

26 HMI Function

GOC provides built-in HMI functionality. It consists of 3 components.

- 128 x 64 pixel graphical LCD with backlit
- 10-key keppad
- Illuminated key panel

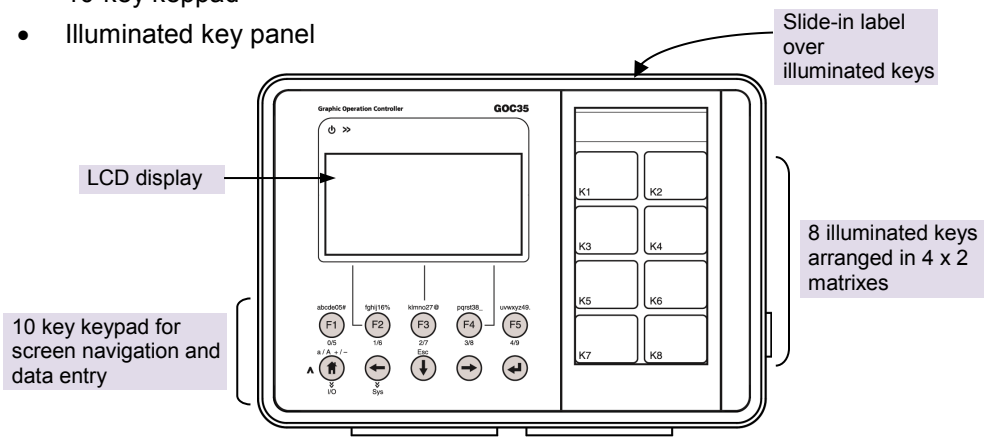


Figure 39: HMI components

Display and keypad is designed to add functionality and value to the micro range of controller. It provides the operator interface required for many small size applications. It has moderate display and setting capability and advanced features. However, it is limited if compared with a true HMI. Programmer should go through the features of PLC function and HMI capability to offer cost effective solution to many applications.

Out of 10 HMI keys, 5 keys can be configured for user functionality like momentary action, toggle action, Go to screen action. User can also define scope of key functionality as local to a screen and global. This helps in assigning multiple functionalities to a single key and enhances operation input and control capability.

Illuminated key panel along with slide-in label enhance usability. This part consists of 8 keys with dual colour (red, green) LEDs. Dual colour LEDs offer tri-colour effect with colors as red, green and yellow (red + green). Keys can be configured for operation like momentary, toggle and Go to screen. LEDs can be switched on/off through application.

Unit is dispatched with default label with key nomenclature as K1 to K2. Label can be designed and printed by customers especially OEMs. Label combined with illuminated keys helps in customizing controller to suit application. It can be used as operator station with push buttons and lamps, alarm annunciation and machine/process mimic.

26.1 Highlighting Features

HMI Function	
Built in LCD Display	▪ 128 x 64 pixel graphics LCD with backlit ▪ White characters with blue background ▪ View size : 70.7 x 38.8 mm ▪ 10 keys for display navigation, data entry and user settings - 5 user configurable function keys with local/ global operation ▪ 8 illuminated keys; keys with dual colour (Green, Red) LEDs ▪ Useful for monitoring and setting PLC data, display machine status and fault status
User Defined HMI Screens	▪ Up to 64 screens ▪ Programmable screen actions (Entry, Display, Exit), screen chaining ▪ 5 fonts, single size, double size, normal/bold, normal/ Inverse selection, blinking option ▪ Support of Static Text and Dynamic (ASCII) Text with inclusion of special characters ▪ Screen objects like Static Text, ASCII, Numeric, Bit, Time, Date, Time of Day, RTC ▪ Monitor/ modify PLC data with different data types like - BOOL (TRUE/FALSE text and graphical symbol) - BYTE, WORD, DWORD, INT, DINT - TIME is seconds and TIME is minutes - TIME OF DAY (TOD), - DATE and DATE_AND_TIME(DT) - RTC TIME and RTC DATE ▪ User defined PLC data attributes - number of digits to be displayed - decimal point position - display format - maximum and minimum limit - allow modification selectively ▪ Alphanumeric data entry ▪ Direct access of PLC variables with symbolic name

Highlighting Features…

HMI Function	
Advanced HMI Features	▪ Graphical symbols (32 x 32 pixel size and 48 x 48 pixel size) ▪ Multi-language support - 3 languages, local and foreign. - Provides all the fonts supported by Windows ▪ Custom image import to show logo, special symbol on LCD - Can import up to 8 images - Consumes data memory ▪ Screen level password - 4 digits numerical password to restrict any unauthorized access to critical parameter settings / screen. - 4 password levels from 1 to 4 with 4 as highest priority level. ▪ Configuration of data logging in SD memory card - Selection of up to 32 log variables - Defining logging interval - Define user controlled log trigger and file switching
Function Keys	▪ 5 HMI keys F1 to F5 ▪ User configurable operation Momentary/ Toggle/ GoTo Screen ▪ Local/ Global operation ▪ Key legend on LCD for F2, F3, F4
Built-in Status and Diagnostics	▪ One Key I/O Monitor ▪ System Menu for - System Status like scan time, current password level logged in, firmware version - Monitoring Input and Output status of Main unit and Extension units - System Diagnostics like CPU status, extension unit status, RTC error, etc - System Settings like RTC, IP Config, LCD settings like backlit timeout and contrast, Multi language selection - To log in, log out and modify password
Integrated Tool with User Friendly GUI *HMI Configuration Tool* [*1]	▪ Assign screen actions (Entry, Display, Exit) ▪ Define screen chaining and navigate using **Back** and **Next** key ▪ Configurable operation for function keys ▪ Configurable operation for illuminated keys and control of dual-colour LEDs ▪ HMI screen definition and key configuration is a part of application

[*1] Even though, maximum 64 user defined screens are allowed, it is limited by application program memory (192 Kbytes) and POU size (8192 bytes) because display related logic i.e. HMI screen definition and key configuration is a part of application program and is available in POU HMI_PRG.

26.2 Screens

HMI function in GOC supports up to 64 user screens where in user can monitor and set application data and show machine/ process status and alarm messages.

HMI screens can be designed and keys can be configured using 'HMI Configuration Tool'. Tool facilitates Fast screen definition with drag n drop objects. Based on user design and configuration, 'HMI Configuration Tool' creates PRG HMI_PRG in PLC application program. So HMI application is part of PLC application program and consumes application program memory which has size of 192 Kbytes. For more details, refer Help of HMI Configuration Tool.

HMI can handle PLC data of different data types. User can monitor/ modify timer delays, counters, temperatures, speed of the machine, production count, machine running hours and various machine/ process parameters. User can select mode of operations like auto/ manual selection, start/ stop any sequence, enable/ disable machine accessories. User can schedule and control activities day wise, week wise dependent on built-in Real Time Clock.

Refer chapter '*Screen Objects*' for list of supported screen objects by GOC.

Screen updation time is 100 ms minimum.

26.2.1 Screen Navigation

After power ON, LCD displays 'Home' screen on LCD display.

Navigation / jump to another screen can be defined in two ways in HMI application.

- Navigation using NEXT key:
  For this set next screen using screen property 'Chain To'. Refer chapter '*Screen Chaining*' for more details.
- Navigation using function keys or illuminated keys
  For this configure function key for GO TO functionality. Refer chapters '*Function Keys*' and '*Illuminated keys*' for more details

User can return to previous screen by pressing 'PREV' key. GOC maintains log of 32 last screens which can be accessed one by one by pressing PREV key. Previous screen log is reset when HOME screen is displayed on LCD.

User can always return to home screen (screen 0) by pressing HOME key.

Triggering HMI screen from PLC application program. Function 'DisplayScreen' can be used in PLC application program to trigger particular screen on LCD display.

Refer *ED-2002-541_GOC35 Util_Library Manual* for more details.

26.2.2 Modes of Operation

HMI LCD display operates in 2 modes; display mode and modification mode. Functionality of HMI keys depend upon mode of operation.

Display mode

By default, HMI is in display mode i.e. user is navigating through user screens or system menu using keys and monitoring it. Function keys F1 to F5 acts as per user configuration e.g. Momentary, Toggle, GoTo Screen and behavior depends upon local and global configuration. Refer section *HMI Keys* for more details of function keys

Modification mode

HMI LCD display is switched to modification mode when user wants to modify any screen object or system parameter on a screen.

Transition from display mode to modification mode takes place when

- User presses **Enter** key with line cursor displayed at modifiable object in a screen.

Transition from modification mode to display mode takes place when

- **Enter** key is pressed to validate data entry. Exits modification mode on valid entry.
- **Esc** key pressed to exit modification mode without data entry.
- Key timeout is occurred i.e. any HMI key is not pressed for 30 sec duration.

HMI key operation depends upon mode of operation as explained in next section.

26.3 HMI Keys

HMI key pad is located below LCD display as shown below.

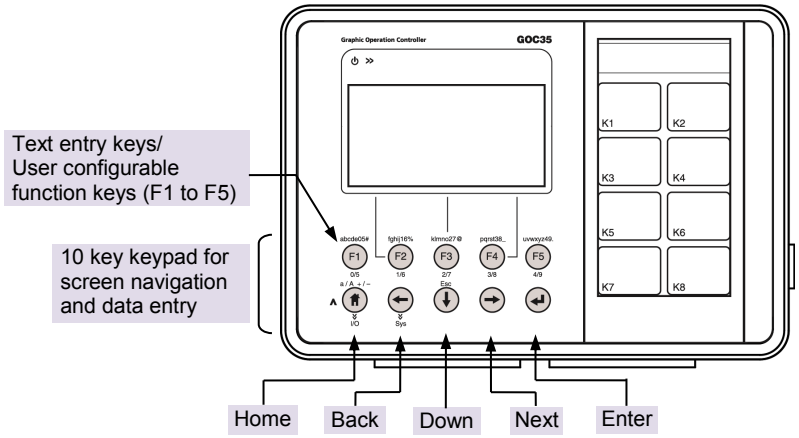


Figure 40: HMI keypad

HMI keypad provides 10 keys arranged in 2 rows for screen navigation and data entry.

User cannot access status of the keys namely **Home**, **Back**, **Down**, **Next** and **Enter** (placed in second row) through application program.

Keys **F1**, **F2**, **F3**, **F4** and **F5** (placed in the first row) are useful to enter numbers (0 to 9) and text characters. These keys can be configured as function keys when display is not in modification mode. For more details, refer section xxx Function Keys.

There are two modes of a key pressing as explained in the table below.

Short press	Press key for less than 2 sec duration. Desired action is performed after key release i.e. at falling edge of key short press.
Long press	Press key for more than 2 sec duration. Desired action is performed when key is kept pressed and 2 sec duration is elapsed i.e. at rising edge of key long press.

All these keys have multiple functionalities depending on display mode of operation as explained in section *Modes of Operation*. The table below explains key operation in display mode and modification mode.

Display Mode	Key	Modification Mode
User Home At any user screen (screen number 0 to 63), press **Home** key to switch to User Home screen (screen number 0)	Home	**Shift** Acts as **Shift** key. Selects lower (a, b, c…) or upper case (A, B, C….) during text object modification. Selects sign (+/-) during numeric (INT, DINT data types) object setting. Green LED is associated with **Shift** key to indicate selection. It glows in case of upper case selection and negative sign selection.
I/O Monitor At any user screen, long press (press key for 2 sec duration) switches screen to I/O Monitor screen. I/O monitor screen shows the status (0/1) of digital input output of Main unit and Extension units. At I/O Monitor screen, if key is pressed, it returns to user screen as per STW.		
System Home At any system menu screen, press **Home** key to switch to System Home screen.		

HMI key operations…

Display Mode	Key	Modification Mode
Previous Screen Switches to previously displayed screen. Display keeps a log of up to 32 last screens. Log is cleared at User Home or System Menu Home screen.	Back 	**Previous digit/ character** It moves cursor to left. Selects previous digit/ character of data object under modification.
Jump To System Menu Long press switches screen from user screen to system menu home screen and vice versa.		**Delete Character** Long press deletes current character during text entry.
Next Object Cursor shifts to next modifiable object on screen. Sequence of navigation for modifiable objects is programmable.	Dn 	**Escape** Exit modification mode, discarding change in the value of the object under modification.
Next Screen Jumps to user defined next screen.	Next 	**Next Character** Selects next character of data object under modification. It moves cursor to right.
		Insert Space Long press inserts space at active cursor position during text entry.
Enter Enters into modification mode for the object selected i.e. .object with line cursor. After entering modification mode, blinking block cursor '▉' appears at first digit / character of the object.	Enter 	**Validate** Validates data entry and modifies value of data object under modification. If modified value is valid, it exits modification mode for selected object. For modified value is not valid, error message is displayed discarding modification carried out. It remains in modification mode with previous valid value and allows user to enter new valid value.

Table on next page provides details of function keys F1 to F5.

HMI key operations…

Display Mode	Key	Modification Mode
Function key F1 Acts as per user configuration (Momentary/ Toggle/ Go to other screen). User can define scope of operation as local to a screen and global.	F1	**Data Entry** For numeric object, key selects **0**/ **5** alternately with quick pressing for numeric objects. For Boolean object, key selects **TRUE**/ **FALSE** alternately with quick pressing. Selects text characters **a**/ **b**/ **c**/ **d**/ **e**/ **0**/ **5**/ **#** with quick pressing in succession for text objects. If Home/Shift selects upper/lower case for alphabets. Glowing green LED indicates upper case alphabet entry.
Function key F2 Acts as per user configuration (Momentary/ Toggle/ Go to other screen). User can define scope of operation as local to a screen and global. User can define key legend (6 characters) to be displayed on LCD bottom row.	F2	**Data Entry** For numeric object, key selects **1**/ **5** alternately with quick pressing for numeric objects. Selects text characters **f**/ **g**/ **h**/ **i**/ **j**/ **0**/ **5**/ **%** with quick pressing in succession for text objects. If Home/Shift selects upper/lower case for alphabets. Glowing green LED indicates upper case alphabet entry.
Function key F3 Acts as per user configuration (Momentary/ Toggle/ Go to other screen). User can define scope of operation as local to a screen and global. User can define key legend (6 characters) to be displayed on LCD bottom row.	F3	**Data Entry** For numeric object, key selects **2**/ **6** alternately with quick pressing for numeric objects. Selects text characters **k**/ **l**/ **m**/ **n**/ **o**/ **2**/ **6**/**@** with quick pressing in succession for text objects. If Home/Shift selects upper/lower case for alphabets. Glowing green LED indicates upper case alphabet entry.
Function key F4 Acts as per user configuration (Momentary/ Toggle/ Go to other screen). User can define scope of operation as local to a screen and global. User can define key legend (6 characters) to be displayed on LCD bottom row.	F4	**Data Entry** For numeric object, key selects **3**/ **7** alternately with quick pressing for numeric objects. Selects text characters **p**/ **q**/ **r**/ **s**/ **t**/ **3**/ **7**/ **_** with quick pressing in succession for text objects. If Home/Shift selects upper/lower case for alphabets. Glowing green LED indicates upper case alphabet entry.
Function key F5 Acts as per user configuration (Momentary/ Toggle/ Go to other screen). User can define scope of operation as local to a screen and global.	F5	**Data Entry** For numeric object, key selects **4**/ **8** alternately with quick pressing for numeric objects. Selects text characters **u**/ **v**/ **w**/ **x**/ **y**/ **z**/ **4**/ **8**/ **.** with quick pressing in succession for text objects. If Home/Shift selects upper/lower case for alphabets. Glowing green LED indicates upper case alphabet entry.

Points to note

1. When user is monitoring System Menu, change in Screen Trigger Word; (`_SysvarHMI.STW`) via application program is ignored. To monitor screens triggered by application program, user has to switch to User Home first.
2. Modification mode is exit automatically, if no key operation is carried out for 30 sec duration.
3. During password entry and password modification, selected digit is displayed and all other digits are masked by ✷.
4. F1 to F5 keys are useful for data entry.

   During text (alphanumeric character) entry, F1 key behaves as explained below when it is pressed in succession. Behavior is explained considering lower case alphabets i.e. Shift key is not pressed. F2 to F5 keys operation is identical but alphanumeric characters are different.

   F1 key operation for text entry

First key press	alphabet '**a**' is inserted at cursor position.
Second key press (within <1 sec time)	alphabet '**b**' is inserted at cursor position.
⋮	⋮
⋮	⋮
Eights key press (within <1 sec time)	character '**#**' gets inserted at cursor position.
Ninth key press (within <1 sec time)	it gets rollover again to insert alphabet '**a**' at cursor position
Key not pressed within 1 sec	cursor moves to next character (on left side) automatically.

   During numeric (0 to 9, sign +/-) entry, F1 key behaves as explained below when it is pressed in succession. On pressing F1 key in succession, it inserts 1 and 6 alternately. F2 to F5 keys operation is identical but numbers are different.

   F1 key operation for numeric entry

First key press	digit **1** is inserted at cursor position.
Second key press (within <1 sec time)	digit **6** is inserted at cursor position.
Third key press (within <1 sec time)	digit **1** is inserted at cursor position.
Third key press (within <1 sec time)	digit **6** is inserted at cursor position.
Key not pressed within 1 sec	cursor moves to next digit (on left side) automatically.

26.4 Data Entry

This section explains how to modify data (PLC variable value) using LCD screen and HMI keys. Refer sections *Overview* and *HMI Keys* to understand basic navigation and functionality of HMI keys. Subsequent sections explain modification of INT data and ASCII data.

26.4.1 Modifying INT data

After displaying screen on LCD, blinking line cursor below first number of variable value indicates that modification is allowed for that variable. If multiple modifiable variables are present on screen, press 'ESC' key to move cursor to desired variable. Press 'ENT' key to enter in to modification mode.

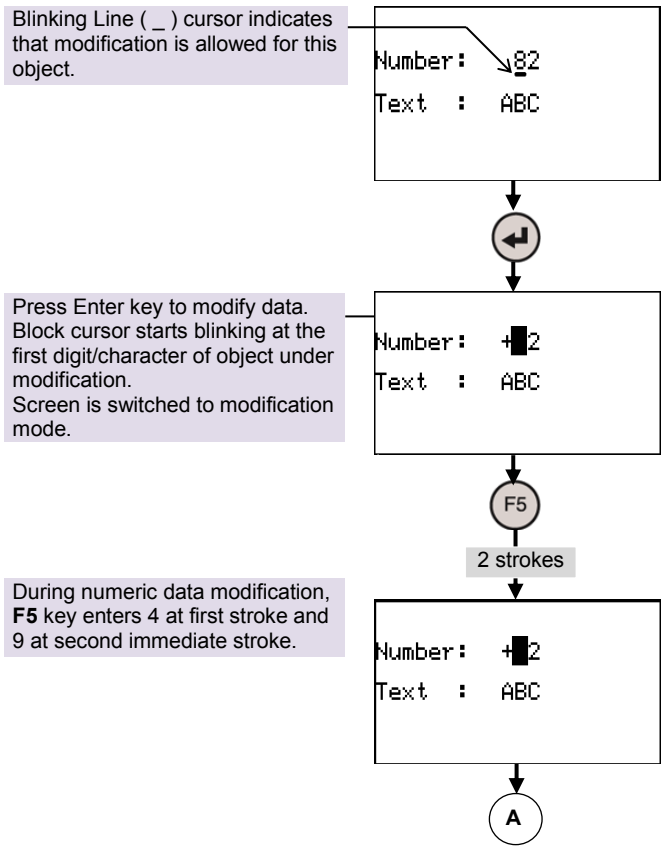
When making numeric entry, press a function key (F1 to F5) marked with the desired number repeatedly until the number appears. Once you stop pressing key, the cursor automatically shifts to next digit of value display.

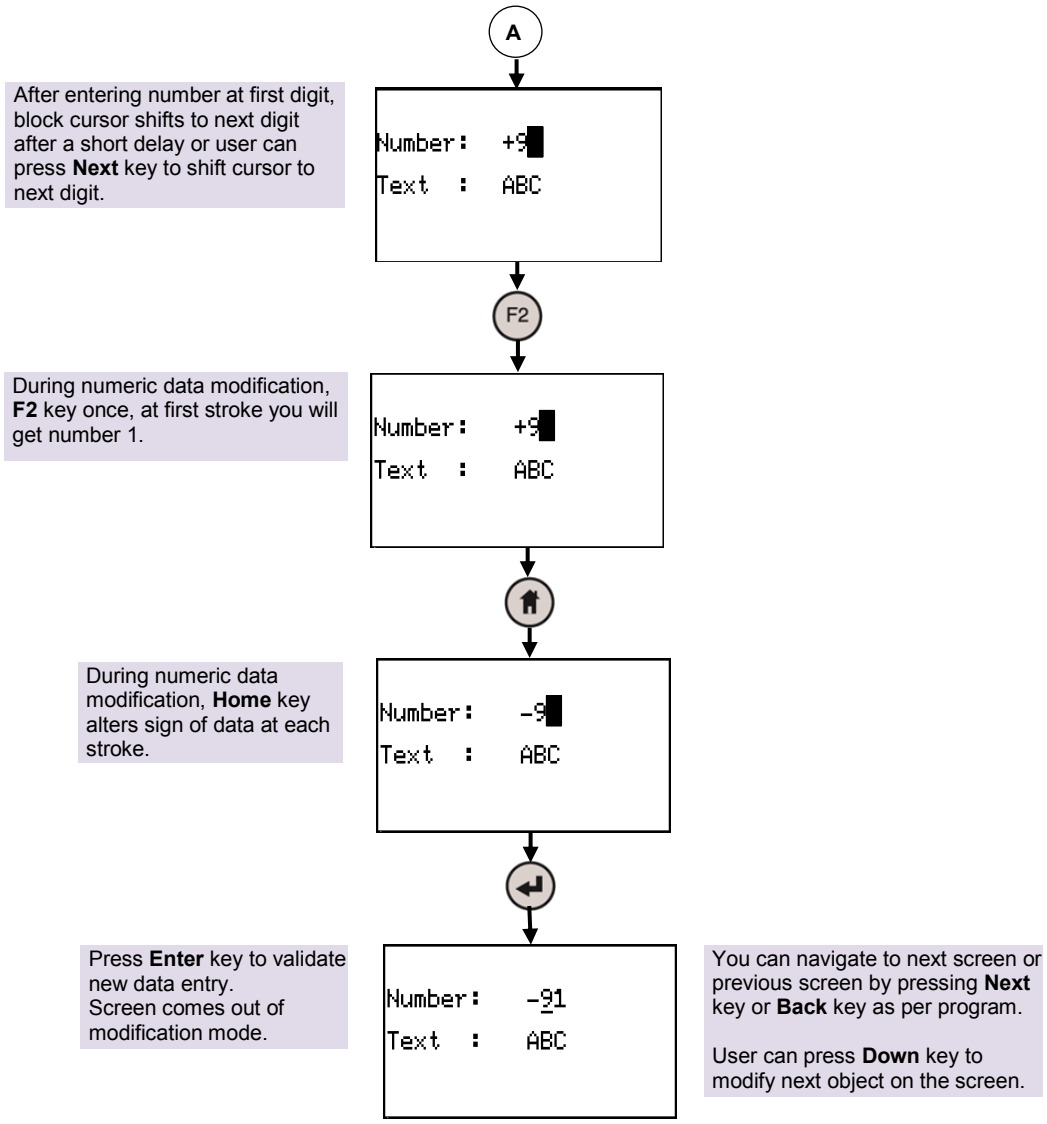
You can use following functions to edit the value

- Use PREV key to move cursor left
- Use NEXT key to move cursor right
- Use HOME key to change sign of entry.

Example

Flowchart below explains how to modify integer data from 82 to -91 for an example. Screen shown below is programmed using *HMI Configuration Tool*. As per user programming, navigate through the screens to display screen shown below.





During data modification for any object (showing blinking block cursor at any digit), user can press **Down** key to come out of modification ignoring entry of new value. This will show line cursor blinking at first digit of the object under modification.

26.4.2 Modifying ASCII text data

After displaying screen on LCD, blinking line cursor below first digit of variable value indicates that modification is allowed for that variable. If multiple modifiable variables are present on screen, press 'ESC' key to move cursor to desired variable. Press 'ENT' key to enter in to modification mode.

When making alphanumeric entry, press a function key (F1 to F5) marked with the desired character repeatedly until the character appears. Once you stop pressing key, the cursor automatically shifts to next digit of value display.

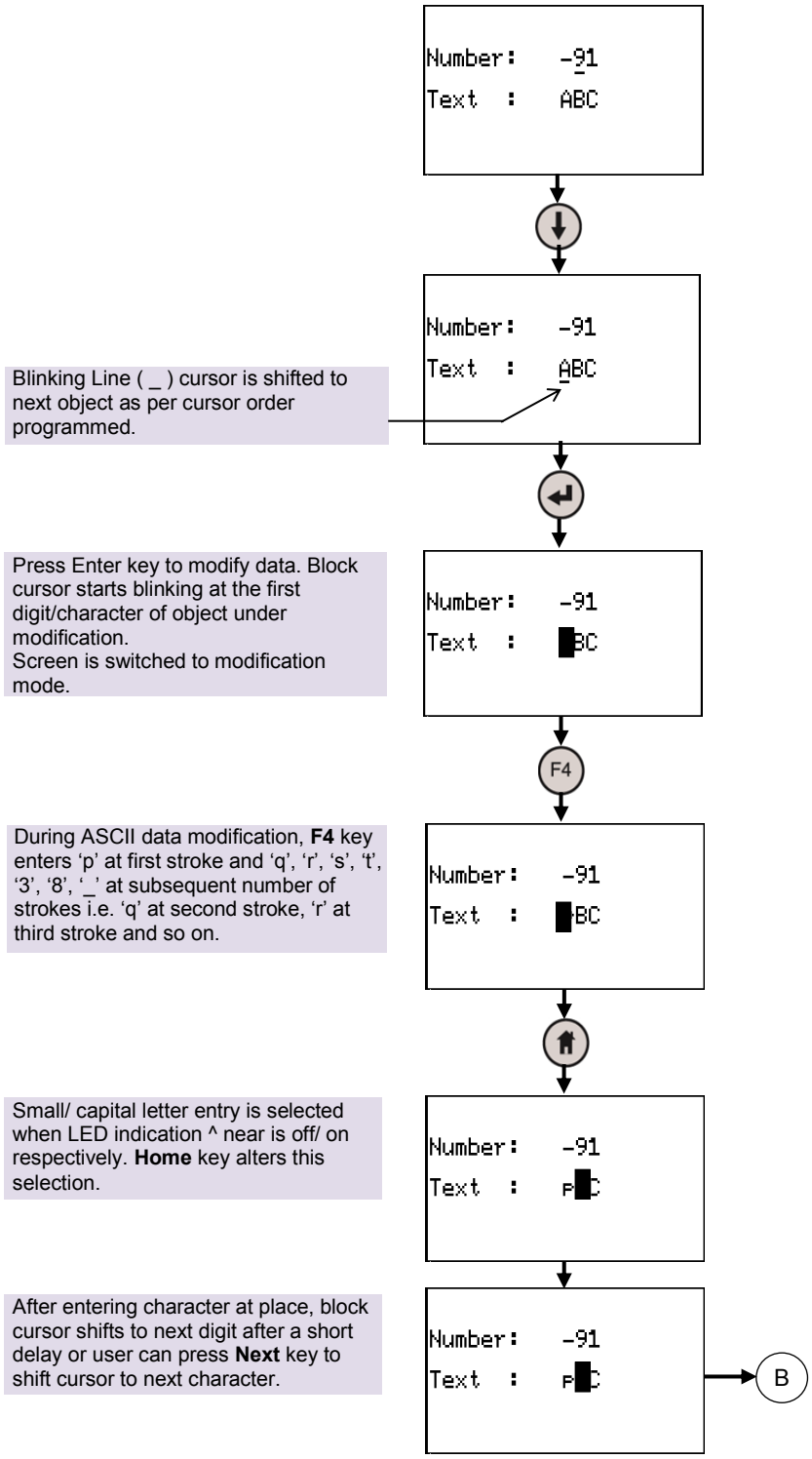
You can use following functions to edit the value

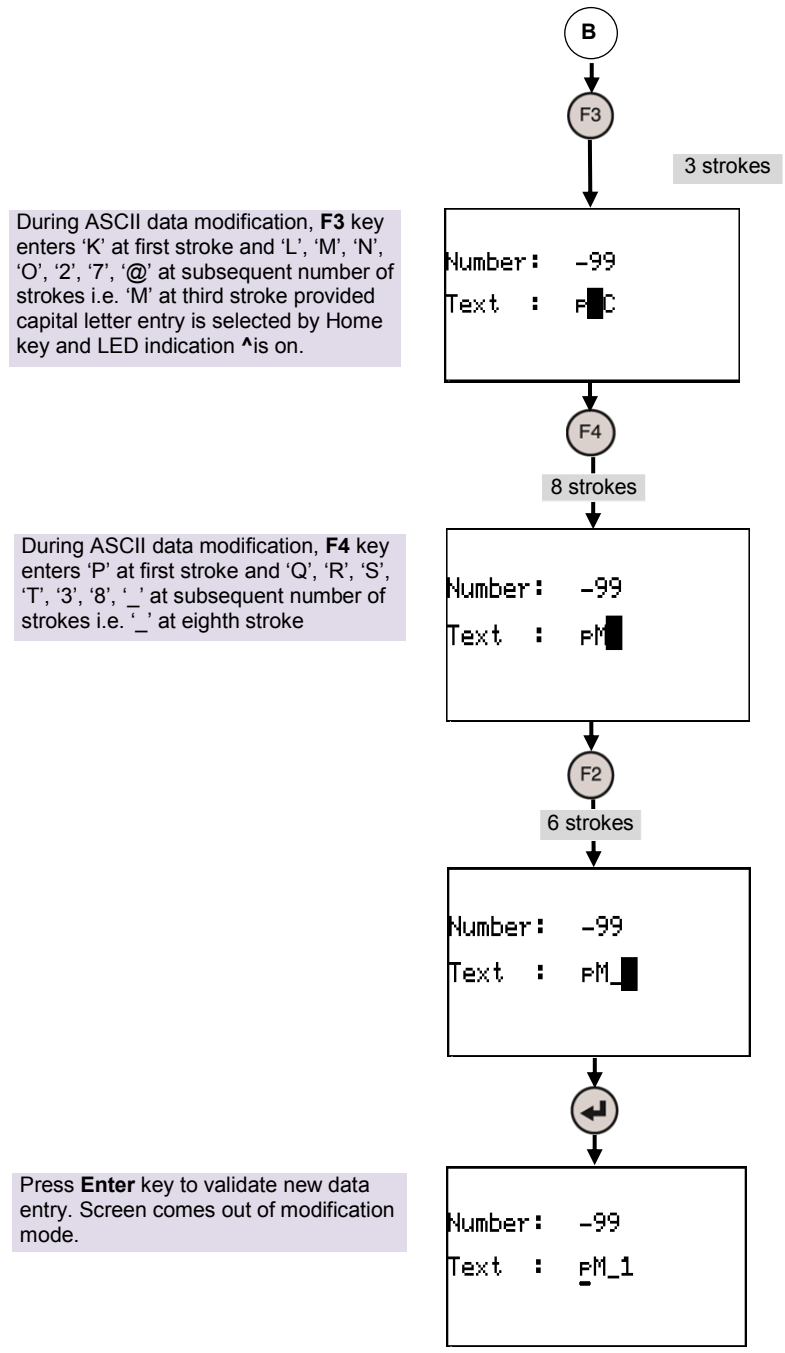
- Use PREV key to move cursor left
- Use NEXT key to move cursor right
- Use HOME key to toggle CAPS LOCK. Green LED near home key indicates CAPS LOCK is ON.

- To delete character press and hold PREV key for 3 sec.
- To insert space press and hold NEXT key for 3 sec.

Example:

Flowchart below explains how to modify ASCII text data from 'ABC' to 'pM_1' for an example. Screen shown below is programmed using *HMI Configuration Tool*. As per user programming, navigate through the screens to display screen shown below.




B
F3
3 strokes
During ASCII data modification, F3 key enters 'K' at first stroke and 'L', 'M', 'N', 'O', '2', '7', '@' at subsequent number of strokes i.e. 'M' at third stroke provided capital letter entry is selected by Home key and LED indication ^is on.
Number: -99
Text :
F4
8 strokes
During ASCII data modification, F4 key enters 'P' at first stroke and 'Q', 'R', 'S', 'T', '3', '8', '_' at subsequent number of strokes i.e. '_' at eighth stroke
Number: -99
Text :
F2
6 strokes
Number: -99
Text :
Press Enter key to validate new data entry. Screen comes out of modification mode.
Number: -99
Text : PM_1

26.5 Screen Objects

User can monitor/ modify timer delays, counters, temperatures, speed of the machine, production count, machine running hours and various machine/ process parameters. User can select mode of operations like auto/ manual selection, start/ stop any sequence, enable/ disable machine accessories. User can schedule and control activities day wise, week wise dependent on built-in Real Time Clock. User can log alarm data with time stamp.

All these features are user specific and user can develop an application for the same.

HMI offers various display objects those can be shown on a screen. These objects are mainly specific to data types. Using *HMI Configuration Tool*, user can select properties of individual objects and present PLC data in suitable format.

The display objects available are listed in the table below

	Display object	Description
1	Static Text	Displays fixed alphanumeric text. Static text is not associated with PLC data. Static text can be shown in various fonts.
2	Multi-lingual Text	Displays fixed multilingual text. Multilingual text can be shown in various windows fonts.
3	ASCII	Displays fixed alphanumeric text. Static text is not associated with PLC data. Static text can be shown in various fonts.
4	Multi-text	Displays a text from predefined list of text strings (text table) based on a value of controller variable associated with the object.
5	Bit	Displays bit data in form of TRUE text and FALSE text.
6	Numeric	Displays PLC data with data types like byte, word, integer, double word, double integer. User can select properties like no. of digits to be displayed, position of decimal point
7	Time	Displays PLC data with TIME data type. User can choose resolution like 1 ms, 0.1 Sec, 1 Sec, 0.1 minute and 1 minute
8	Time Of Day	Displays PLC data with TIME OF DAY (TOD) data type. User can choose format like HH:MM:SS and MM:SS
9	Date	Displays PLC data with DATE (DT) data type. User can choose format like DD:MM:YYYY and MM:DD:YYYY
10	Graphical Symbol	Displays graphical symbols. Various symbols are available with 32 x 32 pixel size and 48 x 48 pixel size.
11	Multi-symbol	Object links symbols from symbol table to the controller variable value. It shows symbol corresponding to a value of associated controller variable.
12	Image	Displays imported image

Points to Note

- User can assign various properties and associate memory address (%I, %Q, %M) or PLC variable with symbolic name.
- For bit type of objects, user can assign direct address like %MX10.2, PLC BOOL data with symbolic name or bit within PLC data with symbolic name with BYTE/WORD/DWORD data type.
- User can allow modification of value of a object using keypad. User can also set minimum limit and maximum limit for value.
- User can decide sequence of navigation of modifiable objects in screen. With the help of **DOWN** key, user can scroll through the modifiable objects with pre-defined sequence. Selected object shows line cursor at first digit/character.
- Virtually there is no limit for number of objects to be displayed on a screen. But size of LCD display and font limits no. of objects in a screen.
- HMI supports up to 64 screens. HMI application which includes screen definition, object properties and key configuration consumes application program memory. So application memory available for HMI application limits number of screens and no. objects per screen.
- Apart from configuring properties of display objects, user can configure properties of screen, functions keys, illuminated keys and dual-colour LEDs.
- *HMI Configuration Tool* creates PROGRAM type POU HMI_PRG in CoDeSys application program. POU HMI_PRG contains program code in Structured Text as per user configuration and is Read only type. This POU should not be changed. It will cause malfunctioning of HMI application or loss of HMI configuration.
- Screen updation time is 100 ms minimum.

26.6 Screen Actions

In some of the applications, user requires to carry out specific action like logic execution, calculations with reference to a particular screen displayed on LCD display. Action may be executed once before screen is displayed or once after screen is exited or continuously as long as screen is displayed on LCD screen. These actions are referred as screen actions.

There are three types of screen actions for a user screen.

1. Screen Entry Action

   Screen Entry action is executed only once when screen is displayed.

2. Screen Run Action

   Screen Run action is executed continuously as long as a particular screen is displayed. As this action is associated with screen updation time, action is executed at the duration of 100 ms.

3. Screen Exit Action

   Screen Exit action is executed only once when screen exits i.e. disappears from LCD display.

Switching of screens depend upon value of system variable `_SysvarHMI.STW`. Additionally, screen can be changed by using **Next** key if screen chaining is defined as discussed in subsequent section.

HMI Configuration Tool helps in defining screen actions for screens. User can link program type of POU with screen action and this POU is executed as defined action. These

26.7 Screen Chaining

Screen chaining feature is similar to GoTo key action linked to function keys and illuminated keys. With this provision, user can trigger defined screen on LCD display when the Next key is pressed. With combination of Next key and Back key, user can decide user screen navigation sequence as per application requirement.

26.8 Function Keys

HMI keypad provides 10 keys arranged in 2 rows for screen navigation and data entry. Refer **Figure 35**. Keys (**F1**, **F2**, **F3**, **F4** and **F5**) in the first row are named as function keys. These keys allow entering text characters and can be configured as function keys when display is not in modification mode. The functionality can be configured with the help of integrated tool *HMI Configuration Tool*.

User cannot configure any application specific functionality and use keys (**Home**, **Back**, **Down**, **Next** and **Enter**) in second row. Behavior of these keys is fixed and these keys are used mainly for display navigation purpose.

26.8.1 Key Action

User can configure various functionalities for 5 function keys; **F1** to **F5** as per application requirement e.g. auto/ manual mode selection, start/stop any sequence, enable/disable machine setting, screen switching, etc. *HMI Configuration Tool* provides following configurable actions for a function key. User can configure one action to a key at a time.

Key Action	Description
None	No action is configured.
Momentary action on bit	Assigned Boolean variable is set to TRUE as long as the key is pressed. When key is released, Boolean variable is reset to FALSE.
Toggle action on bit	Assigned Boolean variable is SET or RESET alternately when the key is pressed.
Go To screen	This action triggers specific screen on LCD when the key is pressed.

User configured functionality of function keys is not available when operator enters into modification mode by pressing **Enter** key.

In modification mode, these keys act as alphanumeric keys for data entry for numeric and text objects.

26.8.2 Local and Global Behavior

Further, scope of these keys can be configured as global and local.

Key Scope	Description
Local	Function key can be configured for different functionality for different screens i.e. 1 action per screen. So functionality of key changes as screen changes. Local key configuration allows multiple usage of a key that is screen specific.
Global	Global functionality is active in the complete application irrespective of screen displayed on LCD if local functionality is not configured for that key. If user has configured global and local functionality to a key, then global functionality is applicable for screen/s where local functionality is not configured. Local configuration is active for a key for which both local and global functionality is configured.

26.8.3 Key Legend

User can define key legend for 3 keys (F2, F3 and F4) which are located exactly below LCD. Key legends are displayed at bottom row of a screen. *HMI Configuration Tool* allows user to define 6 character size legend to identify key functionality. User can define different key legends for keys with local definition i.e. different key legend depending on screen number. Also, if same key has functionality of local and global, then user can define different key legend for global operation and local operation/s. As key

Key legend plays an important role as one key can be assigned with multiple functionality. It helps operator to identify key functionality depending upon screen displayed.

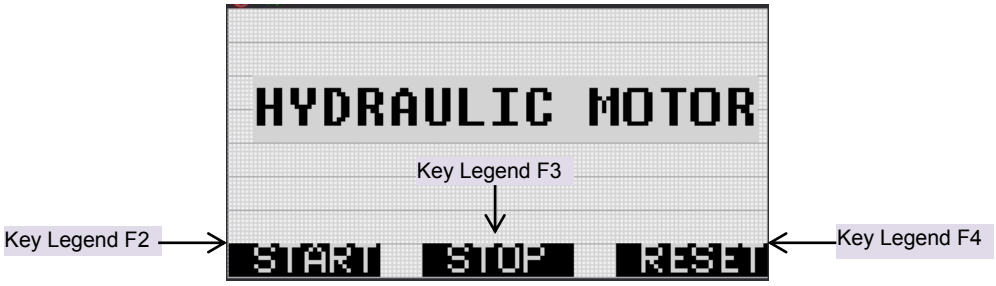


Figure 41: Key legend

26.9 Illuminated Key Panel

Controller comes with additional feature of Illuminated key panel along with slide-in label.

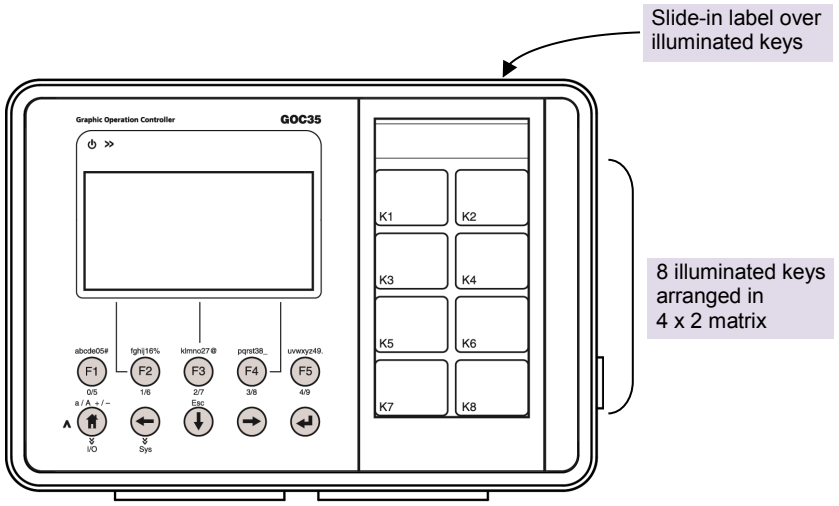


Figure 42: Illuminated keys with slide-in label

26.9.1 Key Action

User can configure various functionalities for 8 illuminated keys; **K1** to **K8** as per application requirement e.g. auto/ manual mode selection, start/stop any sequence, enable/disable machine setting, screen switching, alarm acknowledge/reset, etc. *HMI Configuration Tool* provides following configurable actions for a illuminated key. User can configure one action to a key at a time.

Key Action	Description
None	No action is configured.
Momentary action on bit	Assigned Boolean variable is set to TRUE as long as the key is pressed. When key is released, Boolean variable is reset to FALSE.
Toggle action on bit	Assigned Boolean variable is SET or RESET alternately when the key is pressed.
Go To screen	This action triggers specific screen on LCD when the key is pressed.

By default, scope of action of illuminated key is global. So illuminated key with desired functionality is active irrespective of screen displayed on LCD screen.

26.9.2 Dual LED Control

Dual colour (red, green) LED is provided behind each key (K1 to K8). Dual colour LEDs offer tri-colour effect with colors as red, green and yellow (red + green).LEDs can be switched on/off through application. When LED glow, respective key is illuminated.

User can define LED control byte and control switching on and off for individual dual LED through application program. The table below explains significance of value of LED control byte.

LED Control Byte Value	LED Status	Color
0 or >9	OFF	No color
1	ON	Red
2		Green
3		Yellow
4	Blinking Slow (500 ms ON, 500 ms OFF)	Red
5		Green
6		Yellow
7	Blinking Fast (250 ms ON, 250 ms OFF)	Red
8		Green
9		Yellow

26.9.3 Slide-in Label

Unit is dispatched with default slide-in label with key nomenclature as K1 to K8. Label can be designed and printed by customers especially OEMs as per the guidelines provided by MEI. Label combined with illuminated keys helps in customizing controller to suit application. It can be used as operator station with push buttons and lamps, alarm annunciation and machine/process mimic.

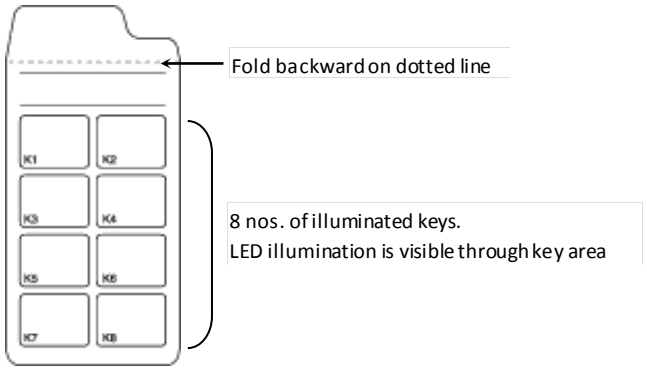


Figure 43: Slide in label

Slide-in label can be inserted from backside of controller front panel. Remaining part of side-in label should be folded to ensure plain surface over illuminated keys especially over K1 and K2 keys. Insertion slit is located at left top on the backside of front panel. Insert label from side opposite to the fold such that fold line is aligned with slit edge. Shape of folded part is designed such that it doesn't obstruct mounding clamp and I/O terminal block.

Refer section *Main Unit Installation* to know how to insert and remove slide-in label.

Guidelines for customizing slide-in label

1. Use ***GOC iKey slide-in label template.cdr*** file for dimensional details and sample label design.
2. Use material PVC with thickness 150 microns with glossy/mat finish.
   a. If failing to do so, may reduce pressing effect for individual key as well as disturbs smooth insertion of slide in label via insertion slot available at top side GOC towards back.
3. Process should be screen printing rather than digital printing which provides better quality and repeatability.
4. Printing should be carried out on the back side of slide-in label.
   a. Ensure that part excluding illuminated key parts (8 nos.) is completely opaque.
   b. User can get letters or any objects in white colour on opaque part of the slide-in label
5. LED light can be transmitted through rectangular illuminated key part. For the same, ensure that required portion of label (depending upon mimic drawing, alarm annunciation requirement, etc) should be translucent white.
   a. Translucent effect is mandatory to ensure uniform illumination.
   b. Illuminated key part should not be transparent completely as it will show illuminated ring appearing on top of the key due to key element projection provided on backside of key
   c. Align mimic objects considering illuminated key parts.
6. Ensure that slide-in label is folded towards backside at dotted line. If not folded. Failing to do so may create bulge below Mylar on GOC front panel.
7. Ensure that profile of slide-in label is maintained to ensure easy insertion.

Appendix 1: GOC File Explorer

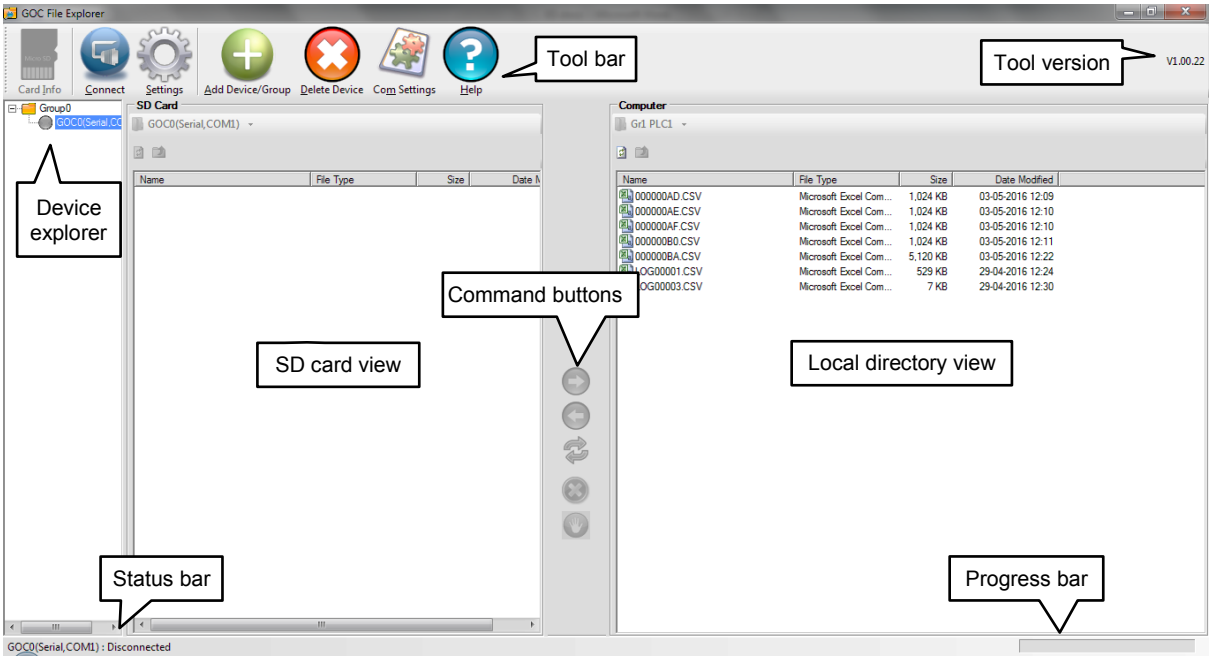
Overview

PC based 'GOC File Explorer' tool can be used to upload files on SD card inserted in GOC Main unit. Installation of GOC File Explorer is available in GOC Toolkit CD. However user needs to install it separately. It will not be installed automatically along with GOC Toolkit.

After installation, user can start the tool from Start menu path as

'Start > All Programs > GOC File Explorer > GOC File Explorer'

Following is the GUI of the tool



As shown in above screen shot, following are sections of GOC File Explorer

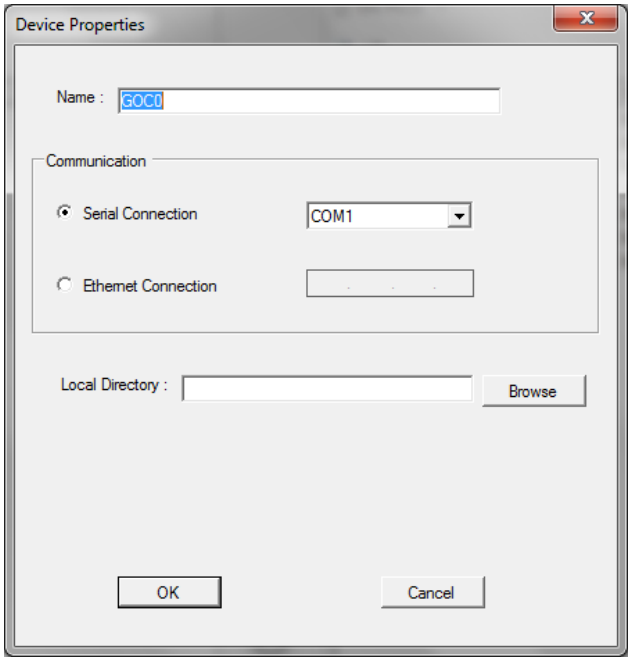
Section	Details
Toolbar	The Tool Bar offers quick access buttons for frequently used commands. Following command buttons are available **Card Info:** Reads SD card information. Button is active only if selected device is connected. If SD card is not present in the GOC Main unit then card size is displayed as 0. **Connect:** Establishes communication with selected device and lists files in SD card view. **Setting:** Displays device properties (Name, Connection type, Local directory path, etc.) of selected device. **Add Device/Group:** Adds device to selected group. If group is not selected then adds new group. **Delete Device:** Deletes selected Device / Group. **Com Settings:** Opens dialogue to set communication settings like timeout, retries. **Help:** Opens Help file.

Section	Details
Tool Version	Displays version of GOC File Explorer
Device Explorer	Device explorer displays all the devices (GOC) configured in tool. Currently tool allows configuration of only one device and one group.
SD Card View	When connected, shows files / folders from SD card of connected device. Displays nothing when SD card is not present in connected device.
Local Directory	Local directory explorer view displays files and folders from local directory assigned for selected device in device explorer
Command buttons	Command buttons are used for various file operations like move files between SD card and local folder, delete file, etc.
Status Bar	Status bar displays connection status and other status messages.
Progress Bar	Displays progress of file / folder reading operation.

Adding New Device (GOC) to Device Explorer

To add new device to device explorer, select a group in which device is to be added and click on 'Add Device' button from tool bar. Additionally, device can be added by right clicking on Group and selecting option 'Add Device'.

Device Properties dialogue appears as



Setting	Description
Name	Identification name for device (GOC). User can assign any identification name of length 32 characters.
Communication	Select connection type **Serial:** Selected COM port from PC that is used for RS232 connection with device. Device will communicate with tool via GC-RS232-COM extension unit inserted in COM1 port of Main unit using standard programming cable.

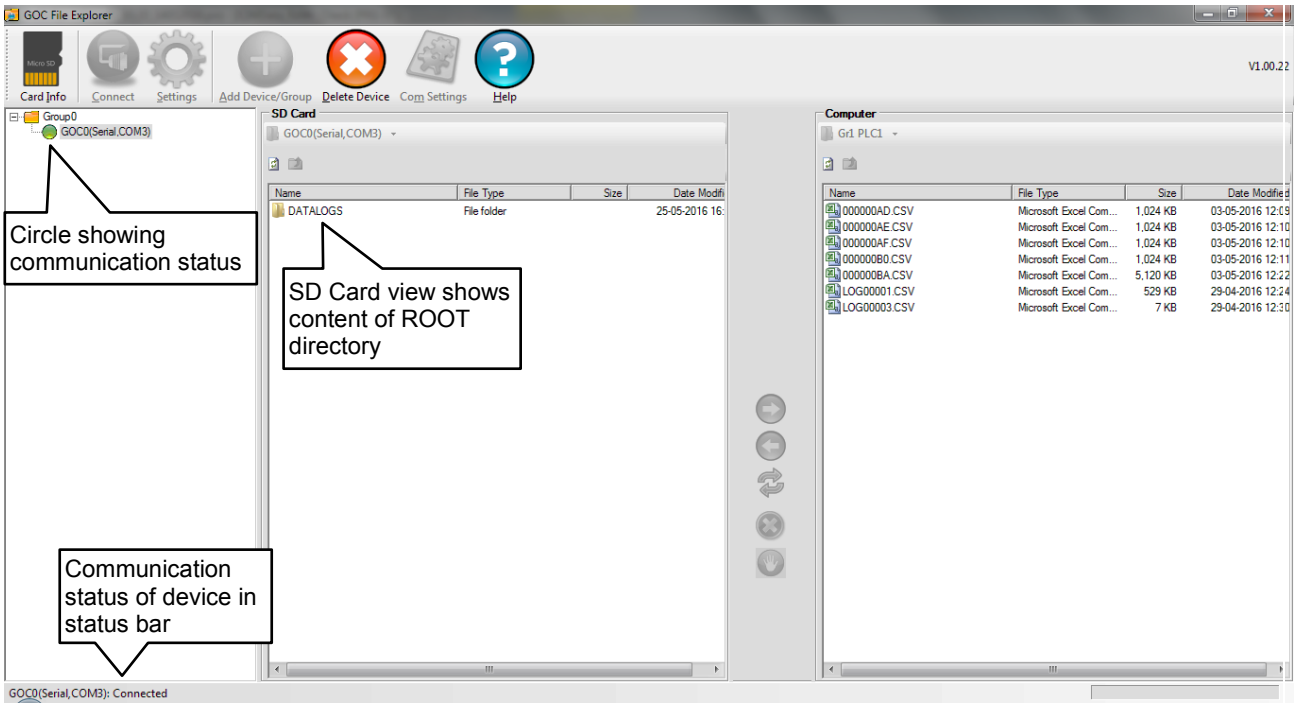
Setting	Description
	Ethernet: Tool will establish Ethernet communication with device. Device will communicate via GC-ENET-COM extension unit inserted in COM1 slot on Main unit. Mention target IP address.
Local Directory	Tool allows assigning particular directory from computer's local drive to a device. This directory is used to store uploaded data log files from the device. This local directory path is remembered by tool.

Connecting to Device

Select device from device explorer and click on Connect button from toolbar or right click on selected device and select option 'Connect'.

Tool establishes communication with selected device with connection type set in device properties.

After successful connection with device, tool reads content of 'ROOT' directory of SD card and display content in SD card view as shown below



Note the circle in front of device in device explorer is now GREEN showing device is connected. Following table shows possible colors of the circle.

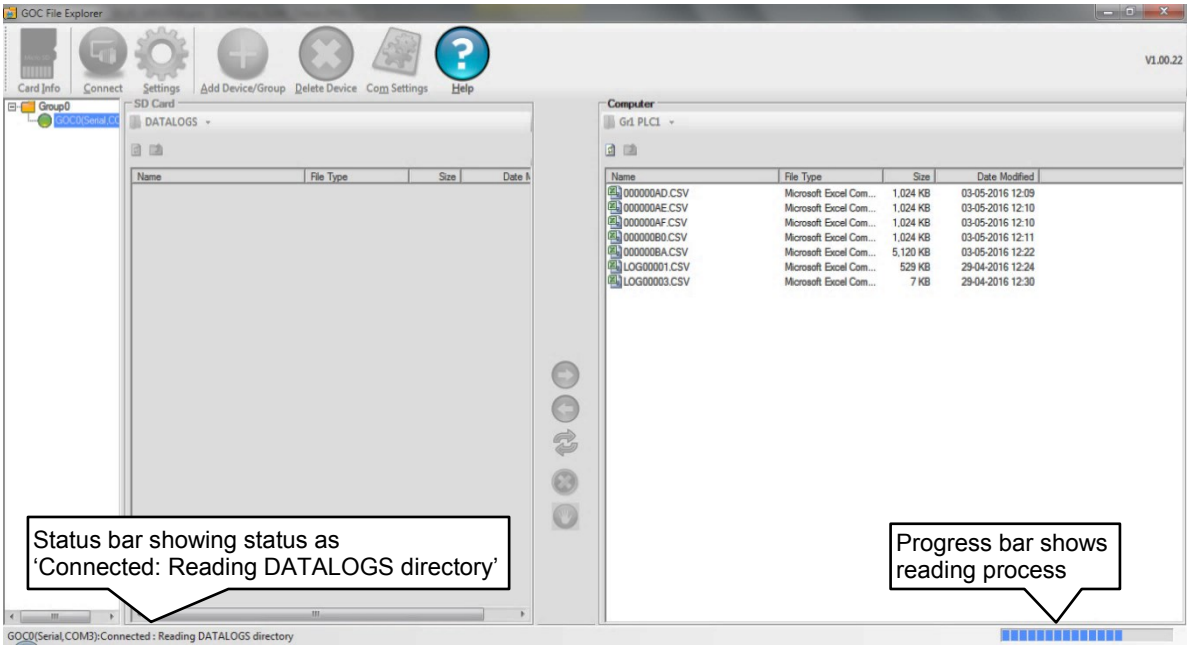
Colour	Significance
	Not connected
	Communication error
	Trying to connect
	Connected

Also note status bar showing communication status as connected.

View Files from Folder

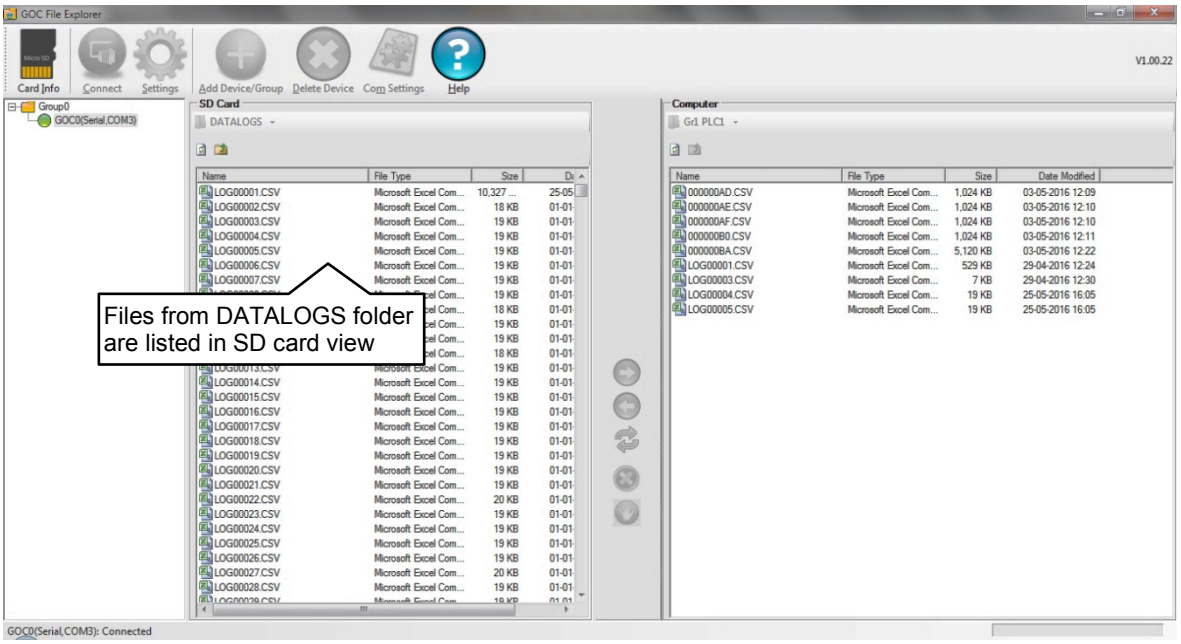
To view files from DATALOGS folder, double click on ‘DATALOGS’ folder in SD card view.

Tool starts reading file list from ‘DATALOGS’ folder from SD card.



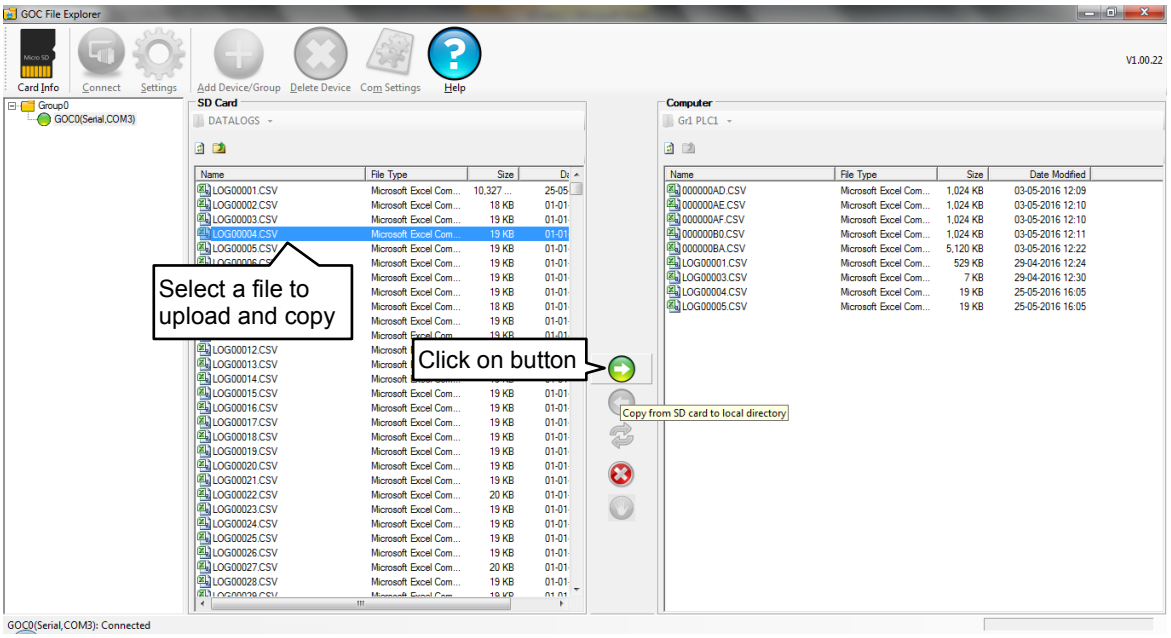
Status bar shows status as ‘Device Name: Connected: Reading DATALOGS directory’ and progress bar shows % of reading process.

After reading is finished, files in folder are displayed in SD card view as shown below

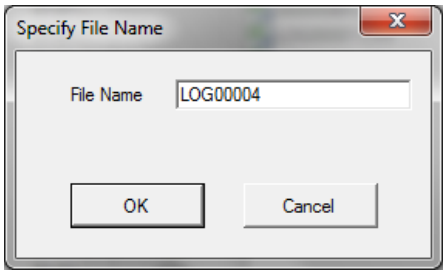


Uploading File

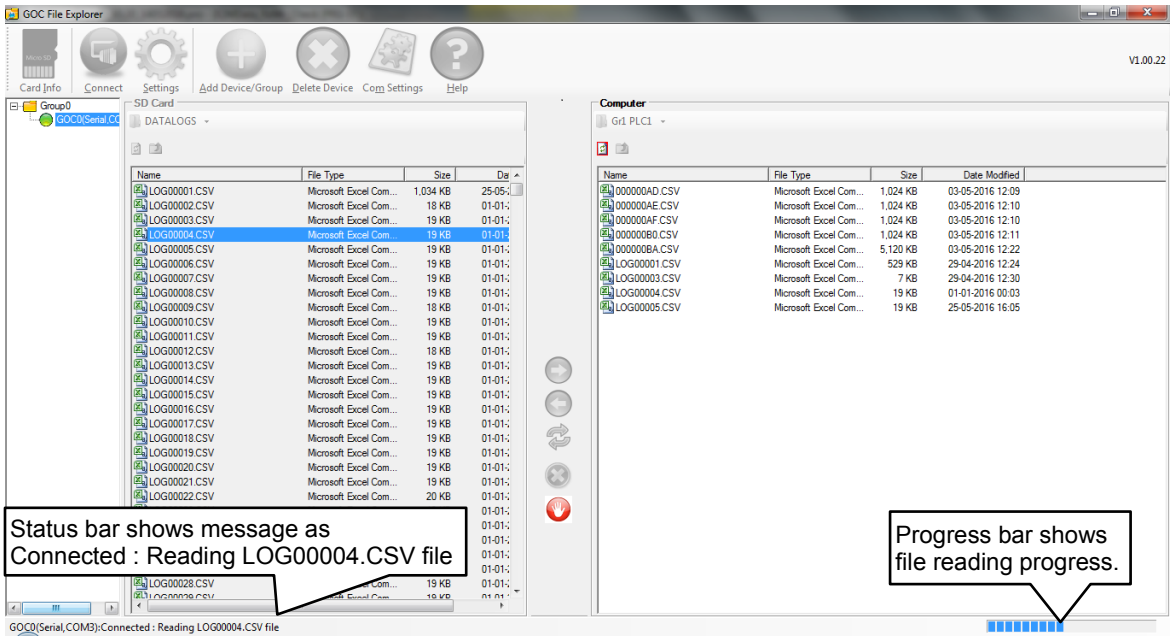
Select a file from SD card view and click right arrow button to upload file from SD card and copy to local directory.



Dialogue pops-up to specify file name as



If required, change the file name and click OK button to start uploading file from connected device and transferring to the local folder.



Following command buttons are provided

Button	Details
	Copy file from SD card to local directory.
	Copy file from local directory to SD card. This button is currently inactive.
	Delete selected file. Button used to delete file from local directory. Currently deleting file from SD card is not supported.
	Synchronise 'Datalogs' folder with local directory. This button is currently inactive.
	Abort file reading operation.

Time required for file reading

Accessing SD card is low priority background process, hence operations from file explorer tool are relatively slow. Hence, reading large log data files takes considerable time.

Following table gives typical time required to read files for various file sizes

File size	Time required
10KB	20 sec
100KB	1 min 25 sec
1MB	9 min
10MB	1 hr 50 min

For copying large size log files it is advised to use card reader.

Mitsubishi Electric India Pvt. Ltd.

Factory Automation and Industrial Division

EL 3, J Block, MIDC Bhosari, PUNE – 411 026 (INDIA)

Email – mei-faid-enquiry@asia.meap.com

Web – http://in.mitsubishielectric.com