Serial Communications

Chapter 5

Table of Contents

Chapter 5: Serial Communications

Communications Parameters Summary 5–2

Summary – Serial Communication Parameters 5–2

Serial Modbus Status Addresses 5–6

Status Addresses (Read Only). 5–6

Serial Communications Overview 5–9

Serial Communications Connectivity. 5–9

Minimum AC Drive Parameter Settings For Serial Communication 5–9

Common Third-Party Modbus RTU Masters. 5–10

AutomationDirect PLCs as Modbus Master 5–10

Connecting Communication Cables 5–11

Detailed Serial Modbus Communication Information 5–13

Data Format 5–13

Communication Protocol 5–14

CMD (command code) and DATA (data characters) 5–15

Common Third-Party Modbus RTU Masters. 5–10

AutomationDirect PLCs as Modbus Master 5–10

Connecting Communication Cables 5–11

Detailed Serial Modbus Communication Information 5–13

Data Format 5–13

Communication Protocol 5–14

CMD (command code) and DATA (data characters) 5–15

Communications Parameters Summary

A summary of the GS30 AC drives Communications Parameters is listed below. Refer to "Parameters" Chapter 4 for a complete listing of all GS30 AC drives parameters, including details and Modbus addresses.

For GS30A-CM-ENETIP communication card parameters and information, please see Appendix B.

Summary – Serial Communication Parameters

GS30 Parameters Summary – Communication Parameters (P09.xx)							
Parameter		**Range**	**Run[1] Read/ Write**	**Modbus Address**		**Settings**	
				Hex	**Dec**	**Default[2]**	**User**
1) ♦ in the Run-Read/Write column indicates that the parameter can be set during RUN mode. R/W indicates "Read/write." Read indicates "Read-only." 2) Parameters can be restored to their default values using P00.02.							
P09.00	Communication address	1–254	♦R/W	0900	42305	1	
P09.01	COM1 transmission speed	4.8–115.2 Kbps	♦R/W	0901	42306	9.6	
P09.02	COM1 transmission fault treatment	0: Warn and continue operation 1: Fault and ramp to stop 2: Fault and coast to stop 3: No warning, no fault, and continue operation	♦R/W	0902	42307	3	
P09.03	COM1 time-out detection	0.0–100.0 sec.	♦R/W	0903	42308	0.0	
P09.04	COM1 communication protocol	1: 7, N, 2 (ASCII) 2: 7, E, 1 (ASCII) 3: 7, O, 1 (ASCII) 4: 7, E, 2 (ASCII) 5: 7, O, 2 (ASCII) 6: 8, N, 1 (ASCII) 7: 8, N, 2 (ASCII) 8: 8, E, 1 (ASCII) 9: 8, O, 1 (ASCII) 10: 8, E, 2 (ASCII) 11: 8, O, 2 (ASCII) 12: 8, N, 1 (RTU) 13: 8, N, 2 (RTU) 14: 8, E, 1 (RTU) 15: 8, O, 1 (RTU) 16: 8, E, 2 (RTU) 17: 8, O, 2 (RTU)	♦R/W	0904	42309	15	
P09.09	Communication response delay time	0.0–200.0 ms	♦R/W	0909	42314	2.0	
P09.10	Communication main frequency	0.00–599.00 Hz	R/W	090A	42315	60.00	
P09.11	Block transfer 1	0–65535	♦R/W	090B	42316	0	
P09.12	Block transfer 2	0–65535	♦R/W	090C	42317	0	
P09.13	Block transfer 3	0–65535	♦R/W	090D	42318	0	
P09.14	Block transfer 4	0–65535	♦R/W	090E	42319	0	
P09.15	Block transfer 5	0–65535	♦R/W	090F	42320	0	
P09.16	Block transfer 6	0–65535	♦R/W	0910	42321	0	
P09.17	Block transfer 7	0–65535	♦R/W	0911	42322	0	
P09.18	Block transfer 8	0–65535	♦R/W	0912	42323	0	
P09.19	Block transfer 9	0–65535	♦R/W	0913	42324	0	
P09.20	Block transfer 10	0–65535	♦R/W	0914	42325	0	
P09.21	Block transfer 11	0–65535	♦R/W	0915	42326	0	

GS30 Parameters Summary - Serial Communication Parameters (P09.xx) - (continued)							
Parameter		**Range**	**Run[1] Read/ Write**	**Modbus Address**		**Settings**	
				Hex	**Dec**	**Default[2]**	**User**
P09.22	Block transfer 12	0–65535	♦R/W	0916	42327	0	
P09.23	Block transfer 13	0–65535	♦R/W	0917	42328	0	
P09.24	Block transfer 14	0–65535	♦R/W	0918	42329	0	
P09.25	Block transfer 15	0–65535	♦R/W	0919	42330	0	
P09.26	Block transfer 16	0–65535	♦R/W	091A	42331	0	
P09.30	Communication decoding method	0: Decoding method 1 1: Decoding method 2	R/W	091E	42335	1	
P09.33	PLC command force to 0	0–65535	♦R/W	0921	42338	0	
P09.35	PLC address	1–254	R/W	0923	42340	2	
P09.60	Communication card identification	0: No communication card 4: Modbus-TCP slave 5: EtherNet/IP slave 10: Backup power supply	Read	093C	42365	0	
P09.61	Firmware version of communication card	Read only	Read	093D	42366	0	
P09.62	Product code	Read only	Read	093E	42367	0	
P09.63	Error code	Read only	Read	093F	42368	0	
P09.74	Set Comm Master Protocol	0: Ethernet IP and Modbus TCP both 1: Ethernet IP 2: Modbus TCP	♦R/W	094A	42379	1	
P09.75	Communication card IP configuration (Ethernet)	0: Static IP 1: Dynamic IP (DHCP)	♦R/W	094B	42380	0	
P09.76	Communication card IP address 1 (Ethernet)	0–255	♦R/W	094C	42381	0	
P09.77	Communication card IP address 2 (Ethernet)	0–255	♦R/W	094D	42382	0	
P09.78	Communication card IP address 3 (Ethernet)	0–255	♦R/W	094E	42383	0	
P09.79	Communication card IP address 4 (Ethernet)	0–255	♦R/W	094F	42384	0	
P09.80	Communication card address mask 1 (Ethernet)	0–255	♦R/W	0950	42385	0	
P09.81	Communication card address mask 2 (Ethernet)	0–255	♦R/W	0951	42386	0	
P09.82	Communication card address mask 3 (Ethernet)	0–255	♦R/W	0952	42387	0	
P09.83	Communication card address mask 4 (Ethernet)	0–255	♦R/W	0953	42388	0	
P09.84	Communication card gateway address 1 (Ethernet)	0–255	♦R/W	0954	42389	0	
P09.85	Communication card gateway address 2 (Ethernet)	0–255	♦R/W	0955	42390	0	
P09.86	Communication card gateway address 3 (Ethernet)	0–255	♦R/W	0956	42391	0	
P09.87	Communication card gateway address 4 (Ethernet)	0–255	♦R/W	0957	42392	0	

GS30 Parameters Summary - Serial Communication Parameters (P09.xx) - (continued)							
Parameter		Range	Run1) Read/ Write	Modbus Address		Settings	
				Hex	Dec	Default2)	User
P09.88	Communication card password (low word) (Ethernet)	0–99	♦R/W	0958	42393	0	
P09.89	Communication card password (high word) (Ethernet)	0–99	♦R/W	0959	42394	0	
P09.90	Reset communication card (Ethernet)	0: Disable 1: Reset to defaults	♦R/W	095A	42395	0	
P09.91	Additional settings for the communication card (Ethernet)	bit 0: Enable IP filter bit 1: Enable internet parameters (1 bit) When the IP address is set, this bit is enabled. After updating the parameters for the communication card, this bit changes to disabled. bit 2: Enable login password (1 bit) When you enter the login password, this bit is enabled. After updating the communication card parameters, this bit changes to disabled.	♦R/W	095B	42396	0	
P09.92	Communication card status (Ethernet)	bit 0: Enable password When the communication card is set with a password, this bit is enabled. When the password is cleared, this bit is disabled.	R/W	095C	42397	0	
P09.93	ENETIP Comm Card Fault Select	0: Warn & Continue Operation 1: Warn & Ramp to Stop 2: Warn & Coast to Stop 3: No Warning & Continue Operation	♦R/W	095D	42398	3	
P09.94	ENETIP Comm Card Time Out Detection	0: Disable 1: Enable	♦R/W	095E	42399	1	
P09.95	ENETIP Comm Card Time Out Duration	0.1 to 100.0 seconds	♦R/W	095F	42400	3.0	

Block Transfer Explanation

Block Transfer allows Parameters from many different Parameter Groups to be consolidated into one (or fewer) Modbus communication messages. This can greatly simplify PLC programming and reduce network traffic.

The Block Transfer parameters are P09.11 through P09.26. To use these parameters, enter the value of another parameter you wish to read or write through the keypad or GSoft2 configuration software. The parameter values must be converted by adding the upper byte value to the lower byte value, convert the sum to hex, then convert the hex to decimal.

Example:

Parameter P02.22. 0200 + 16 (hex of 22) = 0x0216 = result is 534. 534 is what would be entered in the Block Transfer parameter to read or write parameter P02.22.

Examples of Block Transfer are below:

1) Block transfer 1 (P09.11) = 0000 (AC Motor drive identity code). A Modbus read of P09.11 results in a value of 104. In this case, the drive is model # GS21-11P0 and corresponds to the value 104 in Parameter P00.00.
2) Block transfer 2 (P09.12) = 0006 (Firmware version). A Modbus read of P09.12 results in a value of 100. This is the firmware version of the GS30 drive.
3) Block transfer 3 (P09.13) = 8448 (decimal value of 0x2100 Status Monitor 1). A Modbus read of P09.13 returns the current status of Status Monitor 1.

4) Block transfer 4 (P09.14) = 8449 (decimal value of 0x2101 Status Monitor 2). A Modbus read of P09.14 returns the current status of Status Monitor 2.
5) Block transfer 5 (P09.15) = 8451 (decimal value of 0x2103 Output Frequency). A Modbus read of P09.15 returns the current running frequency of the GS30.
6) Block transfer 6 (P09.16) = 0268 (Acceleration time 1 is parameter P01.12. 12 = 0x0c. 0100 + 0c = 0x010C = 0268 decimal). A Modbus write to P09.16 will set the Acceleration time 1 value.
7) Block transfer 7 (P09.17) = 0269 (Deceleration time 1 is parameter P01.13. 13 = 0x0d. 0100 + 0d = 0x010d = 0269 decimal). A Modbus write to P09.17 will set the Deceleration time 1 value.
8) Block transfer 8 (P09.18) = 8192 (Control Word 1 (Run, Stop, etc…) is 0x2000 = 8192). A Modbus write to P09.18 will control the Run/Stop of the drive along with other items.
9) Block transfer 9 (P09.19) = 8193 (Control Word 2 (Frequency Command) is 0x2001 = 8193). A Modbus write to P09.19 will control the commanded Frequency of the drive.

Accessing all of the registers above would typically take about 6 Modbus messages but by blocking them together in the Block Transfer parameters, we can access everything with 1 read and 1 write.

Serial Modbus Status Addresses

The GS30 AC drive has status memory addresses that are used to monitor the AC drive.

Status Addresses (Read Only)

GS30 Addresses						
Description		***Range***		***Modbus Address***		
				Hex	***Dec***	***Octal***
Status Monitor 1 Read Only	Fault Codes	0: No fault record 1: Over-current during acceleration (ocA) 2: Over-current during deceleration (ocd) 3: Over-current during steady operation (ocn) 4: Ground fault (GFF) 6: Over-current at stop (ocS) 7: Over-voltage during acceleration (ovA) 8: Over-voltage during deceleration (ovd) 9: Over-voltage during constant speed (ovn) 10: Over-voltage at stop (ovS) 11: Low-voltage during acceleration (LvA) 12: Low-voltage during deceleration (Lvd) 13: Low-voltage during constant speed (Lvn) 14: Low-voltage at stop (LvS) 15: Phase loss protection (orP) 16: IGBT overheating (oH1) 18: IGBT temperature detection failure (tH1o) 21: Over load (oL) 22: Electronic thermal relay 1 protection (EoL1) 23: Electronic thermal relay 2 protection (EoL2) 24: Motor PTC overheating (oH3) 26: Over torque 1 (ot1) 27: Over torque 2 (ot2) 28: Under current (uC) 31: EEPROM read error (cF2) 33: U-phase error (cd1) 34: V-phase error (cd2) 35: W-phase error (cd3) 36: cc (current clamp) hardware error (Hd0) 37: oc (over-current) hardware error (Hd1) 40: Auto-tuning error (AUE) 41: PID loss AI2 (AFE) 43: PG feedback loss (PGF2) 44: PG feedback stall (PGF3) 45: PG slip error (PGF4) 48: AI2 loss (ACE) 49: External fault (EF) 50: Emergency stop (EF1) 51: External base block (bb) 52: Password is locked (Pcod) 54: Illegal command (CE1) 55: Illegal data address (CE2)	56: Illegal data value (CE3) 57: Data is written to read-only address (CE4) 58: Modbus transmission time-out (CE10) 61: Y-connection / Δ-connection switch error (ydc) 62: Deceleration energy backup error (dEb) 63: Over slip error (oSL) 72: STO Loss (STL1) 76: STO (STo) 77: STO Loss 2 (STL2) 78: STO Loss 3 (STL3) 79: U-phase over-current before run (Aoc) 80: V-phase over-current before run (boc) 81: W-phase over-current before run (coc) 82: Output phase loss U phase (oPL1) 83: Output phase loss V phase (oPL2) 84: Output phase loss W phase (oPL3) 87: Low frequency overload protection (oL3) 89: Rotor position detection error (roPd) 97: Ethernet Card Timeout (CD10) 111: InrCOM time-out error (ictE) 121: Internal communication error (CP20) 123: Internal communication error (CP22) 124: Internal communication error (CP30) 126: Internal communication error (CP32) 127: Internal communication error (CP33) 128: Over-torque 3 (ot3) 129: Over-torque 4 (ot4) 134: Internal communication error (EoL3) 135: Internal communication error (EoL4) 140: Oc hardware error (Hd6) 141: GFF occurs before run (b4GFF) 142: Auto-tune error 1 (DC test stage) (AuE1) 143: Auto-tune error 2 (High frequency test stage) (AuE2) 144: Auto-tune error 3 (Rotary test stage) (AuE3) 149: Auto-tune error 5 (Rotor resistance measure test stage) (AuE5)	0611	41554	3021
Note: *Status Monitor 1 corresponds to P06.17 Fault Record 1.*						

GS30 Addresses (continued)					
Description	***Range***		***Modbus Address***		
			Hex	***Dec***	***Octal***
Status monitor read only	High byte: Warning code / Low Byte: Error code		2100	48449	20400
	bit 1–0	AC motor drive operation status 00B: The drive stops 01B: The drive is decelerating 10B: The drive is in standby status 11B: The drive is operating	2101	48450	20401
	bit 2	1: JOG command			
	bit 4–3	Operation direction 00B: FWD running 01B: From REV running to FWD running 10B: From FWD running to REV running 11B: REV running			
	bit 8	1: Master frequency controlled by the communication interface			
	bit 9	1: Master frequency controlled by the analog / external terminal signal			
	bit 10	1: Operation command controlled by the communication interface			
	bit 11	1: Parameter locked			
	bit 12	1: Enable to copy parameters from keypad			
	bit 15–13	Reserved			
	Frequency command (XXX.XX Hz)		2102	48451	20402
	Output frequency (XXX.XX Hz)		2103	48452	20403
	Display the drive’s output current (XX.XX A). When the current is higher than 655.35, it automatically shifts one decimal place as (XXX.X A). Refer to the high byte of 211F for information on the decimal places.		2104	48453	20404
	DC bus voltage (XXX.X V)		2105	48454	20405
	Output voltage (XXX.X V)		2106	48455	20406
	Current step for the multi-step speed operation		2107	48456	20407
	Reserved		2108	48457	20410
	Counter value		2109	48458	20411
	Output power factor angle (XXX.X)		210A	48459	20412
	Output torque (XXX.X %)		210B	48460	20413
	Actual motor speed (XXXXX rpm)		210C	48461	20414

GS30 Addresses (continued)					
Description	**Range**		**Modbus Address**		
			Hex	**Dec**	**Octal**
Command write only	bit 1–0	00B: No function	2000	48193	20000
		01B: Stop			
		10B: Run			
		11B: JOG + RUN			
	bit 3–2	Reserved			
	bit 5–4	00B: No function			
		01B: FWD			
		10B: REV			
		11B: Change direction			
	bit 7–6	00B: 1st accel. / decel.			
		01B: 2nd accel. / decel.			
		10B: 3rd accel. / decel.			
		11B: 4th accel. / decel.			
	bit 11–8	000B: Master speed			
		0001B: 1st step speed frequency			
		0010B: 2nd step speed frequency			
		0011B: 3rd step speed frequency			
		0100B: 4th step speed frequency			
		0101B: 5th step speed frequency			
		0110B: 6th step speed frequency			
		0111B: 7th step speed frequency			
		1000B: 8th step speed frequency			
		1001B: 9th step speed frequency			
		1010B: 10th step speed frequency			
		1011B: 11th step speed frequency			
		1100B: 12th step speed frequency			
		1101B: 13th step speed frequency			
		1110B: 14th step speed frequency			
		1111B: 15th step speed frequency			
	bit 12	1: Enable bit 06–11 function			
	bit 14–13	00B: No function			
		01B: Operated by the digital keypad			
		10B: Operated by Pr.00-21 setting			
		11B: Change the operation source			
	bit 15	Reserved			
	Frequency command (XXX.XX Hz)		2001	48194	20001
	bit 0	1: E.F. (External Fault) ON	2002	48195	20002
	bit 1	1: Reset command			
	bit 2	1: B.B. ON			
	bit 4–3	Reserved			
	bit 5	1: Enable fire mode			
	bit 15–6	Reserved			

Serial Communications Overview

The GS30 RJ-45 Serial Comm Port will accommodate an RS-485 connection, through which the drive can be controlled by a remote master device on an RS-485 network spanning up to 1200 meters (4000 feet) of cable. RS-232 signals can be converted to RS-485 by using a separate converter.

The GS30 AC drive communication address is specified in P9.00, and the remote master device can control each AC drive according to its individual communication address.

The GS30 AC drive can be configured to communicate using either Modbus RTU or ASCII. The desired protocol is selected in parameter P09.04, COM1 Protocol. (The GS30 drive cannot use both protocols simultaneously.)

- *Standard Modbus protocol using ASCII or RTU transmission modes. Parameter P09.04, Communication Protocol, is used to select the desired mode, number of data bits, parity, and number of stop bits. The mode and serial parameters must be the same for all devices on a Modbus network.*

DURApulse GS30 drives have a provision for shutting down control or power to the inverter in the event of a communications time out. This feature can be set up through parameters P09.02 (COM1 transmission fault treatment) and P09.03 (COM1 time-out detection).

Ethernet connectivity for EtherNet/IP or Modbus TCP communication is possible with an optional communication card # GS30A-CM-EIP1 or GS30A-CM-EIP2.
Refer to "Appendix B: Optional I/O and Communication Cards" for details.

Serial Communications Connectivity

This section contains information regarding wiring connections to the GS30 RS-485 serial communication ports. For information regarding serial connections to AutomationDirect PLCs, please refer to Appendix D of this user manual, or to the applicable PLC user manual.

Minimum AC Drive Parameter Settings For Serial Communication

The following parameters need to be set as shown in order to communicate properly:

Minimum Parameter Settings (for Communication to ADC PLC)		
Parameter Setting	***Description***	***Setting Value Explanation***
P00.21 = 02	1st Source of Operation Command [Remote]	02: RS-485 communication input
P00.31 = 02	2nd Source of Operation Command [Local]	02: RS-485 communication input, Keypad STOP is Enabled (P00.32)
P02.01~P02.07 = 56	Multifunction Inputs (DI1-DI7) Definition	56: Local/Remote selection
P00.20 = 1	1st Source of Frequency Command [Remote]	1: RS-485 communication input
P00.30 = 1	2nd Source of Frequency Command [Local]	1: RS-485 communication input
P09.00 = 1~254	Communication Address	01~254 Drive Comm Address
P09.01 = 4.8~115.2	Transmission Speed	4.8–115.2 Kbps
P09.04 = 1 to 17	COM1 Protocol	1: 7, N, 2 (ASCII) 2: 7, E, 1 (ASCII) 3: 7, O, 1 (ASCII) 4: 7, E, 2 (ASCII) 5: 7, O, 2 (ASCII) 6: 8, N, 1 (ASCII) 7: 8, N, 2 (ASCII) 8: 8, E, 1 (ASCII) 9: 8, O, 1 (ASCII) 10: 8, E, 2 (ASCII) 11: 8, O, 2 (ASCII) 12: 8, N, 1 (RTU) 13: 8, N, 2 (RTU) 14: 8, E, 1 (RTU) 15: 8, O, 1 (RTU) 16: 8, E, 2 (RTU) 17: 8, O, 2 (RTU)

This list of parameter settings is the minimum required to communicate with an AutomationDirect PLC. There may be other parameters that need to be set to meet the needs of your particular application.

Common Third-Party Modbus RTU Masters

- *KEPSERVER EX 5.0 from www.kepware.com*
- *Modbus Poll from www.modbustools.com*

AutomationDirect PLCs as Modbus Master

Serial Modbus-capable AutomationDirect PLCs can communicate with the GS30 drive (for GS30 Ethernet and Modbus TCP connectivity and control, refer to the GS30A-CM-EIP1/EIP2 Communication card information in Appendix B).

Serial Modbus control is easier to accomplish from a PLC that has a built-in RS-485 port and supports dedicated Modbus messaging. [RS-232-only PLCs will require an RS-232–RS-485 converter (FA-ISOCON); and older PLCs may require programming to construct the Modbus strings.] We recommend PLCs with built-in RS-485 ports and dedicated Modbus serial commands: CLICK (with RS-485 ports), Productivity 1000/2000/3000, BRX/Do-more, DirectLogic (DL06, D2-260, or D2-262). Other PLC-Drive connectivity is possible: Please refer to the "Typical ADC PLC to GS30 Serial Connectivity Matrix" below.

Typical ADC PLC to GS30 Serial Communications Connectivity

Typical ADC PLC to GS30 Serial Communications Connectivity Matrix*						
Recommended PLC Connectivity			***Communication***	***Direct Cable***	***GS30***	
PLC	***Port #***	***Port Type***			***Port Type***	***Port #***
CLICK	3	3 screw terminals	RS-485	Q8304-1 cable	RJ45 or SG+ SG- SGND	RJ45
D2-260	2	HD15	RS-485	D2-DSCBL-2		
D2-262	2	HD15	RS-485	D2-DSCBL-2		
DL06	2	HD15	RS-485	D2-DSCBL-2		
BRX/Do-more	RS-485	3 screw terminals	RS-485	Q8304-1 cable		
Do-more H2-DM1	RS-232	RJ12	RS-232 to RS-485	FA-ISOCON with Q8304-1 cable		
P1-550	RS-485	4 screw terminals	RS-485	Q8304-1 cable		
P2-550	RS-485	3 screw terminals	RS-485	Q8304-1 cable		
P3-530	RS-485	3 screw terminals	RS-485	Q8304-1 cable		
P3-550	RS-485	3 screw terminals	RS-485	Q8304-1 cable		
P3-550E	RS-485	3 screw terminals	RS-485	Q8304-1 cable		
Other PLC Connectivity			–	–		
D2-250-1	2	HD15	RS-485	D2-DSCBL-2		
D4-450/D4-454	1	DB25	RS-232 to RS-485	FA-ISOCON with Q8304-1 cable		
DL05	2	RJ12	RS-232 to RS-485	FA-ISOCON with Q8304-1 cable		
DL06 + DCM	2	HD15	RS-485	D2-DSCBL-2		
Do-more H2-DM1 + H2-SERIO-4	3	5 screw terminals	RS-485	Q8304-1 cable		
Do-more T1H-DM1	RS-232	RJ12	RS-232 to RS-485	FA-ISOCON with Q8304-1 cable		
P2-SCM	4	4 screw terminals	RS-485	Q8304-1 cable		
P3-SCM	4	4 screw terminals	RS-485	Q8304-1 cable		

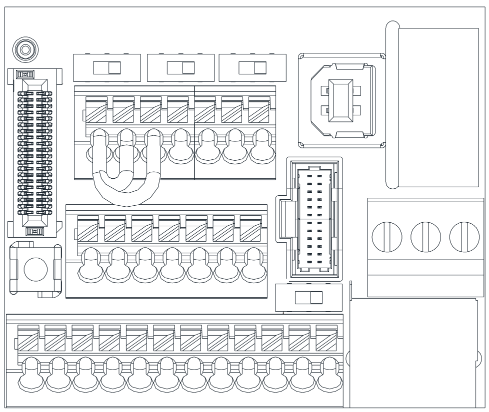
** Ethernet connectivity for EtherNet/IP or Modbus TCP communication is possible with optional communication cards # GS30A-CM-EIP1 or EIP2. Refer to "Appendix B: Optional I/O and Communication Cards" for details.*

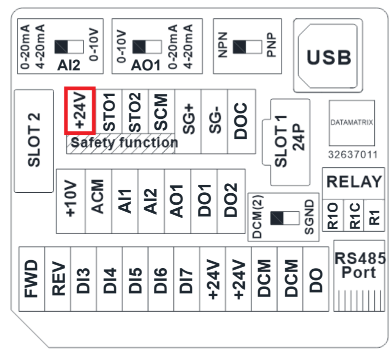
Connecting Communication Cables

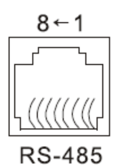
A 120 ohm external terminating resistor is required for the drive end. An external termination resistor may be required on the other end of RS-485 network; especially on long runs. Select resistors that match the impedance of the cable (between 100Ω and 500Ω).

The GS30 serial communication port is an RS-485 input. Please note that terminals SG+ and SG- are shared with the RJ45 connector. That means the user can use standard RJ45 patch cables or industrial RS-485 cabling to access the comm port. GS30 to GS30 serial connections can be accomplished with standard Ethernet patch cables (do not use cross-over cables). RS-232 signals can be converted to RS-485 by using a separate converter (see the FA-ISOCON drawings on page 5–12).

DURApulse GS30 RS-485 Serial Comm Ports







Modbus RS-485
Pin 1, 2, 6: Reserved
Pin 3, 7: SGND
Pin 4: SG-
Pin 5: SG+
Pin 8: +10VS

Note: If using both Modbus connection points (Terminal block and RS-485 Port), ensure you have the same ground reference. Non-equivalent grounding, or grounding from different references, can introduce noise issues that interfere with communications.

Recommended RS-485 cable: Belden 9842, AutomationDirect Q8304-1 series, or equivalent.

Note: When using hardwire terminations for RS-485, you must connect the common wire to the right-hand DCM terminal and set the DIP switch to SGND.

RS-232C to RS-485 Conversion

An RS-485 network cable can span up to 1200 meters (4000 feet). However, many AutomationDirect PLCs have only RS-232C communication ports, and require an FA-ISOCON (RS-232C to RS-422/485 network adapter) in order to make an RS-485 connection.

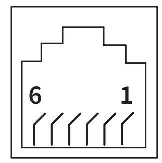
If an FA-ISOCON module is used, set the module dipswitches as required.
Refer to the FA-ISOCON manual for more detailed information.

FA-ISOCON Switch Settings:

- *S21–S23: OFF, ON, ON (19200 baud)*
- *S24–S27: OFF (Automatic Network Transmit Enable)*
- *Terminate: ON (end of run term resistors)*
- *Bias (2): ON (end of run bias resistors)*
- *1/2 DPX (2): ON (RS-485 TXD/RXD jumpers)*

Helpful Hint: Some applications require that the FA-ISOCON baud rate is set faster than the drive/network baud rate.

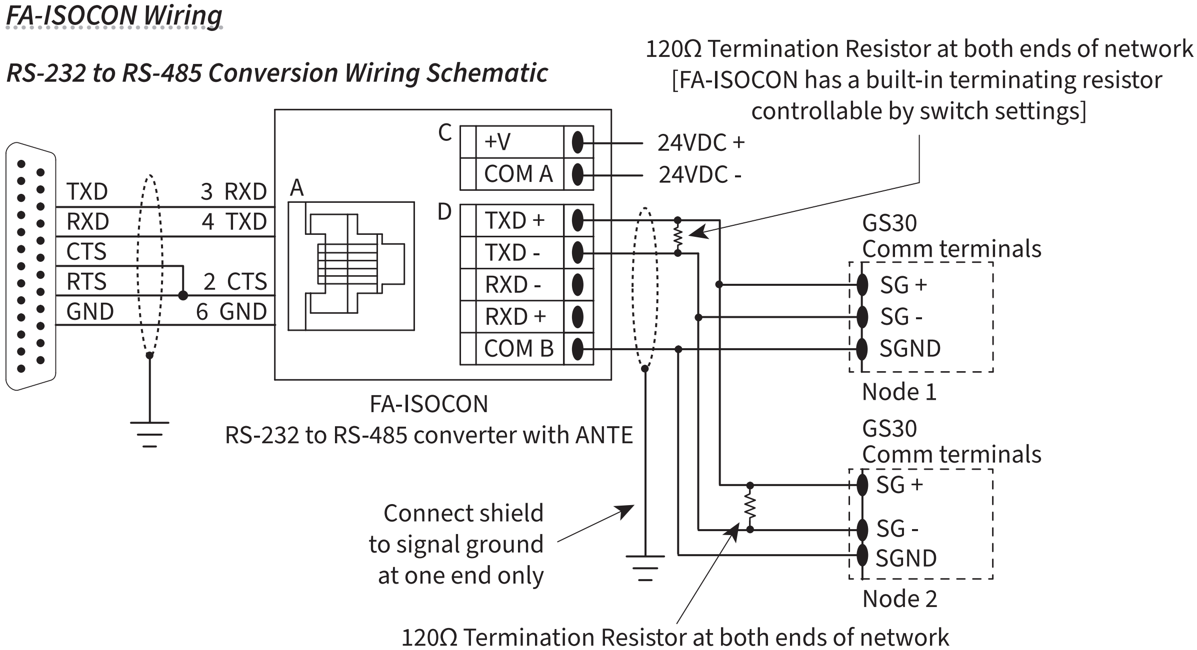
FA-ISOCON RJ-12 Serial Comm Port A
RS-232 Input Port



1: Signal Ground
2: CTS (input)
3: RXD (input)
4: TXD (output)
5: +5VDC in
6: Signal Ground

FA-ISOCON Wiring

RS-232 to RS-485 Conversion Wiring Schematic



Note: When using hardwire terminations for RS-485, you must connect the common wire to the right-hand DCM terminal and set the DIP switch to SGND.

For information regarding configuration of AutomationDirect PLCs or other PLCs, please refer to Appendix D of this user manual, or to the applicable PLC user manual for your application.

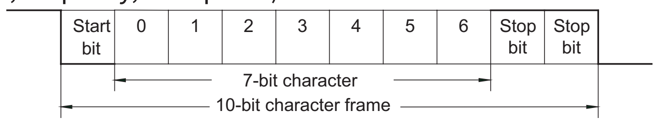
Detailed Serial Modbus Communication Information

The GS30 drive follows the standard Modbus RTU and Modbus ASCII protocols. The following pages provide some brief information on this but if your device does not support these protocols natively and you are required to develop this framework on your own, consult the more detailed documentation at http://www.modbus.org.

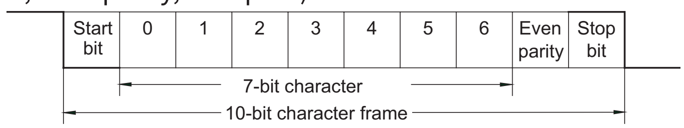
Data Format

ASCII Mode: 10-bit character frame (For 7-bit character):

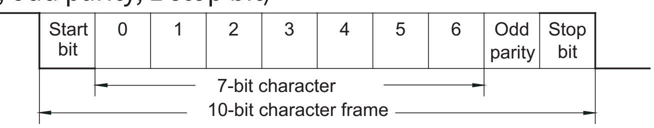
P09.04 = 01 (7 data bits, no parity, 2 stop bits)



P09.04 = 02 (7 data bits, even parity, 1 stop bit)

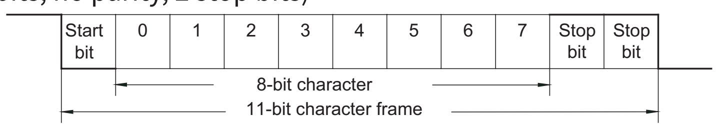


P09.04 = 03 (7 data bits, odd parity, 1 stop bit)

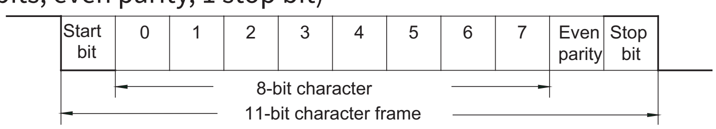


RTU Mode: 11-bit character frame (For 8-bit character):

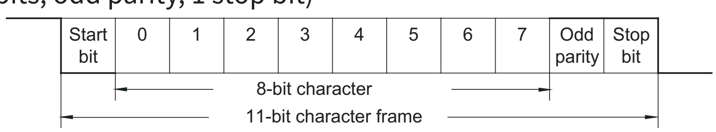
P09.04 = 13 (8 data bits, no parity, 2 stop bits)



P09.04 = 14 (8 data bits, even parity, 1 stop bit)



P09.04 = 15 (8 data bits, odd parity, 1 stop bit)



Communication Protocol

ASCII Mode:

STX	Start Character: (3AH)
ADR 1	Communication Address: 8-bit address consists of 2 ASCII codes
ADR 0	
CMD 1	
CMD 0	
DATA (n-1)	Contents of data: n x 8-bit data consists of 2n ASCII codes. n ≤ 25 maximum of 50 ASCII codes
....	
DATA 0	
LRC CHK 1	LRC check sum: 8-bit check sum consists of 2 ASCII codes
LRC CHK 0	
END 1	END characters: END 1 = CR (0DH); END 0 = LF (0AH)
END 0	

RTU Mode:

START	A silent interval of more than 10 ms
ADR	Communication Address: 8-bit address
CMD	Command Code: 8-bit command
DATA (n-1)	Contents of data: n x 8-bit data, n ≤ 25
....	
DATA 0	
CRC CHK Low	CRC check sum: 16-bit check sum consists of 2 8-bit characters
CRC CHK High	
END	A silent interval of more than 10 ms

ADR (Communication Address)

Valid communication addresses are in the range of 0 to 254. A communication address equal to 0 means broadcast to all AC drives, in which case the drives will not acknowledge any message from the master device.

For example, communication to AC drive with address 16 decimal:

- *ASCII mode: (ADR 1, ADR 0)='1','0' => '1'=31H, '0'=30H*
- *RTU mode: (ADR)=10H*

CMD (Command Code) and DATA (Data Characters)

The format of data characters depends on the command code. The available command codes are described as followed: Command code: 03H, read N words. The maximum value of N is 12. For example, reading continuous 2 words from starting address 2102H of the AC drive with address 01H.

ASCII mode:

Command Message			Response Message	
STX	':'		STX ':'	':'
ADR 1	'0'		ADR 1	'0'
ADR 0	'1'		ADR 0	'1'
CMD 1	'0'		CMD 1	'0'
CMD 0	'3'		CMD 0	'3'
Starting data address	'2'		Number of data (Count by byte)	'0'
	'1'			'4'
	'0'		Content of starting data address 2102H	'1'
	'2'			'7'
Number of data (Count by word)	'0'			'7'
	'0'			'0'
	'0'		Content data address 2103H	'0'
	'2'			'0'
LRC CHK 1	'D'			'0'
LRC CHK 0	'7'			'0'
END 1	CR		LRC CHK 1	'7'
END 0	LF		LRC CHK 0	'1'
			END 1	CR
			END 0	LF

RTU mode:

Command Message			Response Message	
ADR	01H		ADR	01H
CMD	03H		CMD	03H
Starting data address	21H		Number of data (Count by byte)	04H
	02H			'0'
Number of data (Count by word)	00H		Content of data address 2102H	17H
	02H			70H
CRC CHK Low	6FH		Content of data address 2103H	00H
CRC CHK High	F7H			02H
			CRC CHK Low	FEH
			CRC CHK High	5CH

Command code: 06H, write 1 word

For example, writing 6000(1770H) to address 0100H of the AC drive with address 01H.

ASCII mode:

Command Message			Response Message	
STX	':'		STX ':'	':'
ADR 1	'0'		ADR 1	'0'
ADR 0	'1'		ADR 0	'1'
CMD 1	'0'		CMD 1	'0'
CMD 0	'6'		CMD 0	'6'
Data Address	'0'		Data Address	'0'
	'1'			'1'
	'0'			'0'
	'0'			'0'
	'1'		Data Content	'1'
	'7'			'7'
	'7'			'7'
	'0'			'0'
LRC CHK 1	'7'		LRC CHK 1	'7'
LRC CHK 0	'1'		LRC CHK 0	'1'
END 1	CR		END 1	CR
END 0	LF		END 0	LF

RTU mode:

This is an example of using function code 16 for writing to multiple registers.

Command Message			Response Message	
ADR	01H		ADR	01H
CMD	10H		CMD	10H
Starting data address	20H		Starting data address	20H
	00H			00H
Number of registers	00H		Number of data (Count by word)	00H
	02H			02H
Byte count	04H		CRC CHK Low	4AH
Content of data address 2000H	00H		CRC CHK High	08H
	02H			
Content of data address 2001H	02H			
	58H			
CRC CHK Low	CBH			
CRC CHK High	34H			

NOTE Concerning 2100h: When GS30 drive is setup with reference RS-485 (P00.20 = 1 & drive in Remote/Auto) -OR- (P00.30 = 1 & drive in Local/Hand) -AND- Reference > P01.00 Drive Max Out Freq, the GS30 drive goes up to Max Out Freq and remains there until Max Out Freq is modified or a lower Freq Ref or a Stop Command is sent to the drive.

CHK (check sum)

ASCII Mode:

LRC (Longitudinal Redundancy Check) is calculated by summing up module 256, the values of the bytes from ADR1 to last data character, then calculating the hexadecimal representation of the 2's-complement negation of the sum.

For example, reading 1 word from address 0401h of the AC drive with address 01h.

Command Message	
STX	':'
ADR 1	'0'
ADR 0	'1'
CMD 1	'0'
CMD 0	'3'
Starting data address	'0'
	'4'
	'0'
	'1'
Number of data (Count by word)	'0'
	'0'
	'0'
	'1'
LRC CHK 1	'F'
LRC CHK 0	'6'
END 1	CR
END 0	LF

01h+03h+04h+01h+00h+01h=0Ah;
the 2's complement negation of 0Ah is F6h.

RTU Mode:

Response Message	
ADR	01h
CMD	03h
Starting data address	21h
	02h
Number of data (Count by word)	00h
	02h
CRC CHK Low	6Fh
CRC CHK High	F7h

CRC (Cyclical Redundancy Check) is calculated by the following steps:

10) Load a 16-bit register (called CRC register) with FFFFh.
11) Exclusive OR the first 8-bit byte of the command message with the low order byte of the 16-bit CRC register, putting the result in the CRC register.
12) Shift the CRC register one bit to the right with MSB zero filling. Extract and examine the LSB.
13) If the LSB of CRC register is 0, repeat step 3; else Exclusive or the CRC register with the polynomial value A001h.
14) Repeat step 3 and 4 until eight shifts have been performed. When this is done, a complete 8-bit byte will have been processed.
15) Repeat steps 2 to 5 for the next 8-bit byte of the command message.

Continue doing this until all bytes have been processed. The final contents of the CRC register are the CRC value.

When transmitting the CRC value in the message, the upper and lower bytes of the CRC value must be swapped, i.e. the lower order byte will be transmitted first.

The following is an example of CRC generation using C language. The function takes two arguments:

```
Unsigned char* data ← a pointer to the message buffer
Unsigned char length ← the quantity of bytes in the message buffer
   The function returns the CRC value as a type of unsigned integer.
      Unsigned int crc_chk(unsigned char* data, unsigned char length){
         int j;
         unsigned int reg_crc=0xFFFF;
         while(length--){
                  reg_crc ^= *data++;
                  for(j=0;j<8;j++){
                     if(reg_crc & 0x01){  /* LSB(b0)=1 */
                        reg_crc=(reg_crc>>1) ^ 0xA001;
                  }else{
                        reg_crc=reg_crc >>1;
                  }
               }
            }
         return reg_crc;
      }
```

RTU mode is preferred. Limited support is available to ASCII users.