
QuickSilver SilverMax Servomotor

Application Note

This document describes how to configure a Paradigm operator interface terminal to allow communications with a QuickSilver SilverMax Servomotor. The communications protocol supports access to immediate and program commands. Please read this document carefully before attempting to configure communications with these devices.

Introduction

The EDICT-97 configuration software has been designed to allow the user to enter a Parameter mnemonic and number in a manner that should be familiar to a user of a QuickSilver SilverMax Servomotor. The driver allows the exchange of data with the Servomotor.

Accessing Data

The QuickSilver SilverMax Servomotor communications protocol allows access to a number of Parameters over a serial communications link. The driver described here supports a subset of these parameters and these are given in the table below.

NOTE: Version numbers higher than 2.04 have had the data register access RRG and WRI combined into one command REG, which executes the function indicated by the desired operation. The programmer MUST be aware of which registers are read-only, and which registers are write-only, and assign them to Comms Blocks in order to avoid creating unintended communication errors.

NOTE: The 57.6K Baud Driver has its data format parameters fixed at 8N1 57600. Changing the values displayed in the configuration will have no effect on the program. This driver functions on the Paradigm Series CL010050 or CL010060, only.

NOTE: THE FOLLOWING IS IMPORTANT FOR PROPER OPERATION.

Numerous commands depend upon the command parameters associated with them. The associated parameters are those that have 4 letter identifiers. The first three letters match those of the command, such as: MATP, MATA, MATT, MATE, MATS -- all associated with the command MAT. The MAT command in the Quicksilver requires those associated parameters be sent in the serial command, so it is imperative that the programmer initialize those parameters before executing the command.

The correct way to program all of the 3 letter (actual command) selections of these types (MAT, MAV,... XRV) is by using a Comms Block whose Access is set to PULSE. The data values can be programmed in any order either in another Comms Block or as a Direct Reference. Any values written to, or read from, them reference internal memory and are not transmitted to the drive, except by activating the 3 letter command.

When the 3 letter command (e.g. MAV set up in Comms Block Z[5]) has to be executed, the programmer creates the Action Z[5] := 1. As long as Comm Block Z, in this example, is set up for access PULSE, as instructed, the command will be executed properly with the stored values for the MAVP, MAVA, MAVV, MAVE, and MAVS items transferred to the drive.

OTHER COMMANDS THAT REQUIRE USING THE PULSE MODE IN COMMS BLOCKS:

Motor commands, items that do not take data, but instruct the motor to perform an action also require being placed in a Comms Block with Access set to PULSE. These commands are:

COB, CIS, CME, DDB, DEM, DLC, DMD, DMT, EDB, EDH, EEM, EMD, EMT, END, GCL, GOP, HLT, HSM, KDD, KED, MDC, PCG, PCM, PMX, RST, RUN, SLC, SSP, SSE, TTP, WDL, ZTG, ZTP

Execute these commands in a manner similar the example MAV above. It is acceptable, and recommended, to place all of these commands in one Comms Block. Map the Comm Block location to a meaningful name in Named Data, and use "Name := 1" to execute the command.

PARAMETER INFORMATION TABLES

| Parameter | Data Type | COMMAND NAME | COMMAND PARAMETERS | Range | Access |
|-------------------|-----------|--------------|--------------------|--------------------|--------|
| Status Commands | Immediate | CPL | | 0 to 65535 | Write |
| | Immediate | POL | | | Read |
| | Immediate | RIO | | | Read |
| | Immediate | RIS | | | Read |
| | Immediate | RPB | Length Address | 1 to 8 0 to 199 | Read |
| Override Commands | Immediate | SSP | | | Write |
| | Immediate | SSE | | | Write |
| | Immediate | STP | | -1 to 536,870,911 | Write |
| | Immediate | RST | | | |

| Parameter | Data Type | COMMAND NAME | COMMAND PARAMETERS | Range | Access |
|--------------------------------|-----------|--------------|---|---|--------------|
| Motion & Profile Move Commands | Program | MAT | Position Acceleration Time Total Time Stop Enable Stop State | +/- 2,147,483,648 0 to 65534 2 to 2,147,483,647 0 to 65535 0 to 65535 | Read / Write |
| | Program | MAV | Position Acceleration Velocity Stop Enable Stop State | +/- 2,147,483,648 1 to 1,073,741,823 0 to 2,147,483,647 0 to 65535 0 to 65535 | Read / Write |
| | Program | MRT | Position Acceleration Time Total Time Stop Enable Stop State | +/- 2,147,483,648 0 to 65534 2 to 2,147,483,647 0 to 65535 0 to 65535 | Read / Write |
| | Program | MRV | Distance Acceleration Velocity Stop Enable Stop State | +/- 2,147,483,648 1 to 1,073,741,823 0 to 2,147,483,647 0 to 65535 0 to 65535 | Read / Write |
| | Program | RAT | Data Register Acceleration Time Total Time Stop Enable Stop State | 10 to 40 0 to 65534 2 to 2,147,483,647 0 to 65535 0 to 65535 | Read / Write |
| | Program | RAV | Data Register Acceleration Velocity Stop Enable Stop State | 10 to 40 1 to 1,073,741,823 0 to 2,147,483,647 0 to 65535 0 to 65535 | Read / Write |
| | Program | RRT | Data Register Acceleration Time Total Time Stop Enable Stop State | 10 to 40 0 to 65534 2 to 2,147,483,647 0 to 65535 0 to 65535 | Read / Write |

| Parameter | Data Type | COMMAND NAME | COMMAND PARAMETERS | Range | Access |
|--------------------------------|-----------|--------------|--|--|--------------|
| Motion & Profile Move Commands | Program | RRV | Data Register Acceleration Velocity Stop Enable Stop State | 10 to 40 1 to 1,073,741,823 0 to 2,147,483,647 0 to 65535 0 to 65535 | Read / Write |
| | Program | XAT | Starting Data Register Stop Enable Stop State | 10 to 40 0 to 65535 0 to 65535 | Read / Write |
| | Program | XAV | Starting Data Register Stop Enable Stop State | 10 to 40 0 to 65535 0 to 65535 | Read / Write |
| | Program | XRT | Starting Data Register Stop Enable Stop State | 10 to 40 0 to 65535 0 to 65535 | Read / Write |
| | Program | XRV | Starting Data Register Stop Enable Stop State | 10 to 40 0 to 65535 0 to 65535 | Read / Write |

| Parameter | Data Type | COMMAND NAME | COMMAND PARAMETERS | Range | Access |
|-----------------------|-----------|--------------|--|--------------------------------------|--------------------------------------|
| Program Flow Commands | Program | CLC | Operation Data Register | 0 to 16 0 to 40, 200 & 201 | Read / Write |
| | Program | DLY | Tick Count | 0 to 4,294,967,295 | Read / Write |
| | Program | RUN | | | Write |
| | Program | PCI | Program Buffer Loc. I/O Bit Enable I/O Bit State | 0 to 199 -1 to -14 0 or 1 | PCI=Command PCIx=R/W |
| | Program | PCL | Program Buffer Loc. Condition Enable Condition State | 0 to 199 0 to 65535 0 to 65535 | PCL=Command PCLx=R/W |
| | Program | PRT | Condition Enable Condition State | 0 to 65535 0 to 65535 | PRT=Command PRTx=R/W |
| | Program | WBE | Input Code Input Transition | 1 to 14 0 or 1 | WBE=Command WBE _x =R/W |
| | Program | WBS | Input Code Input State | 1 to 14 0 or 1 | WBS=Command WBS _x =R/W |

| Parameter | Data Type | COMMAND NAME | COMMAND PARAMETERS | Range | Access |
|------------------------|-----------|--------------|------------------------------|---|--------------|
| Data Register Commands | Program | ATR | Data Register Data to Add | 10 to 40 +/- 2,147,483,648 | Read / Write |
| BEFORE v 2.04 | Program | RRG | Data Register | 0 to 40 & 200 to 205 | Read / Write |
| BEFORE v 2.04 | Program | WRI | Data Register Data | 10 to 40 +/- 2,147,483,648 or 0 to 4,294,967,295 | Read / Write |
| | Program | WRP | Data Register Data | 10 to 40 0 to 4,294,967,295 | Read / Write |
| Version 2.04+ | Program | REG | Data Register | 0-255 | Read/Write |

| Parameter | Data Type | COMMAND NAME | COMMAND PARAMETERS | Range | Access |
|---------------------------------|------------------|---------------------|---------------------------|--------------|---------------|
| Miscellaneous Function Commands | Program | CIS | | | Write |
| | Program | ZTG | | | Write |
| | Program | ZTP | | | Write |

| Parameter | Data Type | COMMAND NAME | COMMAND PARAMETERS | Range | Access |
|------------------|------------------|---------------------|------------------------------|--------------------|-------------------------|
| I/O Commands | Program | ACR | Analog Channel Data Register | 0 to 8 10 to 40 | ACR=Command ACRx=R/W |
| | Program | ARI | Analog Channel Data Register | 0 to 8 10-40 | ARI=Command ARIX=R/W |
| | Program | CIO | Setting | -1 or 0 or 1 | Write |
| | Program | COB | | | Command |

Examples

Read Polling Status Word:

[POL]

Write to Clear the Polling Status Word:

[CPL] := 65535

Write to Stop Immediate:

[STP] := -1

Write Register, Immediate Type:

[REG(20)] := -6000, [REG(21)]:=6942, [REG(22)]:=69442

Knowledge of Unit Operation Is Assumed

In all cases, the simple principle of 'pass-through' is maintained: there is no attempt to validate a value in terms of the end use of the unit: both familiarity with the control functions and knowledge of system operation are assumed.

Communications

Communications with the QuickSilver SilverMax Servomotor is via an RS-232 point to point link, or RS-485 multi-drop. The default serial communications format is RS-485, baud rate of 19200, 8 data bits, No parity, 1 stop bit, and address 1.

The connections details are described in the table below.

RS232

| Paradigm unit (RS 232 port) | SilverMax Servomotor COMM Connector (Female DB-15) |
|------------------------------------|---|
| Pin 1 (Tx) | Pin 12 (RX) |
| Pin 2 (Rx) | Pin 2 (TX) |
| Pin 3 (RTS) | |
| Pin 4 (CTS) | |
| Pin 5 (0v) | Pin 8 (GND) |

In addition a link must be fitted between Pin 3 (RTS) and Pin 4 (CTS) on the Paradigm unit.