

5100 Modbus TCP Slave & RTU Serial Server Configuration

Blank Page

Table of Contents

Modbus	5
Modbus TCP Slave	5
Modbus Serial RTU	. 11
Test Summary	. 13
Assigning IP Address, Subnet Mask, and Gateway Address	. 13

Blank Page

Modbus

The Modbus Protocol is a messaging structure developed by Modicon in 1979. It is used for master-slave/client-server communication between intelligent devices and has become an industry standard. Details of the protocol may be found at the web site <u>www.modbus.org</u> for further details. There are numerous deviations of the Modbus protocol of which the 5100 supports those described within this document. Tools used to test the protocol are available from a number of sources. The 5100 was tested using those available from <u>www.win-tech.com</u>, namely their ModScan32 for RTU Slave testing and ModSim32 for Master.

This document discusses the configuration and testing when using Modbus TCP Slave (server) to interact with the 5100, while at the same time connecting to COM1 with a serial cable running the Modbus RTU Serial protocol. Note on both connections the 5100 operates as a slave (server), returning information as requested by the polling master.

Modbus TCP Slave

The Modbus TCP Slave protocol allows a TCP master to periodically poll the 5100 to collect desired information. The protocol allows for interfaces to such things as coils, analog, register, etc. Since the 5100 is able to access anything via its register interface, only the Holding Register commands are supported; Write Single Register (function code 0x06), Write Multiple Registers (function code 0x10), and Read Holding Registers (0x03).

				Funct	ion Codes	<u>^</u>
				code	Sub code	(hex)
	1	Physical Discrete Inputs	Read Input Discrete	02		02
Data Access		Internal Bits	Read Colls	01	0 3	01
	Bit access	Or	Write Single Coll	05		05
		Physical colls	Write Multiple Colls	15		OF
		Physical Input Registers	Read Input Register	04		04
		Internal Registers Or Physical Output Registers	Read Multiple Registers	03		03
	16 bits		Write Single Register	06		06
	access		Write Multiple Registers	16		10
			Read/Write Multiple Registers	23		17
			Mask Write Register	22		16
	File record access		Read File record	20	6	14
			Write File record	21	6	15
	Encapsul	ated Interface	Read Device Identification	43	14	28

Modbus Function codes from Modbus.org, Modbus Application Protocol Specification, May 8, 2002

You should also note that Modbus registers are 16 bits in width and that of the 5100 are 32 bits, since Modbus is Big Endean, this means reading register 1 in the 5100, the high 16 bits equates to Modbus register 1 and the low 16 bits to Modbus register 2. Modbus register 3 would be the high 16 bits of the 5100 register #2. A maximum of 50 Modbus registers can be read at once, or 25 5100 sequential registers.

As a demonstration of the functionality of the 5100 Modbus TCP/Slave interface this section details the interface of Win-Tech's ModScan32 software and how it applies with regard to our product. As mentioned before, we only support the Holding Register interface. Upon installation of ModScan32 a screen such as Figure 7.3 will appear. Note that the 'Address' field is set to 1, but the display screen starts at 40001. This is Modbus nomenclature. 'Address' of 1 is the same as the upper 16 bits of the 5100 register 1. Note 'Length' is set to 50, the maximum allowable number of Modbus registers in a single read, Device ID is ignored since TCP is point to point.



Figure 1.2: ModScan32 Master Scanning Program (only Holding Register supported)

Figure 9.3 shows the setup for an interface to a 5100 with a TCP address of 12.40.53.199 and the Modbus Slave running a server on the standard port of 502:

ni Ni	dScan32 - [ModScan e Connection Setup	1] i Wawi Windowi Ha	þ		_ # ×
		1	N?		
Adc	Connection Details		_	×	
Ler	Re	nole TCP/IP Server		* rs	
		IP Addens	12.40.53.199		
4001	- Configuration	Service Port:	502	Modbus Protocol Selections	x
4001 4001 4001 4001	Baud Role:	3200 💌	Hardware Elow Control	Tianonisoion Mode STANDARD	DANIEL/ENRON/OWN
4001 4001 400:	Word Length: 8	E I I	Bulay 0 monather B' barrondtog	C ASOI (• RTU	C ASCI C RTU
400: 400: 400:	Stop Bits:	¥	Delay D insister lo before rele		000 (neect)
400: 400: 400: 400: 400:		Pro DK	stacal Selections	Delay Between Pr	sta 50 (noecs)
4002 4002 4002 4002	2: (DDDDD) 4D 3: (DDDDD) 4D 4: (DDDDD) 4D	048: <00000> 049: <00000> 050: <00000>		Force modbus comm [Tip be used in cases single point write fund	and 15 and 16 for single-point writes. where the slave does not support the fore 05 and 05.)
4002	6: <00000> 6: <00000>			0	K Cancel

Figure 1.3: ModScan32 Master Scanning Program TCP Connection Setup

In order to do a single register write to a Modbus 16 bit register double click that register. Below shows changing Modbus register 40002 (Address 2) to a value of 5, this would translate to the lower 16 bits of Quickstep register 1. Remember Modbus Address 1 is the upper 16 bits.

10001	Write Register 🛛 🗶
40001: (00000) 40027: (00000)	
40002: <00000> 40028: <00000>	
40003: <000kg0> 40029: <00000>	Node:
40004: <00000> 40030: <00000>	
40005: <00000> 40031: <00000>	
40006: <00000> 40032: <00000>	Address: 2
40007: <00000> 40033: <00000>	
40008: <00000> 40034: <00000>	
40009: <00000> 40035: <00000>	Value: 5
40010: <00000> 40036: <00000>	
40011: <00000> 40037: <00000>	
40012: <00000> 40038: <00000>	Under Count 1
40013: <00000> 40039: <00000>	Update Cancel
40014 < 40000 > 40040 < 40000 >	
40015 (00000) 40041 (00000)	
40016 (00000) 40042 (00000)	
40012 (00000) 40042 (00000)	
40018 (00000) 40044 (00000)	
40019: (00000) 40044. (00000)	
40020: (00000) 40045: (00000)	
40021 (00000) 40048. (00000)	
40021: (00000) 40047: (00000)	
40022: (00000) 40048: (00000)	
40023: (00000) 40049: (00000)	
40024: (00000) 40050: (00000)	
40025: <00000>	
40026: <00000>	

Figure 1.4: Single register write, value 5 to 40002

Changing a number of register all at once is known as a Write Multiple Register access. This can be done using the Extended Access option:

Copyright 2002 © Control Technology Corporation All Rights Reserved. October 3, 2002 Revision A.

ModScan32 - [ModSca1]								
💼 File - C	onnection	Setup	View	Windo	w	Help		
	a 💻	Data Disp	a Defini lay Opt	tion ions 🕨		१ №		
<u> </u>	13 🐼 🛙	Exte	ended	•		Force Coils		
	.· 0001	Text Dba:	: Captu se Capl	re :ure		Preset Regs Mask Write		N
Length:	50	Capl	ture Of	f	-	User Msg Run Script	-	۷;
		Rese	et Ctrs		J			

Figure 1.5: Write Multiple register (Preset Regs) selection

The Preset Multiple Registers will appear. Note that in TCP the 5100 ignores any slave or node identifiers since it is a single device and not acting as a gateway. Set the Modbus register you wish to start changes with and the number of registers to change, up to a maximum of that you are viewing:

16: PRESET MULTIPLE	REGISTERS 🔀
Slave Device:	1
Address:	1
Number of Points:	50
OK	Cancel

Figure 1.6: Preset Multiple register dialog

In this case we will change Addresses 1 to 10 to sequential numbers 1 to 10:

16: PRESET MULTIPLE	REGISTERS 🔀
Slave Device:	1
Address:	1
Number of Points:	10
OK	Cancel

Figure 1.7: Select number of multiple writes to do

Address: 0001 Length: 50	Device MOD 16 03: HOLD	rt- 1 PRESET MULTIPLE REGISTERS Address: 0001 Lengfi: 0010	×	
40001: (00000) 40002: (00005) 40003: (00000) 40005: (00000) 40005: (00000) 40005: (00000) 40007: (00000) 40009: (00000) 40010: (00000) 40011: (00000) 40012: (00000) 40015: (00000) 40016: (00000) 40016: (00000)	$\begin{array}{ccccc} 40027:&<0000\\ 40028:&<0000\\ 40029:&<0000\\ 40030:&<0000\\ 40031:&<0000\\ 40032:&<0000\\ 40033:&<0000\\ 40035:&<0000\\ 40036:&<0000\\ 40036:&<0000\\ 40038:&<0000\\ 40038:&<0000\\ 40039:&<0000\\ 400403:&<0000\\ 40040:&<0000\\ 40041:&<0000\\ 40041:&<0000\\ 40044:&<0000\\ 40044:&<0000\\ 40044:&<0000\\ 40044:&<0000\\ 40044:&<0000\\ 40044:&<0000\\ 40044:&<0000\\ 40044:&<0000\\ 40044:&<0000\\ 40044:&<0000\\ 40044:&<0000\\ 40044:&<0000\\ 40044:&<0000\\ 40044:&<0000\\ 40044:&<0000\\ 40044:&<0000\\ 40044:&<0000\\ 40044:&<0000\\ 40044:&<0000\\ 40044:&<0000\\ 40044:&<0000\\ 40044:&<0000\\ 40044:&<0000\\ 40044:&<0000\\ 40044:&<0000\\ 40044:&<0000\\ 40044:&<0000\\ 40044:&<0000\\ 40044:&<0000\\ 40044:&<0000\\ 40044:&<0000\\ 40044:&<0000\\ 40044:&<0000\\ 40000\\ 40044:&<0000\\ 40000\\ 40044:&<0000\\ 40000\\ 40044:&<0000\\ 4000\\ 400\\ 4000\\ 4000\\ 400$	Update U010	From File To File	
40020: (00000) 40021: (00000) 40022: (00000) 40022: (00000) 40024: (00000) 40025: (00000) 40025: (00000)	40045: <0000 40046: <00000 40047: <00000 40048: <00000 40049: <00000 40050: <00000			

As shown below the current register values are displayed in the dialog box.

Figure 1.8: Preset Multiple register dialog viewing existing values

Note below that each register value has been changed, also we scrolled down so we could get to register 10. Click Update and note the changed register values from the previous display, 40002 is no longer 5 but now 2.

16: PRESET MULTIP	LE REGISTER	15	×
4	ddress: 0001		
	Length: 0010		
0003:	00003	-	From File
0004:	00004		To File
0005:	00005		
0006:	00006		
0007:	00007		
0008:	80000		
0009:	00009		
0010:	00010	-	
Updal	× C	ncel	
	-0		

Figure 1.9: Preset Multiple new values entered

Upon clicking the Update key, the new values are written to the 5100 registers and new values read back using the Read Multiple Register command.

ModScan32 - [ModSca	1]		- @ X
Dicking elesis			181 X
Address: 0001 Length: 50	Device ld: 1 MODBUS Point Type 03: HOLDING REGISTER •	Number of Polls: 1216 Valid Slave Responses: 1216 Reset Ctrs	
40001: (00003) 40 40002 (00002) 40 40003 (00005) 40 40006 (00005) 40 40006 (00005) 40 40006 (00005) 40 40007 (00007) 40 40007 (00007) 40 40007 (00007) 40 40001 (00005) 40 40011 (00005) 40 40012 (00005) 40 40015 (00005) 40 40015 (00005) 40 40015 (00005) 40 40016 (00005) 40 40017 (00005) 40 40017 (00005) 40 40017 (00005) 40 40018 (00005) 40 40019 (00005) 40 40021 (00005) 40 40021 (00005) 40 40022 (00005) 40 40022 (00005) 40 40022 (00005) 40 40023 (00005) 40 40023 (00005) 40	0027: 000000 1028: 000000 1029: 000000 1031: 000000 1032: 000000 1031: 000000 1032: 000000 1033: 000000 1034: 000000 1035: 000000 1037: 000000 1037: 000000 1037: 000000 1037: 000000 1038: 000000 1041: 000000 1044: 000000 1044: 000000 1045: 000000 1044: 000000 1044: 000000 1044: 000000 1044: 000000 1044: 000000 1044: 000000 1044: 000000 1044: 000000 1044: 000000 1044: 000000 1044: 000000 1044: 000000		

Figure 1.10: New values written and read back, Quickstep registers 1 to 5, Modbus 1 to 10

Should any errors occur a Modbus exception will occur. One such common error is attempting to read too many registers or illegal registers. Below is what is returned if > 50 Modbus registers are attempted:

Image: Second and the second and	- MedScan3	2-[Mod	Scal]	Marshare 14	ulo.				- # ×
Image: Second	Distal	6161			11/2				-140 Z
Address: Dovice Id: 1 MODBUS Point Type Number of Polls; 1459 Valid Slave Responses: 1440 Length: 75 03: HOLDING REGISTER • Reset Ces *** 00001 40027 400001 40053 • *** 00001 40027 600000 40053 • • *** 00001 40027 600000 40055 • • *** 00001 40027 000000 40055 • • *** 00001 40029 000000 40055 • • 40001 00004 40057 000000 • • • 40005 000053 40029 000000 40057 000000 • 40005 000054 40030 00057 000000 • • 40006 000054 00032 • • • • 40005 000054 00059 • • • • 40010					10-1				
Address: D001 MODBUS Point Type Number of Polls; 1459 Length: 75 03: HOLDING REGISTER • Reset Cere ve. 00007 5.500mpt Joon 700000 40053 c00000 40001 c000012 40027 c000002 40055 c000002 40003 c00003 40027 c000002 40055 c000002 40004 c00003 40027 c000002 40055 c000002 40005 c00003 40027 c000002 40055 c000002 40005 c00005 40055 c000002 40055 c000002 40005 c00005 40057 c000002 40058 c000002 40005 c00005 40058 c000002 40058 c000002 40007 c00005 40052 c000002 40058 c000002 40008 c00005 40052 c00000 40056 c000002 40010 40012 c00000 40065 c00000	ET fet mi	<u> </u>	4 21 20	600					
Length: 76 03: HOLDING REGISTER • 40001 000012 40027 000002 40053 000002 40001 000012 40027 000002 40053 000002 40001 000012 40027 000002 40053 000002 40001 000012 40029 000000 40055 000000 40005 000005 40055 000000 40056 000002 40005 00005 40032 000000 40057 000000 40005 00005 40032 000000 40057 000000 40005 00005 40033 000000 40058 000000 40006 00005 40033 000000 40059 000000 40007 000070 40052 000000 40051 000000 40010 000010 40033 000000 40063 000000 40012 000000 40063 000000 40064 000000 <tr< th=""><th>Address:</th><th>0001</th><th></th><th>Device la MODBUS</th><th>f: 1 S Point T</th><th>ype</th><th>Number of Pol Valid Slave Re</th><th>lis: 1459 esponses: 1440</th><th></th></tr<>	Address:	0001		Device la MODBUS	f: 1 S Point T	ype	Number of Pol Valid Slave Re	lis: 1459 esponses: 1440	
# (DDBUS) Excerpt tool Respective Status Device *** 40001 (00001) 40027 (00000) 40053 (00000) 40002 (00002) 40028 (00000) 40054 (00000) 40003 (00002) 40028 (00000) 40055 (00000) 40004 (00004) 40023 (00000) 40055 (00000) 40005 (00005) 40033 (00000) 40055 (00000) 40006 (00005) 40033 (00000) 40055 (00000) 40007 (00007) 40033 (00000) 40059 (00000) 40008 (00005) 40034 (00000) 40059 (00000) 40010 (00010) 40034 (00000) 40059 (00000) 40011 (00010) 40034 (00000) 400514 (00000) 40012 (00000) 40054 (00000) 40045 (00000) 40012 (00000) 40064 <th>Length:</th> <th>76</th> <th>03</th> <th>HOLDING</th> <th>REGIST</th> <th>ER •</th> <th>ia.</th> <th>Reset Cits</th> <th></th>	Length:	76	03	HOLDING	REGIST	ER •	ia.	Reset Cits	
40007 (00002) 40020 (00000) 40054 (00000) 40003 (00003) 40029 (00000) 40055 (00000) 40004 (00005) 40031 (00000) 40057 (00000) 40004 (00005) 40031 (00000) 40057 (00000) 40004 (00005) 40032 (00000) 40058 (00000) 40004 (00005) 40033 (00000) 40059 (00000) 40004 (00005) 40033 (00000) 40059 (00000) 40005 (00005) 40034 (00000) 40052 (00000) 40010 (00010) 40036 (00000) 40062 (00000) 40010 (00010) 40037 (00000) 40063 (00000) 40010 (00010) 40037 (00000) 40063 (00000) 40012 (00000) 40063 (00000) 40064 (00000) 40014 (00000) 40064 (00000) 40045 (00000) 40014 (00000) 40064 <td>ADDD1 CD</td> <td>Except</td> <td>ADD27</td> <td>CODDDS</td> <td>dn Slat</td> <td>· Device</td> <td></td> <td></td> <td></td>	ADDD1 CD	Except	ADD27	CODDDS	dn Slat	· Device			
40003 400043 400043 400044 400044 400044 400044 400044 400044 400044 400044 400044 400044 400044 400044 400044 400044 400044 400044 400044 400044 40044	40002 <0	0007>	40028	<00000>	40054	<000000	6		
40005 400004 40005 40005 400005 40005 400005 40005 40005 400005 40005 40007 40007 40007 40007 40007 40006 400085 40023 400005 40069 400005 40007 400085 40024 400005 40069 400605 40008 400085 40024 400005 40061 400005 40011 400085 400005 40061 400005 40061 40011 400085 400005 40064 400005 40011 40011 400005 40065 400005 40061 400005 40011 400005 40064 400005 40011 400005 40015 400005 40067 400005 40067 400005 40016 400005 40067 400005 40011 400005 40017 400005 40067 400005 40011 4000005	40003 <0	0003>	40029	<00000>	40055	<00000)	8		
40006 (00006) 40032 (00000) 40058 (00000) 40008 (00008) 40033 (00000) 40059 (00000) 40008 (00008) 40033 (00000) 40061 (00000) 40009 (00008) 40034 (00000) 40061 (00000) 40010 (00010) 40056 (00000) 40062 (00000) 40011 (00000) 40036 (00000) 40064 (00000) 40012 (00000) 40056 (00000) 40066 (00000) 40013 (00000) 40066 (00000) 40067 (00000) 40014 (00000) 40067 (00000) 40068 (00000) 40015 (00000) 40068 (00000) 40068 (00000) 40015 (00000) 40068 (00000) 40017 (00000) 40017 (00000) 40068 (00000) 40017 (00000) 40019 (00000) 40071 (00000) 40071 (00000) 40019 (00000) 40072 <td>40005 (0</td> <td>0004</td> <td>40030</td> <td>< 000000 »</td> <td>40055</td> <td>CODDDD</td> <td></td> <td></td> <td></td>	40005 (0	0004	40030	< 000000 »	40055	CODDDD			
40007 <00007> 40033 <00000> 40059 <00000> 40008 <00008> 40034 <00000> 40060 <00000> 40009 <00009	40006 (0	0006	40032	<00000>	4005B	(00000)	63 - C		
40008 <00008> 40024 <00000> 40060 <00000> 40009 <000010> 40034 <00000> 40062 <00000> 40010 <00000> 40037 <00000> 40063 <00000> 40011 <000000> 40037 <00000> 40063 <00000> 40012 <00000> 40043 <00000> 40064 <00000> 40013 <00000> 40040 <00000> 40066 <00000> 40014 <00000> 40066 <00000> 40045 <00000> 40014 <00000> 40068 <00000> 40047 <00000> 40017 <00000> 40069 <00000> 40047 <00000> 40018 <00000> 40070 <00000> 40047 <00000> 40018 <00000> 40071 <00000> 40047 <00000> 40020 <00000> 40073 <00000> 40047 <00000> 40021 <00000> 40074 <00000> 40073 <00000> 40021 <000000> 40074	40007 <0	0007>	40033	<00000>	40059	<000000>	63		
40010 (00010) 40038 (00000) 40061 (00010) 40023 (00000) 40062 (00000) 40063 (00000) 40063 (00000) 40063 (00000) 40063 (00000) 40063 (00000) 40063 (00000) 40063 (00000) 40063 (00000) 40064 (00000) 40064 (00000) 40064 (00000) 40064 (00000) 40064 (00000) 40064 (00000) 40064 (00000) 40064 (00000) 40064 (00000) 40064 (00000) 40064 (00000) 40064 (00000) 40064 (00000) 40064 (00000) 40064 (00000) 40064 (00000) 40064 (00000) 40064 (00000) 40064 400000) 40064 400000) 40064 400000) 40067 400000) 40067 400000) 40067 400000) 40067 400000) 40067 400000) 40067 400000) 40067 400000) 40067 4000000) 40067<	4000B <0	DDDB>	40034	<00000>	40060	< ODDDD >	8		
40011 (00000) 40035 (00000) 40052 (00000) 40012 (00000) 40037 (00000) 40064 (00000) 40013 (00000) 40039 (00000) 40065 (00000) 40013 (00000) 40044 (00000) 40065 (00000) 40014 (00000) 40064 (00000) 40064 (00000) 40015 (00000) 40041 (00000) 40068 (00000) 40016 (00000) 40042 (00000) 40068 (00000) 40017 (00000) 40064 (00000) 40070 (00000) 40019 (00000) 40070 (00000) 40071 (00000) 40019 (00000) 40072 (00000) 40073 (00000) 40021 (00000) 40073 (00000) 40073 (00000) 40022 (00000) 40075 (00000) 40075 (00000)	40009 <0	DDD9>	40036	(DDDDD)	40063	(DDDDD)	8		
40012 (00000) 40038 (00000) 40044 (00000) 40013 (00000) 40039 (00000) 40045 (00000) 40014 (00000) 40044 (00000) 40045 (00000) 40015 (00000) 40041 (00000) 40067 (00000) 40016 (00000) 40042 (00000) 40066 (00000) 40017 (00000) 40043 (00000) 40070 (00000) 40018 (00000) 40045 (00000) 40070 (00000) 40019 (00000) 40043 (00000) 40070 (00000) 40019 (00000) 40070 (00000) 40070 (00000) 40019 (00000) 40071 (00000) 40071 (00000) 40019 (00000) 40072 (00000) 40073 (00000) 40020 (00000) 40073 (00000) 40074 (00000) 40022 (00000) 40075	40010 (0	ODDD>	40035	(00000)	40062	000000	83 - C		
40013 400000 40010 40045 400000 40014 400000 40040 400000 40066 400000 40015 600000 40041 600000 40067 600000 40015 600000 40042 600000 40067 600000 40016 600000 40043 600000 40069 600000 40017 600000 40044 600000 40070 600000 40018 600000 40070 600000 40070 600000 40012 600000 40072 600000 40073 600000 40021 600000 40073 600000 40073 600000 40023 600000 40075 600000 40075 600000	40012 (0	DDDD	40038	(DDDDD)	40064	(DDDDD)	<u> (</u>		
40014 <00000> 40040 <00000> 40066 <00000> 40015 <00000> 40041 <00000> 40067 <00000> 40017 <00000> 40043 <00000> 40069 <00000> 40017 <00000> 40043 <00000> 40069 <00000> 40018 <00000> 40043 <00000> 40070 <00000> 40019 <00000> 40044 <00000> 40072 <00000> 40020 <00000> 40047 <00000> 40073 <00000> 40022 <00000> 40049 <00000> 40075 <00000> 40023 <00000> 40075 <00000> 40075 <00000>	40010 <0	DDDD	40039	< DDDDD>	40065	< DDDDD	8		
40015 (00000) 40041 (00000) 40067 (00000) 40016 (00000) 40042 (00000) 40068 (00000) 40017 (00000) 40043 (00000) 40069 (00000) 40018 (00000) 40044 (00000) 40070 (00000) 40019 (00000) 40044 (00000) 40071 (00000) 40020 (00000) 40046 (00000) 40072 (00000) 40022 (00000) 40047 (00000) 40073 (00000) 40022 (00000) 40049 (00000) 40073 (00000) 40022 (00000) 40049 (00000) 40075 (00000)	40014 <0	(GDDD	40040	<000000>	40066	< DDDDD)	63		
40016 (00000) 40042 (00000) 40068 (00000) 40017 (00000) 40043 (00000) 40069 (00000) 40018 (00000) 40044 (00000) 40070 (00000) 40019 (000000) 40045 (00000) 40071 (00000) 40020 (00000) 40047 (00000) 40073 (00000) 40022 (00000) 40048 (00000) 40073 (00000) 40022 (00000) 40048 (00000) 40074 (00000) 40023 (00000) 40049 (00000) 40075 (00000)	40015 <0	<0000	40041	< 00000 >	40067	< 000003	8		
40017 (00000) 40043 (00000) 40059 (00000) 40018 (00000) 40044 (00000) 40070 (00000) 40020 (00000) 40045 (00000) 40072 (00000) 40021 (00000) 40047 (00000) 40073 (00000) 40022 (00000) 40048 (00000) 40073 (00000) 40022 (00000) 40049 (00000) 40075 (00000)	40016 <0	DDDD>	40042	< DDDDD>	40068	< DDDDD)	6		
40019 (D0000) 40045 (D0000) 40071 (D0000) 40020 (00000) 40045 (00000) 40072 (00000) 40021 (00000) 40047 (00000) 40073 (00000) 40022 (00000) 40048 (00000) 40073 (00000) 40022 (00000) 40049 (00000) 40075 (00000)	40017 (0	DDDD>	40043	CODDDDS	40059	CODDDDS			
40020 (00000) 40046 (00000) 40072 (00000) 40021 (00000) 40047 (00000) 40073 (00000) 40022 (00000) 40048 (00000) 40073 (00000) 40023 (00000) 40049 (00000) 40075 (00000)	48819 (0	88882	40044	(000000)	48871	(000000)	83 - C		
40021 (00000) 40047 (00000) 40073 (00000) 40022 (00000) 40048 (00000) 40074 (00000) 40023 (00000) 40049 (00000) 40075 (00000)	40020 <0	DDDD	40046	(DDDDD)	40072	(00000)			
40022 <00000> 40040 <00000> 40074 <00000> 40023 <00000> 40049 <00000> 40075 <00000>	40021 (0	<0000	40047	<00000>	40073	(00000)	65		
40023 (00000) 40049 (00000) 40075 (00000)	40022 <0	<00000	40048	<000005	40074	<000003	60)		
	40023 <0	CODD>	40049	<00000>	40075	<0000003	1.2		
40024 (00000) 40050 (00000)	40024 (0	00000	40050	<000000×					
40025 600000 40051 600003 40026 200000 40052 200000	40026 (0	00000	40051	(000000)					

Figure 1.11: Modbus Exception Example > 50 registers

Copyright 2002 © Control Technology Corporation All Rights Reserved. October 3, 2002 Revision A. Editing the 75 appropriately will update the error. Below is an example of displaying registers in the 13002 block of the controller. 13002 is the system tic counter, real time clock/date values can also be seen incrementing in other register dynamically. Note that 26003 is the high 16 bits of 13002 and 26004 (13002 X 2) is the base lower 16 bits.

	_				
ModSca	m32 - [Mod5	ca1]			_ # ×
💼 File Co	onnection Set	up View Winds	ом Неір		_ @ X
Dietin	lister in	w les el la	asi o i voi		
		100 100 100			
	20002	- Dev	rice ld: 1	Number of Dolle: 1658	1
Address	20003	M01	DBUS Point Type	Valid Slave Responses: 1545	
Length:	50	03: HOL			1
congar.	00		Ding negisten	Reset Ctrs	
426003:	<00318>	426029: <0	0000>		
426004:	<05462>	426030: <0	0029>		
426005:	<00000>	426031: <0	0000>		
426006:	<40201>	426032: <0	0010>		
426007:	<000003	426033: 00	000035		
426009:	<00000>	426035: <0	0000>		
426010:	<00000>	426036: <0	0001>		
426011:	<00000>	426037: <0	0000>		
426012:	<000005	426038: <0	0000>		
426013:	(55282)	426039: 00	00003		
426015:	<00000>	426041: <0	0000>		
426016:	<00015>	426042: <0	0000>		
426017:	<00000>	426043: <0	0000>		
426018:	<000000>	426044: (0	0000>		
426020:	<000000>	426046: <0	0000>		
426021:	<00000>	426047: <0	0000>		
426022:	<00000>	426048: <0	0000>		
426023:	<000000>	426049: <0	(0000)		
426024:	<000005	426050: (0	00005		
426026:	<00001>	426052: <0	0000>		
426027:	<00000>				
176020.	<000435				

Figure 1.12: Display of 5100 system tic, dynamically updating

Modbus Serial RTU

The Modbus Serial RTU protocol functions exactly like that of Modbus TCP with regards to how to access information and ModScan32 operation (see figure 1.13 for serial port setup versus TCP). There are some key differences since an RS232 connection is used versus a network connection. They are as follows:

- 1. Only COM1 can be used for the Modbus Serial RTU protocol. COM2 uses an intelligent controller chip which does not currently support the protocol. COM2 support may be added in the future.
- 2. The virtual TCP communication ports (when interacting with a terminal server) may also be used but only for point to point operations with a single address

Copyright 2002 © Control Technology Corporation All Rights Reserved. October 3, 2002 Revision A. present. In other words the communications traffic of other Modbus nodes should not be present (can be on COM1). This is necessary because Modbus specifies a 3.5 character quiet time between packets and a maximum of 1.5 intercharacter delay during the continuous transmission of a packet data stream. The virtual ports can not guarantee these timing constraints, although from a high level protocol viewpoint, the ports do comply.

- 3. By default the Modbus protocol is disabled on the serial and virtual ports at power up. To enable the port it must be the active port in the 12000 register and the Modbus Slave address value must be written to register 12320. Note that by default the slave port address is 2 and that any value written as the Modbus slave address will be that used on all serial ports, system wide. Note that writing a value of 0 to 12320 will disable Modbus on that port only and not effect the system wide address.
- 4. When Modbus is enabled on a serial port using CTCMON no further communications will be available on that port except with Modbus. In other words you will loose your CTCMON link if talking on the same port.

1110.41	
dress: 0001 Device ldt 2 Number of Polit MODBUS Point Type Valid Slave Re	is: 0 spanses: 0
ngth: 50 03: HOLDING REGISTER 🔹	Reset Cles
storChet Behale	20
Eurosed Uring	
Direct Canvection to CDH1	
24pp;51400- 1(2.4153.20	
Security (Ma	Madhus Profacal Selectama
Hardware Flow Castul	- Transmission Mode
Bead Plate 19200 + Wait to DSR ton slave	STANDARD DANEL/ENRON/CHIN
Ward Length 10 Delay 10 ns. atter RTS before	CASOL FINTU CASOL CINTU
Party PIDNE	- Size Response Travout
Skp Bits 1 * Delay 3 mr aller last character Safara telearen BTS	500 (resca)
Protocol Subartisms	20 Internet
or I card I	
	Foce modeus command 15 and 16 for angle-point writes.
	(To be used in cares where the stave does not support the range-point write functions 05 and 06.1
	DK Castal

Figure 1.13: ModScan32 Master Scanning Program Serial Connection Setup

Since at power up Modbus is disabled on the COM1 serial port, it must be enabled by writing the desired Modbus address the 5100 is to respond to. This is written to register 12320. Typically done via a Quickstep program, for test purposes it can be also be done

Copyright 2002 © Control Technology Corporation All Rights Reserved. October 3, 2002 Revision A.

with CTCMON or in the test case presented here, via the Modbus TCP Connection. When done with the TCP connection simply do a write operation to Modbus address 24640 (12320 X 2) of the desire serial RTU address and COM1 will immediately respond to Modbus requests.

Performante mentre in anna en a	
Fighter Strat	
Address: 24640 Device M: 1 MODBUS Point Type	Number of Polls: 2204 Valid Slave Responses: 2204
Length: 30 03: HOLDING REGISTER	Renet Cirs
	write fregister
474640 (00000) 424456 (00000)	Hade 1
434641 (00000) 424657 (00000) 434642 (00000) 424658 (00000)	The second se
424643 (00000) 424659 (00000) 424644 (00000) 424660 (00000)	Addens (24640
474645 (00000) 474661 (00000) 474646 (00000) 424662 (00000)	Vale D
424647 (00000) 424663 (00000) 474648 (00000) 424664 (00000)	
434649: (80080) 424665: (00800) 424650: (80080) 424666: (00800)	Updes Cancel
424651 <000000 424667 <000000 474652 <000000 474668 <000000	
424653 (00000) 474669 (00000) 424654 (00000)	
424655 : <80080>	

Figure 1.14: ModScan32 Master TCP changing RTU serial address to 3

Above shows device address 3 being written to Modbus register 24620, thereby setting the address for COM1 to respond to. Device ID #1 is connected via TCP.

Test Summary

- 1. Visit <u>www.win-tech.com</u> and download/install ModScan32 per their instructions.
- 2. Install CTCMON and set the appropriate IP address, subnet mask, and gateway (if needed) within the 5100 via the serial port. Details are below for the proper register settings:

Assigning IP Address, Subnet Mask, and Gateway Address

To communicate using UDP, TCP/IP, or Modbus/TCP, an IP address and Subnet Mask must be set on the controller. If the controller is to communicate with devices that are not part of the local subnet, then a Gateway Address must also be set. To determine the IP address to be used, consult your IT department. A. Set the IP Address in 20048-20051

If IP Address is 12.40.53.200: 20048 = 12

- 20049 = 4020050 = 5320051 = 200
- B. Set the Subnet Mask in 20064-20067:

If Subnet Mask is 255.255.0.0: 20064 = 255 20065 = 255 20066 = 0 20067 = 0

- C. Set the Gateway, 20080-20083, a gateway of 0 (default), disables it. Gateway 12.40.53.204
 - 20080 = 1220081 = 4020082 = 5320083 = 204

D. After setting the appropriate IP information write a 1 to register 20096 (this may respond with an error, that is normal and can be ignored). This writes the new values to Flash (and deletes the 5100.ini file).

E. Cycle the controller power. Changes will be effective on power up.

- 3. Invoke ModScan32 and configure as per figure 1.2 and 1.3 for TCP operation.
- 4. With TCP communications established poke the Serial RTU address into Modbus register 24620. This is done by double clicking that address on the ModScan32 screen whereupon the "Write Register" dialog will appear, figure 1.14. Com1 is now running the RTU Serial protocol and will only respond to the address value entered. Make sure you are set for Holding Registers and the Length field is less than or equal to 50.
- 5. Invoke another copy of ModScan32 on the same or different computer, with the serial port (COM1) connected to the 5100. Configure as per figure 1.13.
- 6. Modify values as desired by double clicking the screen. As changes are made via RTU serial they will appear on the TCP side, and vice versa.