Control Technology Corporation

Model 2601 Automation Controller Installation Guide

Model 2601 Automation Controller

This document is current as of the following revision levels:

- Controller Firmware – 2.24
- Controller Hardware – B

This guide describes how to install a 2601 Automation Controller and connect inputs and outputs.

Contents

Control Technology Corporation, 25 South Street, Hopkinton, MA 01748
Doc. No. 2601-IG
Rev. C 5/00

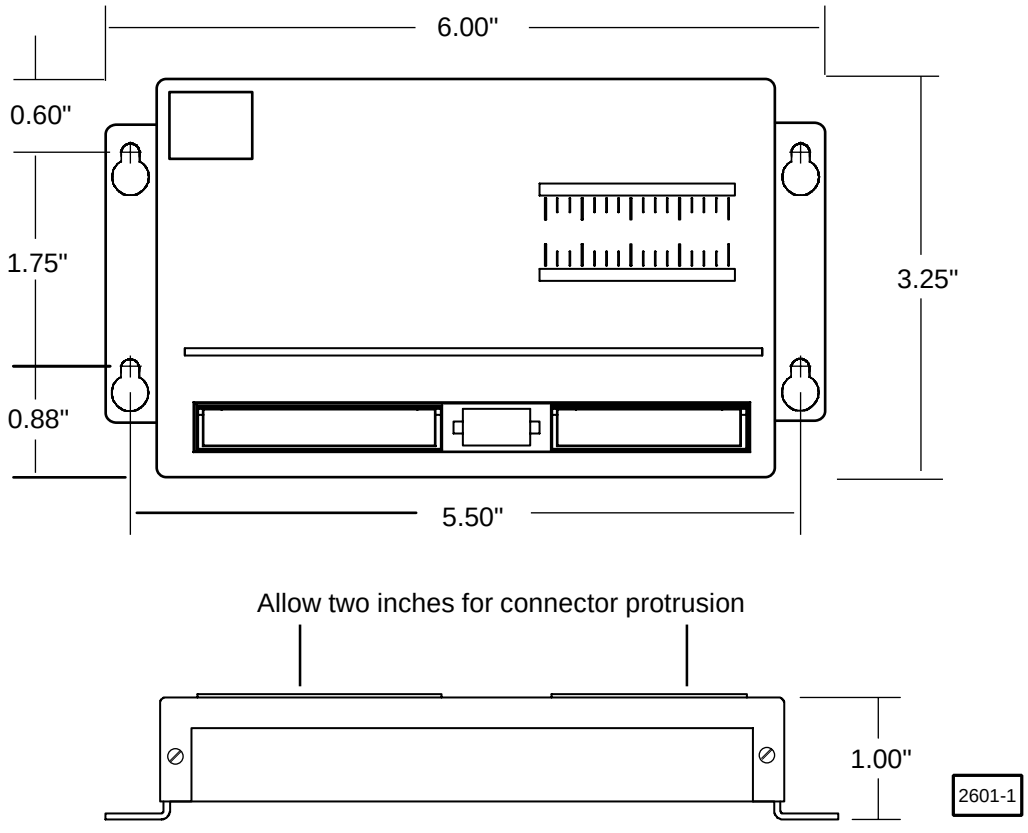
Mounting the Model 2601

The model 2601 Automation Controller is provided with mounting ears, allowing them to be easily mounted to a flat surface (for example, a NEMA-rated electrical enclosure) with four mounting bolts. You should follow the guidelines described in this chapter to ensure a successful design.

Mounting Considerations

When selecting a mounting location for the controller, care should be taken to provide protection against various environmental factors:

- The controller should not be exposed to flying metal chips (be careful during installation and subsequent machine construction work!), conductive dusts, liquids or condensing humidity. In environments where these hazards may be present, the controller should be housed in a NEMA 4 or NEMA 12 rated enclosure, as appropriate.
- The controller is not intended for mounting in an environment requiring explosion-proof practices.
- If possible, the controller should be mounted physically distant from devices producing Electromagnetic Interference (EMI) or Radio Frequency Interference (RFI). This includes motor starters, relays, large power transformers, ultrasonic welding apparatus, etc.



Model 2601 Mounting Dimensions

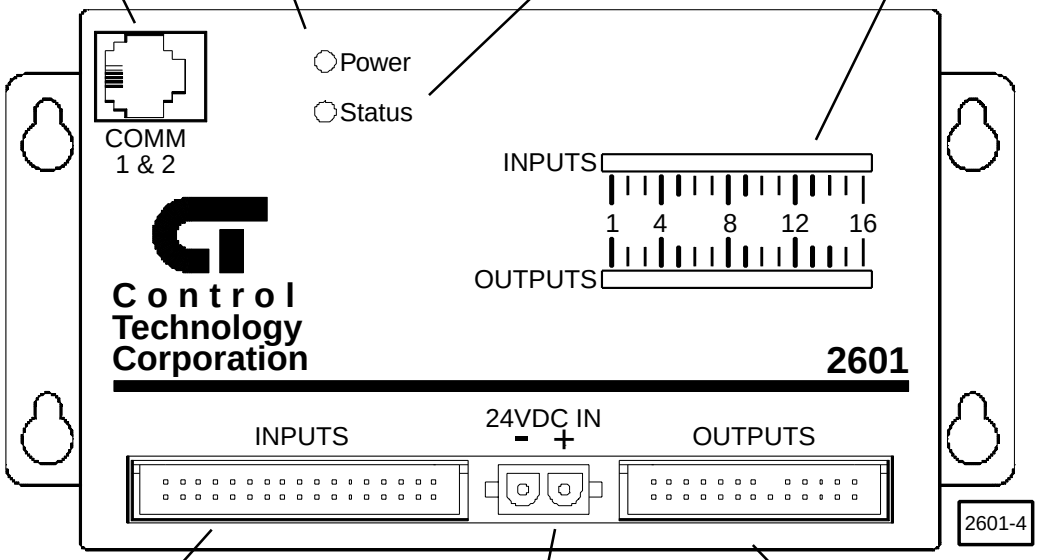
Model 2601 Overview

RS-232 Communications Port Connections – Provides both programming and data communications via a personal computer using Quickstep™. To connect to the communications ports use a model 2886 adapter. See page 5 for the pin connections for the 2886. For more information on controller – RS-232 communications, refer to, *Computer Based Programming and Communications*, in this installation guide.

Power Indicator – Indicates when there is power coming into the controller. This light is lit when the power is on whether or not the controller is in operation. This indicates that the logic supply is present.

LED Indicators - Indicates when an input or output is active. Each input and output has an LED indicator that the controller lights up when it is active (i.e.; on).

Status Light – Indicates a hardware or software faults. Refer to the section *Status Light Description* for additional information.

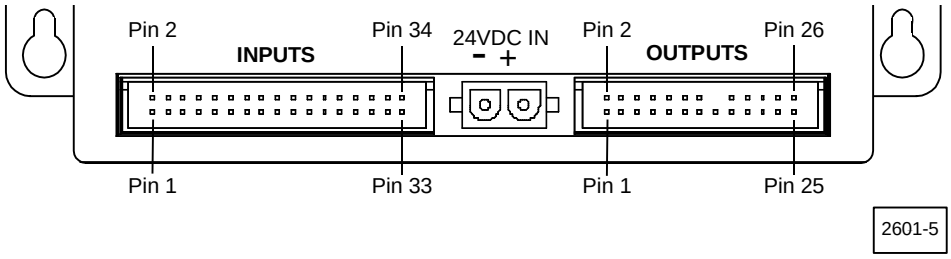


Input Connector - Provides access to the model 2601's 16 inputs. The inputs are numbered 1 through 16. To connect to the inputs use a model 2676 Input Pigtail Cable, 2653 connector kit, or 2376 terminal block. You can also create your own cable with an AMP housing #102387-8 with 87523-6 contacts. Another alternative is to mate a ribbon cable connector with this input connector (for example, 3M part #3414-6000).

Output Connector - Provides access to the model 2601's 16 outputs. The outputs are numbered 1 through 16. To connect to the outputs use a model 2675 Output Pigtail Cable, 2653 connector kit, or 2375 terminal block. You can also create your own cable with an AMP housing #102387-6 with 87523-6 contacts. Another alternative is to mate a ribbon cable connector with this input connector (for example, 3M part #3399-6000).

Power Connector – The power connector provides 24 VDC power to the controller. Use a model 2685 Pigtail Cable or a 2695 Connector Kit. You can also create your own with a AMP Pin model No. 350550-1 and AMP Connector model No. 350777-1.

Pin Connections for Inputs and Outputs



Input Connections

Pin No.	Signal	Pin No.	Signal
1	Input No. 1	2	Return[1]
3	Input No. 2	4	Return
5	Input No. 3	6	Return
7	Input No. 4	8	Return
9	Input No. 5	10	Return
11	Input No. 6	12	Return
13	Input No. 7	14	Return
15	Input No. 8	16	Return
17	Input No. 9	18	Return
19	Input No. 10	20	Return
21	Input No. 11	22	Return
23	Input No. 12	24	Return
25	Input No. 13	26	Return
27	Input No. 14	28	Return
29	Input No. 15	30	Return
31	Input No. 16	32	Return
33	+ 24 VDC[2]	34	Return

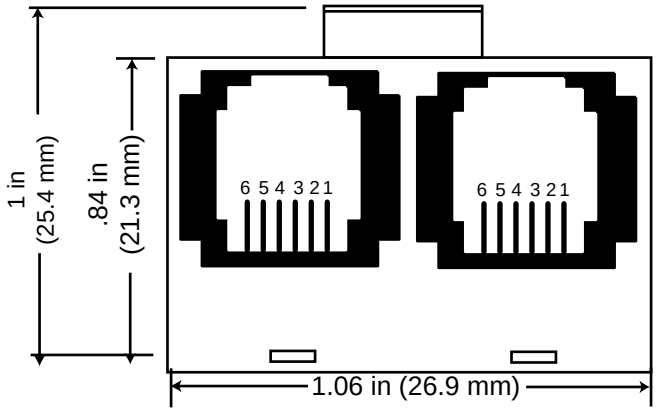
Output Connections

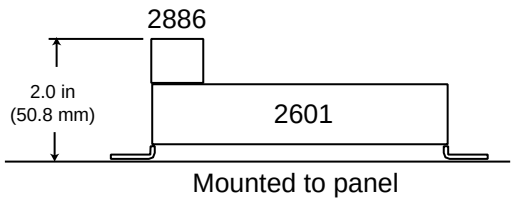
Pin No.	Signal	Pin No.	Signal
1	Output No. 1	2	Output No. 14
3	Output No. 2	4	Output No. 15
5	Output No. 3	6	Output No. 16
7	Output No. 4	8	Return[1]
9	Output No. 5	10	Return
11	Output No. 6	12	Return
13	Output No. 7	14	Return
15	Output No. 8	16	No pin
17	Output No. 9	18	+ 24 VDC[2]
19	Output No. 10	20	+ 24 VDC
21	Output No. 11	22	+ 24 VDC
23	Output No. 12	24	+ 24 VDC
25	Output No. 13	26	N/C

1. Common to 24 volt power supply return
2. Connected to 24 VDC

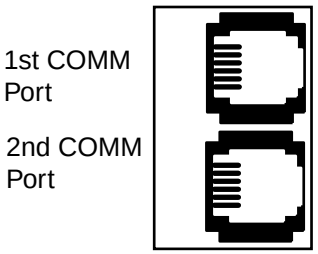
NOTE: This is the same pin configuration as Control Technology's Model 2203 Input/Output Module.

Model 2886 Duplex Adapter

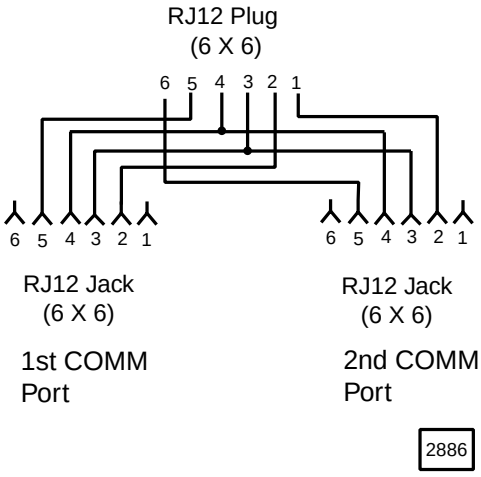




Orientation of connector
when plugged into 2601



Wiring Diagram



2886

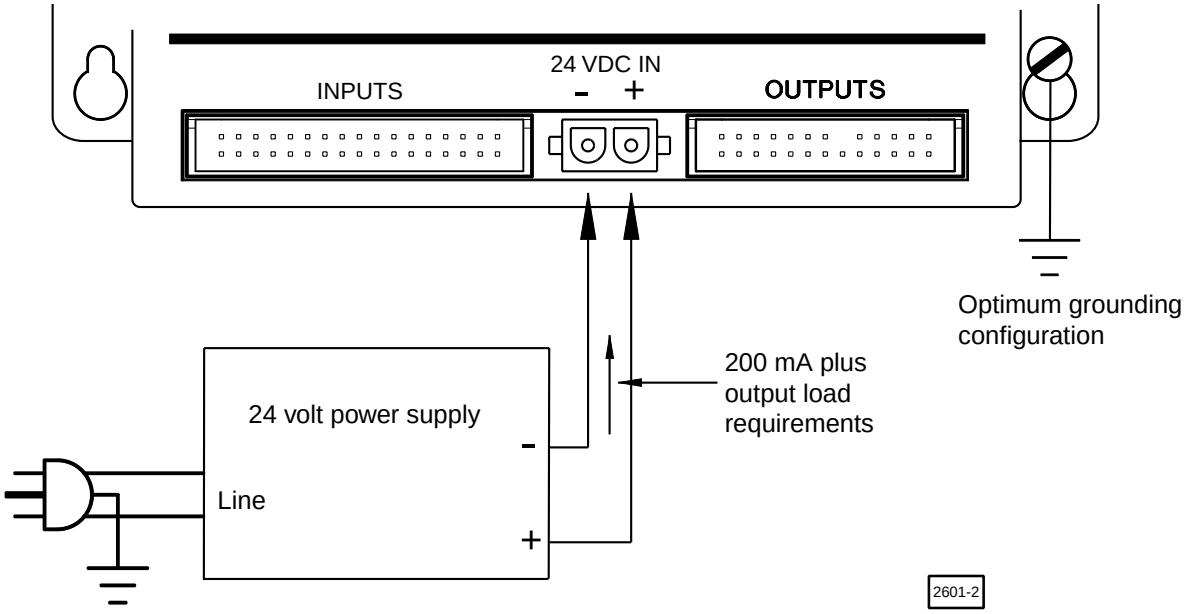
2601 Specifications

Description	Min.	Typical	Max.	Units
Temperature Ratings				
Ambient Temperature:				
Operating	0		+50	°C
Storage	-20		+80	°C
Electrical Specifications				
Applied Power Supply Voltage (V_{PS})	18	24	27	VDC
Applied Output Voltage[2]	0		V_{PS}	VDC
Output Current (DC)				
Single Output			500	mA
Total Limit (All Outputs Combined)			5	A
Current Requirement (V_{PS} = 24V)	50	100	210	mA
User Memory Capacity (10 year battery-backed RAM)			24K	Bytes
Input Off Voltage (I_{in} = 0 mA)		V_{PS}		VDC
Input On Current (V_{in} = 0V, V_{PS} = 24V)		-2.3	-3.0	mA DC
Input On Current Threshold (V_{PS} = 24V)		-1.35	-1.75	mA DC
Input Off Current (Typical Leakage Allowed)			-250.0	µA DC
Output On Voltage (I_o = 500 mA)		0.8	1.2	VDC
Output Off Leakage (Applied Voltage = 24V)		0.01	0.75	µA DC
RS-232 Operating Characteristics				
RS-232 Transmitters		± 9	± 12	
RS-232 Receivers	± 3		± 12	VDC
Common Mode Voltage Range	-10.0		+10.0	VDC
Controller Resource Summary				
Multi-tasking (Tasks)			28	
Volatile Registers (32-bit)			488	
Non-volatile Registers (32-bit)			500	
Data Table Elements (16-bit, non-volatile)			8000+	
Input-linkable Counters			8	
Flags			32	
Program Steps			1024	

Notes:

1. Under normal operation, no external input voltage should be applied. Inputs should be externally switched to the input common.
2. An on-board protection diode returns to +24 V from each output.
3. All power requirements are worst case, with all inputs and outputs activated.

Connecting Power to the Model 2601



WARNING! Do not reverse polarity.

Status Light Description

The status light on the model 2601 can indicate one of the following conditions:

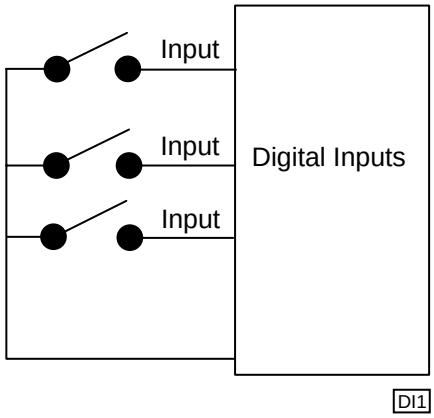
- Software Fault: A non-periodic flashing light indicates the absence of a working Quickstep program. To correct this condition, download a new program. While in this mode the controller's outputs are disabled and cannot be turned on via the RS-232 port.
- Software Fault: A periodic flashing light is a program software fault. This means the controller was unable to execute due to an application problem within the program. To determine the type of software fault you can view the program status using Quickstep's program monitoring utility. If a program software fault occurs, the controller is idle and all setable resources (e.g., outputs, registers and flags) are left in the state they were in prior to the software fault. You can program register 13009 to turn off a specific output in the event of a software fault. Refer to the list of special purpose registers included with this installation guide for more information.
- Hardware Fault: A steady red light indicates that the internal watch dog timer has disabled the controller's CPU. If this occurs, the controller's outputs are also disabled. Try cycling the power, re-downloading your Quickstep program, or both. If the fault continues, your controller may have to be returned to Control Tech. for repair. Contact Control Tech. technical support prior to returning your controller for further details.

Connecting Inputs

Connecting Inputs

The 16 inputs require only a switch closure to the **Return** (the common for the controller's 24 Volt supply) to actuate. Each input is internally self-powered from the 24 Volt power supply through a current limiting resistor, and is optoisolated from the controller's logic.

The controller senses when any of the inputs have been pulled down to return by a switch closure, and a Monitor instruction or any other programmed instruction referring to a general-purpose input can use this information.



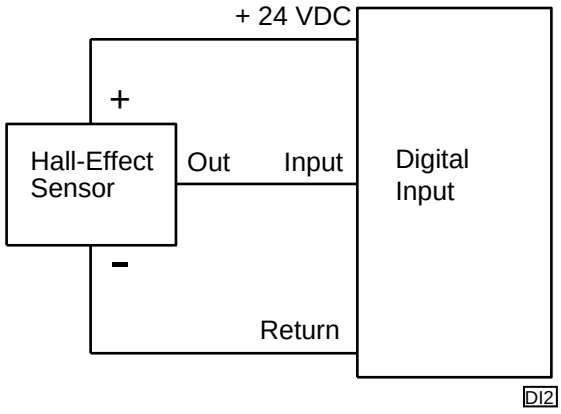
Inputs require only a switch closure to the return point

Using Solid State Sensors

You can connect many types of electronic sensors to the inputs. You can connect three wire Hall-effect sensors, proximity sensors, and phototransistors without any additional circuitry. These devices should be specified as having sinking-type open-collector outputs (NPN) and must be capable of withstanding at least +24 volts on their output terminals when in the off state.

NOTE: Do not use two-wire solid state sensors.

Electronic sensors typically require an external power source for powering their internal circuitry. The following illustration shows how to connect a solid state sensor.



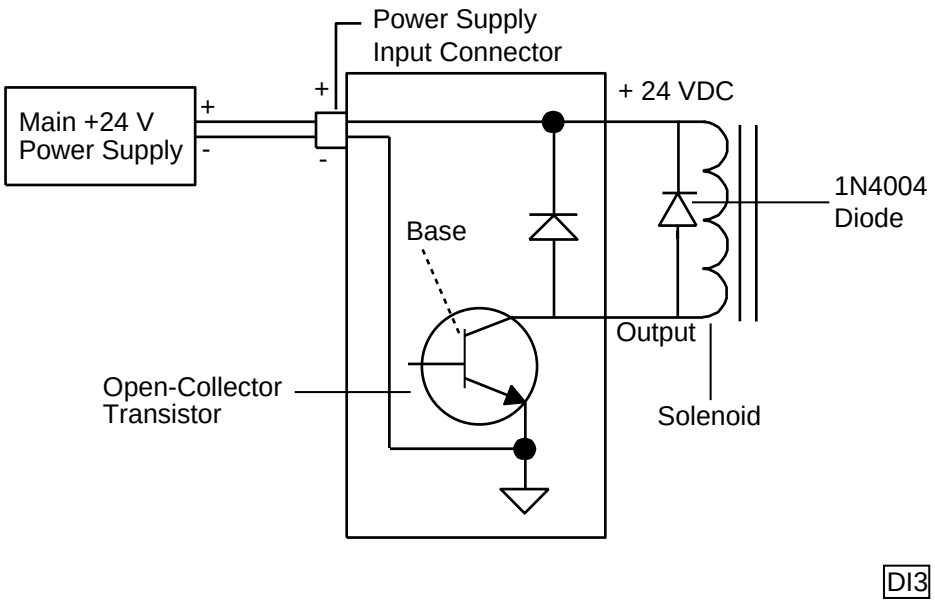
Connecting Outputs

Using Open-collector Outputs

The model 2601 provides 16 outputs for driving external loads, such as solenoid valves, indicators, solid-state relays and other low-power DC loads. These outputs are in the form of open-collector transistors capable of switching loads up to 0.5 Amp DC

This type of output gets its name from the fact that the collector terminal of the output transistor is left open, or unconnected, to allow greater flexibility in its use.

An open-collector output, shown schematically below, performs roughly the same function as a switch contact with one side of the switch connected to ground. When the output is turned off, no current can flow through the transistor. This is the equivalent of the switch contact being open, because the device being controlled is turned off.



When the output is turned on, current is allowed to flow through the transistor, just as though a switch contact had been closed. The controlled device turns on in response to the flow of current.

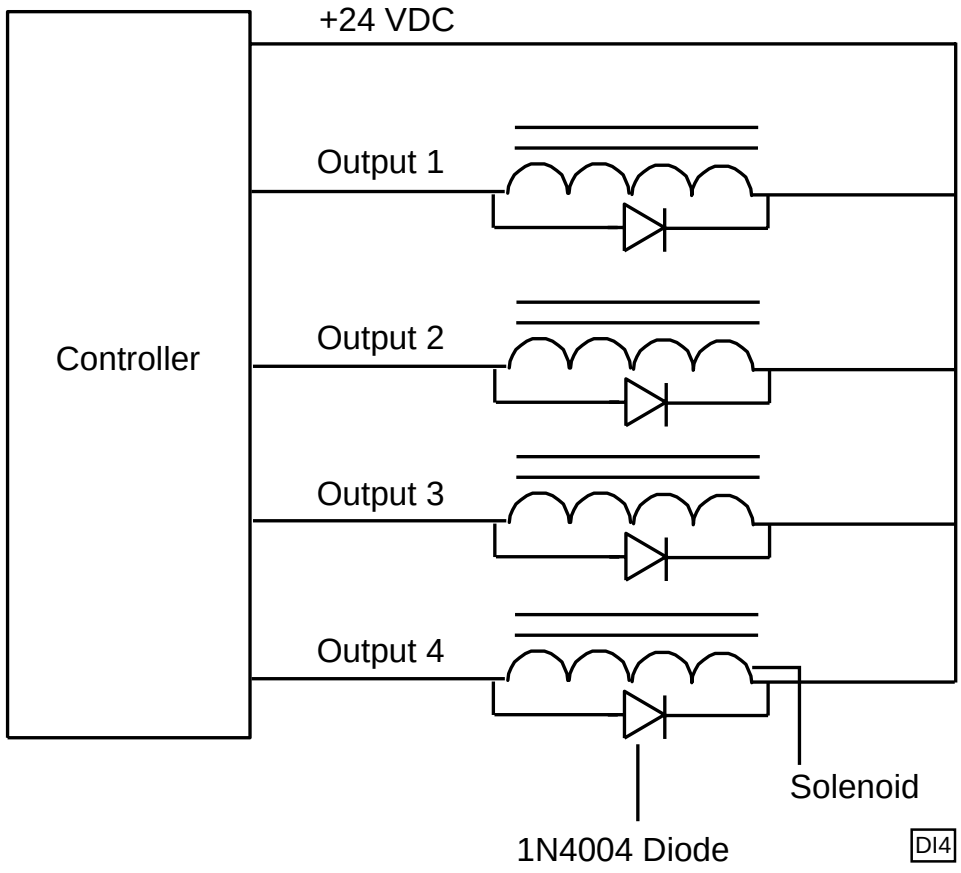
To connect a device to an open-collector output, one terminal of the device is connected to the open-collector output (if the device is polarized, the negative [-] terminal is connected to the output). The remaining terminal of the device is connected to the positive side of the power supply.

IMPORTANT! Control Tech recommends that you place a suppression diode across inductive loads. Use a 1N4004 diode or equivalent. The diode should go as close to the load as possible, as shown in the illustrations.

Care should be taken not to exceed the rated current of the power supply being used. When calculating the current requirements of your system, you only need to consider the maximum number of output devices to be turned on simultaneously plus .2 AMPs for the Model 2601 controller in your calculation.

Connecting Multiple devices

When powering multiple devices from the same power source, each device is connected with one of its leads attached to an independent output, and the other lead connected to the positive terminal of the power source. The following diagram shows four solenoid valves being controlled by outputs 1 through 4. All outputs are powered by the power supply which is powering the controller.

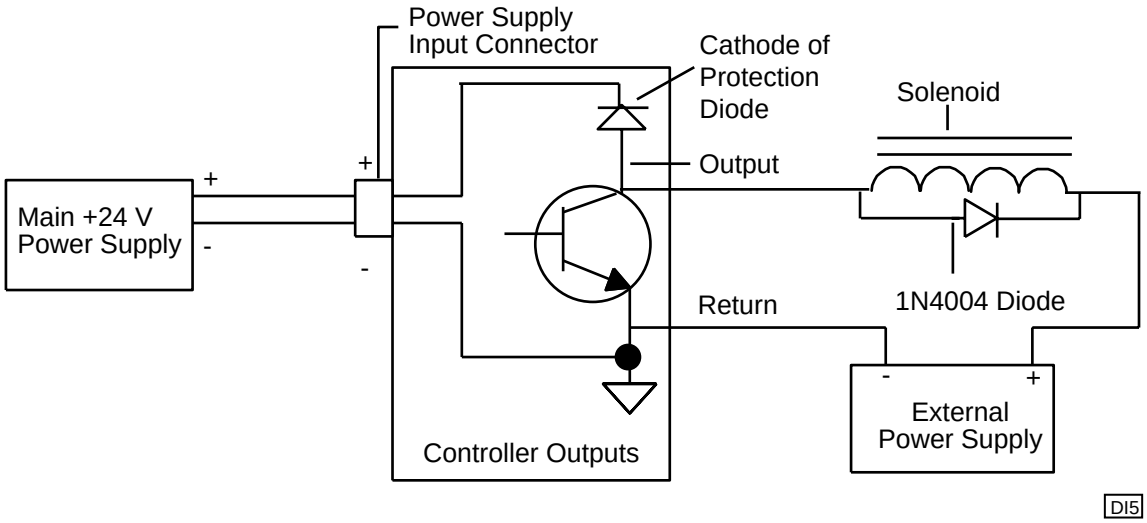


Connecting multiple devices to the controller inputs

Connecting to a Second External Power Supply

It is also possible to power some of the devices from a second external power supply, while powering others from the supply powering the controller. To do this, you must connect each device being controlled to the positive terminal of the appropriate power supply. See the diagram below. For an illustration show how to connect multiple devices to an external power supply, see the diagram on the next page. When connecting to an external power supply, do not to connect the positive terminals of the two supplies together, either directly or indirectly.

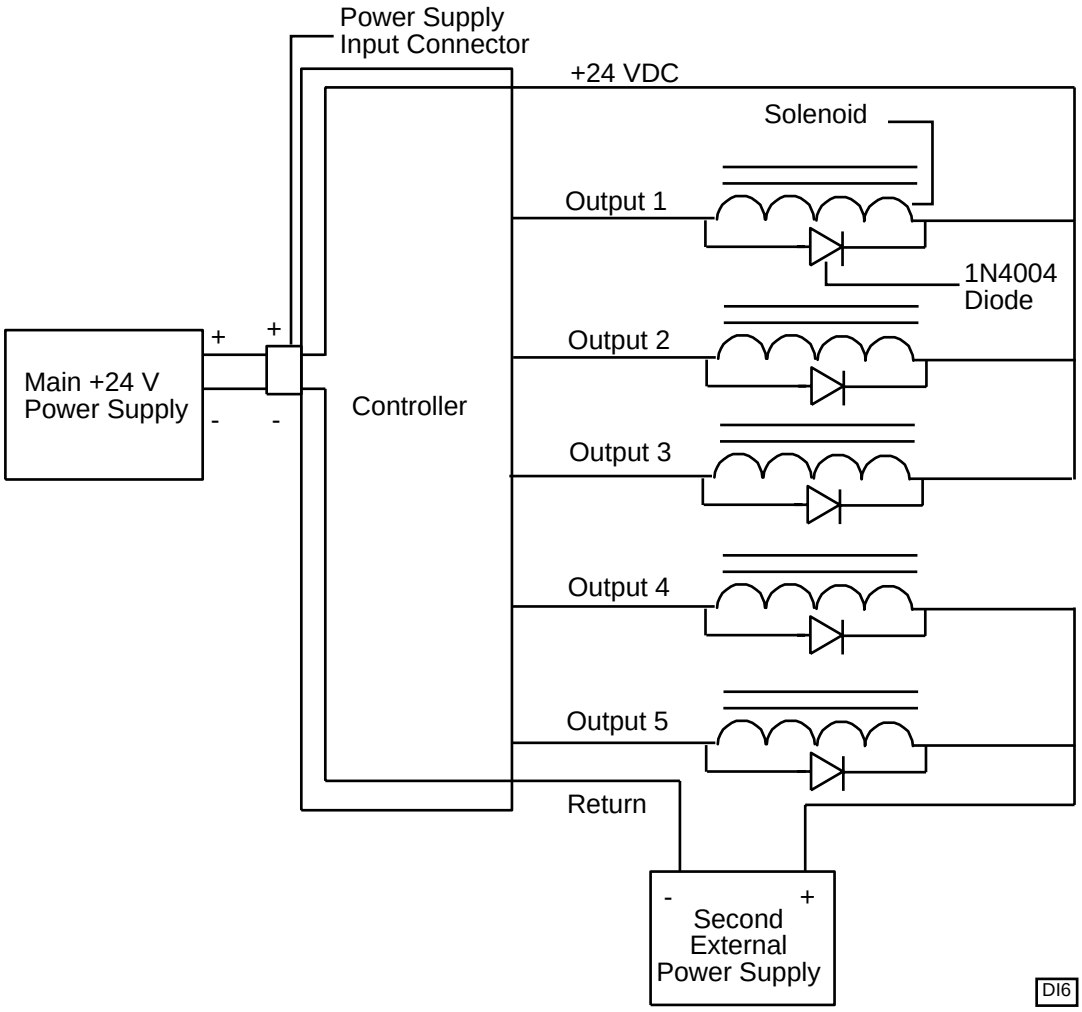
IMPORTANT! Control Tech recommends that you place a diode across inductive loads. Use an IN4004 diode or equivalent. The diode should go as close to the load as possible, as shown in the illustrations.



Open Collector Outputs Using an External Power Supply

WARNING! Each output has a protection diode with its cathode connected to the + 24 VDC power supply at the input connector. This diode prevents damage to the output when connected to an inductive load. If a separate power supply is used for the external devices, as shown above, a current path exists between the two supplies through the devices being controlled. Under normal operation this practice is okay. However some power supplies when powered down, tend to offer a low impedance with respect to power supply return. If in the above configuration, the main power supply is powered down and the external one is not, the current from the external supply can energize the device connected to the output, turning it on. To prevent this, make sure that both supplies are powered up and down together.

In the diagram below the connection between the negative [-] terminal of the external power supply and the return terminal on the controller's output connector. This is necessary to provide a complete circuit for the current travelling through the device being controlled.



Using a second external power supply with multiple devices

IMPORTANT! Do not use an external power supply with an output voltage greater than the output voltage rating of the outputs.

Do not connect the positive [+] terminals of the power supplies together! Damage to one of the supplies may result.

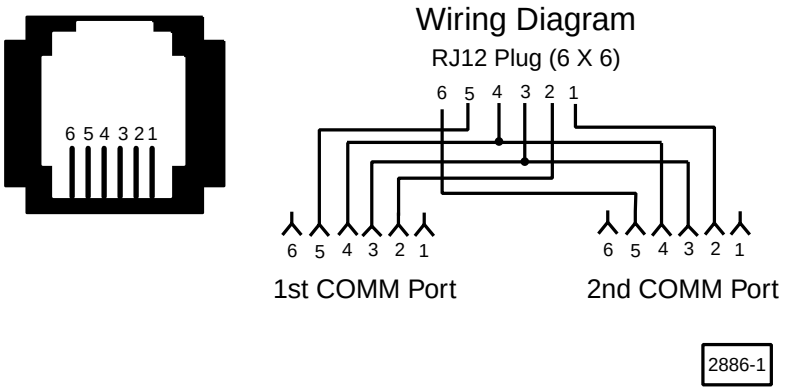
Computer Based Programming and Communications

The RS-232 port on your model 2601 controller provides a means for both programming and data communications via a personal computer using Quickstep™ (Control Tech.'s personal-computer-based programming software). The controller is also equipped with a built-in protocol allowing direct computer communications with the controller's RS-232 port. This protocol is described in the *Guide to CTC Serial Data Communications*. It allows an external computer to directly interact with many of the controller's resources (i.e.; counters, registers, I/O, flags, etc.), without modifying the controller's program.

This communications capability facilitates the use of the model 2601 controller in an intelligent factory environment, since it can derive parametric or production information from a central data store (computer) and supply status and quality control data for centralized reporting.

RS-232 Connections

The connection to the controller's RS-232 ports is made via a modular jack on the controller (labelled "COMM"). This jack carries the receive signal, two grounds, and the transmit signal for the communications channel. The pin connection diagram illustrates the wiring of the jack.



To connect to this jack you need a 2886 duplex addapter and a 2881, 2882, or 2883 Control Tech. cable. The 2886 duplex adapter is shown on page 5, and the cables are show in the diagram on the following page. As an alternative to using the 2881, 2882, or 2883 cable, you may substitute one of the commonly-available telephone cables.

NOTE: Do not connect the controller to a telephone line.

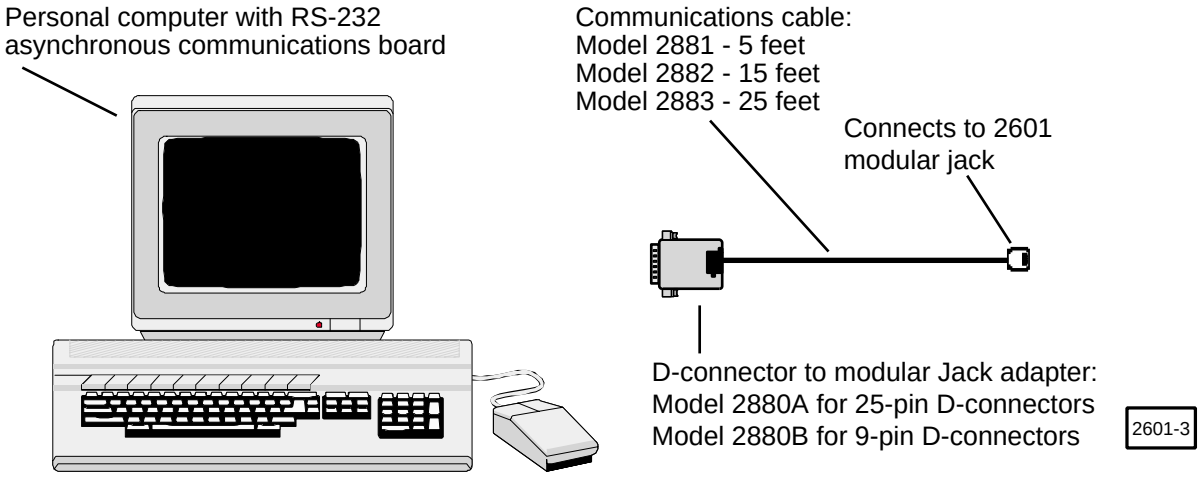
Selecting a Communications Port

The value in register 12000 indicates which communications port the 2601 should use. Writing zero to register 12000 selects COMM port 1 and writing one selects COMM port 2. The default is zero. Once the a port is selected, the same communications register may be used to read the transmission status.

Connecting to a D Connector

RS-232 ports on computers are frequently brought out through 25-pin or 9-pin D type connectors. There is a standard for wiring such connectors, followed by IBM and many other PC manufacturers.

Control Tech. has adapters available, the model 2880A or -B, that connect directly to a male 25-pin (-A version) or 9-pin (-B version) D connector. These adapters provide a modular jack wired for compatibility with the COMM port. To be fully compatible when using this adapter, the computer's communications port should be wired as a DTE device: pin 2 = TxD, pin 3 = RxD, pin 7 = Ground.



Appendix: Register List for the Model 2601 Controller

Contents

General Purpose Registers

General purpose registers within Control Tech.'s controllers are 32-bit data areas and may be accessed for a variety of reasons such as, counting, math calculations (ranging), data storage, preventive maintenance processing, etc. You can access register data from any program step using a STORE instruction and test the value in a register using an IF instruction.

NOTE: The R designation means the register may be Read by the controller. The W designation means the register may be Written to.

Registers 1 - 8: Counters

R/W Counters may also be used as general purpose registers.

You can assign the first eight registers to automatically count signals from external sources. The START COUNTER instruction allows you to assign any one of the controller's inputs for this purpose.

NOTE: For proper operation, the pulse width must greater than one millisecond with a 50% duty cycle.

The debounce parameter within the start counter instruction is ignored in the 2600 / 2700 series controllers.

Example:

```
start ctr_1 up(in_10A)
```

Start counter No. 1 and assign input No.10.

```
if ctr_1 > 1200 goto close_box
```

When 1200 pulses come in from input No.10, goto a step labeled close_box.

Registers 9 - 125: General Purpose Registers

R/W Registers 9- 125 are volatile, i.e., data is lost when the controller is reset or powered down.

Example:

```
store reg_10 + 1 to reg_10
```

Increment the value in register No.10 by one.

```
if reg_10  > 100 goto process_complete
```

When the content of register No.10 exceeds the value of 100, goto a new step called process_complete.

Registers 129 - 130: General Purpose Registers

R/W Registers 133 -500 are volatile registers. See the description for registers 9 -125.

Registers 133 - 500: General Purpose registers

R/W Registers 133 -500 are volatile registers. See the description for registers 9 - 125.

Registers Nos. 133 - 500 are not available in 2200 and 2200XM controllers.

Registers 501 - 1000: Non-volatile General Purpose Registers

R/W Registers 501 - 1000 are non-volatile, i.e., they maintain their current data when the controller is powered down.**NOTE:** The R designation means the register

may be Read by the controller.
The W designation means the register may be Written to.

Register 126: Data Table Pointer

R/W Data Table pointer may also be used as general purpose register.

Register No. 126 is the Data Table row pointer. The value within this register defines the row number of the Data Table currently being referenced.

NOTE: There must be a Data Table present prior to using this function or software fault results.

Example:

```
store 5 to reg_126
Turn motor_1 to col_2
```

select row 5 of the Data Table
Turn stepper motor 1 to the position located in row 5, column 2 of the data table.

Registers 131 - 132: Data Table Row and Column Pointers

R/W Register 131 is the Data Table row pointer and is used with register No.9000. It may also be used as general purpose register.

R/W Register 132 is the Data Table column pointer and is used with register No.9000. It may also be used as general purpose register.

Register Nos. 131 and 132 and register No.9000 are used for writing data to the controller's Data Table from with your program.

NOTE: A Data Table must be present prior to using this function or software fault will result.

Example:

```
store 5 to reg_131
```

Select row No. 5 of the Data Table.

```
store 2 to reg_132
```

Select column No. 2 of the Data Table.

```
store 1234 to reg_9000
```

Write the number 1234 to row No. 5 column No. 2 of the Data Table.

NOTE: The R designation means the register may be Read by the controller.

The W designation means the register may be Written to.

Registers 127 - 128: Pointer for Phantom Register and Phantom Register

R/W The pointer for phantom register may also be used as general purpose register.

Phantom Register is used only in conjunction with register No.127.

Register Nos. 127 and 128 are used for indirectly accessing any of the controller's resources. Any value stored to register No. 127 actually references the register number of that value. Register No. 128 can then be accessed to read and write the content of the selected register.

Example:

```
store 501 to reg_127
```

Select register No. 501.

```
if reg_128 = 1234 goto process_complete
```

Same as: `if reg_501 = 1234`.

This technique is typically used for applications such as, data logging, data handling, data retrieving.

NOTE: The R designation means the register may be Read by the controller. The W designation means the register may be Written to.

Registers 1001 - 1016: Outputs 1 - 16

R/W Gives you alternative access to outputs 1 - 16.

Quickstep allows you reference all outputs as a register that is expressed as an instruction rather than a command. This gives you the ability to conditionally set or test each output in the body of a step. (0 = off, 1 = on)

Examples:

```
store 1 to reg_1002
```

Turn on output No. 2.

```
store 0 to reg_1002
```

Turn off output No. 2.

```
if reg_1015 = 1 goto next
```

If output No. 15 is on, goto next step.

Registers 2001 - 2016: Inputs 1 - 16

R only Gives youlalternative access to inputs 1 - 16.

All inputs may be referenced as a register instead of the ‘monitor’ instruction. (0 = open, 1 = closed)

Example:

```
if reg_2009 = 1 goto next
```

If input No.9 is on, goto next.

Register 9000: Access to the Data Table

R/W Gives you bidirectional access to Data Table.

Refer to the information for register Nos.131 and 132.

Group Access to I/O Points

NOTE: The R designation means the register may be Read by the controller. The W designation means the register may be Written to.

Registers 10101: Access to Outputs as a 16-bit Number

R/W Gives you access to outputs in groups of 16 as a 16-bit number.

This register allows 16-bit binary representation of outputs.

Example:

```
store 2048 to reg_10101
```

Turn on output 12 and turn 1 - 11 and 13 - 16 off.

Registers 10201 - 10202: Access to Outputs as a 8-bit Number

R/W Gives you access to outputs in groups of 8 as an 8-bit number.

These registers allow 8-bit binary representation of outputs, where register No.10201 is the first group of 8 outputs in the controller, register No.10202 is the second group, etc.

Example:

```
store 255 to reg_10201
```

Turn on outputs 1 - 8.

Registers 11101: Access to Inputs as a 16-bit Number

R only Gives you access to inputs in groups of 16 as a 16-bit number.

This register allows 16-bit binary representation of inputs.

Example:

```
if reg_11101 = 3 goto next
```

If input Nos. 1 and 2 are on and input Nos. 3 - 16 are off, continue to the next step.

Registers 11201 - 11202: Access to Inputs as a 8-bit Number

R only Gives you access to inputs in groups of 8 as an 8-bit number.

These registers allow 8-bit binary representation of outputs, where register No.11101 is the first group of 8 inputs in the controller, register No.11102 is the second group, etc.

Example:

```
if reg_11201 = 129 goto next
```

If input Nos.1 and 8 are on and input Nos. 2-7 are off, continue to the next step.

NOTE: The R designation means the register may be Read by the controller. The W designation means the register may be Written to.

Register 12000: Select Controller Communications Port – Write Access

See the next description for information on read access for register 12000.

Register 12000 specifies which communications port to use. The default is zero for COMM port one. To specify COMM port two, enter one in register 12000. Once the a port is selected, the same communications register may be used to read the transmission status.

In this instruction, storing 1 to register 12000 selects COMM port two.

```
store 1 to reg_12000
```

Register 12000: Message Transmission Status for Controllers – Read Access

R Gives you the message transmission status of a 2200XM, 2600XM, 2601 or 2700 controller.

If a message from the controller's Data Table is in transit (via RS-232), register No.12000 automatically contains a value of one. Once the message is complete, this register contains a zero. This allows your program to detect when a comm port is complete and ready to send another message.

- 0 = no message being transmitted
- 1 = transmitting in progress

Example:

```
if reg_12000 = 0 goto next
```

If the outbound message is complete, continue to the next step.

Register 12001: Transmit Message from Data Table

W Commences RS-232 message transmission from Data Table

Storing the row number of a message to be transmitted from the controller's Data Table to register No.12001 initiates transmission.

Example:

```
store 5 to reg_12001
```

Send the message on line five of the Data Table.

```
if reg_12000 = 0 goto next
```

When the message is complete, continue to the next step.

NOTE: A message will never be sent during the first 5 seconds of power application.

Registers 12001 - 12248: Receive Buffer Access for Controllers

R only Gives you access to RS-232 receive buffer for 2600, 2600XM, 2601, and 2700 controllers.

Register 12300: ASCII Protocol Variation

R/W Controls RS-232 ASCII protocol variation.

CTC's ASCII protocol defaults to terminal mode, responding to all commands with a <13><10> carriage return and line feed. Storing a 0 to register No.12300 in the beginning of your program causes it to return a <13> carriage return only.

- 0 = Computer (ASCII) protocol
- 1 = Terminal (ASCII) protocol

Register 12301: Baud Rate Selection

W only Sets baud rate of a selected channel.

NOTE: Not available in 2200/2200XM Controllers; communications is fixed at 9600 bps.

This register selects the baud rate of the controller's RS-232 port. Storing the assigned baud rate number causes the controller's communications port to configure after the first 5 seconds power application.

- 0 = 300 baud
- 1 = 600 baud
- 2 = 1200 baud
- 3 = 2400 baud
- 4 = 4800 baud
- 5 = 9600 baud
- 6 = 19200 baud
- 7 = 38400 baud

Register 12302: Character Count in Receive Buffer

R Counts the characters in receive buffer of selected channel

Register 12302 is typically used in conjunction with register Nos. 12001 (and up) to detect the count of characters received by the serial port.

Example:

If an external device is sending the controller 10 characters as a response to a message sent, then:

```
if reg_12302 >= 10 goto next
```

When ten characters are received from your external device, continue to the next step.

NOTE: If the incoming message contains a carriage return <cr> as its message terminator, the controller automatically sets register No.12302 to the value of 255 for 2600/2600XM controllers and 65 for 2200/2200XM controllers to indicate completetion.

Register 12302: Serial Buffer Message

W Discards the incoming message and resets the characters input buffer.

When you store any value to this register, the controller discards the incoming serial message and sets the input character buffer to the beginning.

Register12303: Disable Automatic Parsing

R/W Controls automatic controller parsing of selected channel.

Control Tech. controllers default to standard ASCII or binary protocols; it will respond to received commands found in the *Guide to Serial Communications*. If the external device connected to the controller does not conform to Control Tech.'s protocols, but you still need bring in characters to use within your program, we recommended that you disable the normal/automatic parsing of Control Tech. protocols. The controller, however, never allows parsing to be disabled during the first five seconds of power application.

Use the following values to inhibit or recieve standard protocols:

- 0 = inhibit controller response to incoming messages
- 1 = normal controller response to incoming messages

Register 12304: Extract Number from RS-232 Recieve Buffer

R only Extracts a number, expressed as a contiguous series of ASCII characters, from the receive buffer of selected channel.

NOTE: Not available in 2200/2200XM controllers.

Some external devices send strings of ASCII characters containing numeric information. This register automatically assemble ASCII strings into a numeric value. The result is a signed 32-bit number. The controller multiplies the number by 10,000, allowing decimal points to 4 places. Leading minus signs are also allowed.

Register 12305: Communications Priority When Running Multiple Tasks

R/W Gives you to grant a higher prioity to communications when running multiple tasks.

All Control Tech. controllers automatically receive characters regardless of what the controller is doing. This register gives you the ability to designate when the controller actually responds to a normal ASCII or binary protocol message, e.g., at the end all tasks being executed or after each step execution.

- The default is parsing messages after all tasks have been scanned.
- Storing 1 to register No.12305 causes the controller to parse an in coming message after each step execution.

Register 12306: Serial Port Switching

R/W Allows you to initiate and deactivate serial port switching.

At times you may want to perform normal 'Quickstep' communications while conversing to an external display.

Storing 1 to register 12306 causes the controller to switch a relay (the controllers transmit line) sending a message to a display and then switch back to communicate with the computer. The relay coil is connected to an output on the controller (see register No.12309) and automatically switches the relay when necessary (see Control Tech. Application notes for further wiring details).

Storing a 0 to register 12306 deactivates serial port switching.

Register 12307: Serial Port Switching Delay

R/W Allows you to specify a time delay for the relay used in serial port switching.

We recommend that you use a fast relay when implementing the relay switching feature. However, Register No.12307 allows you to program the switching time (in ms). The default is 2 ms.

```
store 10 to reg_12307
```

Delay for10 milliseconds after the relay has been switched but prior to sending any characters.

Register 12309: Specify Output for Serial Port Switching

R/W Specifies the output used to control the serial port switching relay.

You can assign the output the controller uses for the relay switching feature.

NOTE: Output No. 16 is fixed for this purpose on the 2200XM. Register 12309 doesn't exist in the 2200XM controller.

Example:

```
store 32 to reg_12309
```

Assign the relay switching feature to outputNo. 32.

Register 12310: Data Configuration for On-board Comm Port

R/W Specifies the data configuration for on-board comm-port data for 2200XM, 2600XM, 2601, and 2700 controllers.

To select the data configuration, store the following values in register 12310:

- 1 causes the on-board port to operate at 7-data bits, Odd parity.
- 0 causes the port to return to the default configuration of 8-data bits, no parity.

WARNING: Be careful, Quickstep does not operate in the 7-data bit, odd parity mode.

NOTE: The R designation means the register may be Read by the controller. The W designation means the register may be Written to.

Registers 5891 - 5894: Pulse Output Configuration For Output No. 1

R/W Register No. 5891 specifies the number of pulses to be sent out of output No. 1

- Maximum 32767 counts
- For continuous output of pulses set this register to -1.

R Register No. 5892 allows you to read the current pulse duration, in 100 microsecond intervals

R/W Register No. 5893 allows you to read or write the pulse on-time, in 100 microsecond intervals

- Maximum 65535 (or 6.5535 seconds)
- Minimum 100 microseconds

R/W Register No. 5894 allows you to read or write the pulse duration, in 100 microsecond intervals

- Maximum 65535 (or 6.5535 seconds)
- Minimum 200 microseconds

Registers 5895 - 5898: Pulse Output Configuration For Output No. 2

R/W Register No. 5895 specifies the number of pulses to be sent out of output No. 2.

- Maximum 32767 counts
- For continuous output of pulses set this register to -1

R Register No. 5896: Allows you to read the current pulse duration, in 100 microsecond intervals

R/W Register No. 5897 allows you to read or write the pulse on-time, in 100 microsecond intervals

- Maximum 65535 (or 6.5535 seconds)
- Minimum 100 microseconds

R/W Register No. 5898 allows you to read or write the pulse duration, in 100 microsecond intervals

- Maximum 65535 (or 6.5535 seconds)
- Minimum 200 microseconds

Example:
This step sends 10000 pulses out of output#1 at 500 pulses per second

```
store 10 to reg_5893
store 20 to reg_5894
store 10000 to reg_5891
if reg_5891 = 0 goto next
```

Set on-time to 1 ms.
Set duration to 2 ms.
Trigger 10000 counts to be sent out of output No. 1.
After 10000 counts have been sent, the program will continue.

Registers 5901 - 5904: Pulse Output Configuration For Output No. 1

R/W Register No. 5901 specifies the number of pulses to be sent out of output No. 1

- Maximum 32767 counts
- For continuous output of pulses set this register to -1.

R Register No. 5902 allows you to read the current pulse duration, in milliseconds

R/W Register No. 5903 allows you to read or write the pulse on-time, in milliseconds

- Maximum 65535 (or 65.535 seconds)
- Minimum 1 millisecond

R/W Register No. 5904 allows you to read or write the pulse duration, in milliseconds

- Maximum 65535 (or 65.535 seconds)
- Minimum 2 milliseconds

Registers 5905 - 5908: Pulse Output Configuration For Output No. 2

R/W Register No. 5905 specifies the number of pulses to be sent out of output No. 2.

- Maximum 32767 counts
- For continuous output of pulses set this register to -1

R Register No. 5906: Allows you to read the current pulse duration, in milliseconds

R/W Register No. 5907 allows you to read or write the pulse on-time, in milliseconds

- Maximum 65535 (or 65.535 seconds)
- Minimum 1 millisecond

R/W Register No. 5908 allows you to read or write the pulse duration, in milliseconds

- Maximum 65535 (or 65.535 seconds)
- Minimum 2 milliseconds

Example:
This step sends 10000 pulses out of output#1 at 500 pulses per second

```
store 1 to reg_5903
store 2 to reg_5904
store 10000 to reg_5901
if reg_5901 = 0 goto next
```

Set on-time to 1 ms.
Set duration to 2 ms.
Trigger 10000 counts to be sent out of output No. 1.
After 10000 counts have been sent, the program will continue.

Register 6500: Snapshot of Controller's Step Status

W Takes a snapshot of active step numbers.

Storing any value triggers a snapshot of the controller active steps numbers.

Example:

```
store 2 to reg_6500
```

Take a snapshot of active step numbers.

Register 6500: Number of Active Tasks

R Indicates quantity of active tasks at last snapshot.

Example:

```
store 1 to reg_6500
store reg_6500 to disp_2
```

Take a snapshot of the controllers active tasks.
Store the number of active tasks on display No. 2.

Registers 6501-6532: Step Number of Active Tasks

R only Lists the step numbers of active tasks

Registers 6501-6532 list the step numbers of the active steps at last snapshot. The numbers are offset by -1.

Register 6599: Step number scanned

R only Lists the step number of the task being scanned at last snapshot.

Register 13002: Continuous millisecond counter

```
R/W     Automatically increment the continuous millisecond
counter.
```

Register 13002 automatically increments by the value one after every millisecond regardless of program operation. It is typically used to time functions on a machine or to keep track of system up-time, etc. You can set this register to a new value or reset to zero using the store instruction.

This register continuously counts up to the maximum 32-bit value (2,147,483,647) and then rolls-over to a negative number.

Example:

```
store 0 to reg_13002
```

Clear the millisecond counter.

```
if reg_13002 > 2000 goto next
```

When two seconds have past, continue to the next step.

Register 13003: Revision level of Firmware

R only Lists the revision level of the controller firmware (mult. x 10)

Register 13004: Controller Off

R only Indicates Control Technology's "expanded architecture" in 2600XM, 2601, and 2700 controllers.

Returns a 1, indicating that the controller has Control Technology's "expanded architecture."

Register 13005: Flag Access as 32-bit Number

R/W Gives you access to Flags #1 - #32 as a 32-bit number.

Example:

```
store 0 to reg_13005
```

Clear all flags in the controller.

```
if reg_13005 = 549 goto next
```

If flags 1, 3, 6, 10 are set and all others are clear go to the next step.

Register 13008: Controller Model Code

R/W Contains a code indicating the controller model number:

In the 2600/2600XM and 2601 controllers, register 13008 must be set to a 3 in order to use the utilities programs. You can do this with a store or monitor instruction.

Register 13009: Automatically Turn Off Output

R/W Allows you turn off an output automatically in case of a Software fault.

Storing the number of an output to register 13009 causes the controller to automatically turn off that output in the event of a software fault. Storing a zero to the register, turns off this feature.

Example:

```
store 10 to reg_13009
```

Turn output No. 10 off in the event of a software fault.

Register 13011: Task Priortiy

R/W Establishes a selected task to be executed on a high priority basis.

After the controller starts executing the program tasks, storing a specific task number (for high priority servicing) to register No. 13011 once, causes the controller to execute that task after each other task is serviced. If you do not know the task number, storing 1000 to register No.13011 causes the task that executed the store instruction to have high priority servicing.

Available in model 2600, 2601 and 2700 controllers, Rev 1.6 or greater.

Register 13012: Current Task Number

R only Indicates the number of the current task.

Reading the value in register No.13012 gives you the current task being executed for external viewing and internal “what task am I?”.

Available in model 2600/2600XM, 2610 and 2700 controllers, Rev 1.6 or greater.