General Description

The DS28E80 evaluation system (EV system) consists of a single evaluation kit (EV kit) that includes a package of five DS28E80 devices in a 6-pin TDFN package, a DS9120Q+ socket board with RJ11 cable, and a DS9481R-3C7+ USB-to-1-Wire[®]/iButton[®] adapter with USB cable for PC connectivity. This EV system provides the special hardware and software system to exercise the features of the DS28E80. The evaluation software runs under Windows[®] 8, Windows 7, Windows Vista[®], and Windows XP[®] operating systems (32-bit and 64-bit versions). It provides a handy user interface to exercise the DS28E80 features.

The DS28E80 is a user-programmable nonvolatile memory chip. In contrast to the floating-gate storage cells, the DS28E80 employs a storage-cell technology that is gamma radiation resistant. The DS28E80 has 248 bytes of user memory, organized in blocks of 8 bytes. Individual blocks can be write-protected. Each memory block can be written eight times. The DS28E80 communicates over the single-contact 1-Wire bus at standard speed or overdrive speed. Each device has its own guaranteed unique 64-bit registration number factory programmed into the chip. The communication follows the 1-Wire protocol with a 64-bit registration number acting as node address in the case of a multiple-device 1-Wire network.

Ordering Information appears at end of data sheet.

Benefits and Features

- Demonstrates the Features of the DS28E80 IC
- 1-Wire USB Adapter Creates a Virtual COM Port on Any PC
- Fully Compliant with USB 2.0 Specification
- Standard USB Cable Interface
- Software Runs on Windows 8, Windows 7, Windows Vista, and Windows XP Operating Systems (32-Bit and 64-Bit Versions)
- 3.3V ±3% 1-Wire Operating Voltage
- Convenient On-Board Test Points and TO-92 Socket
- Standard RJ11 Connector Interfaces to DS9120 Socket Boards
- Evaluation Software Available Upon Request
- Proven PCB Layout
- Fully Assembled and Tested

EV Kit Contents

QTY	DESCRIPTION
5	DS28E80Q+ 1984-bit gamma radiation resistant 1-Wire memory (6-pin TDFN)
1	DS9120Q+ socket board (TDFN, TO-92)
1	DS9481R-3C7+ USB-to-1-Wire/iButton adapter
1	USB type-A to Mini-USB type-B cable
1	RJ11 cable

1-Wire, and iButton are registered trademarks of Maxim Integrated Products, Inc.

Windows, Windows Vista, and Windows XP are registered trademarks and registered service marks of Microsoft Corporation.



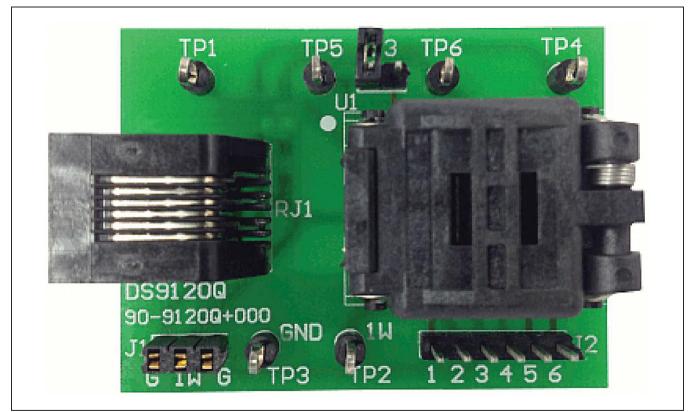


Figure 1. DS9120Q+ Socket Board



Figure 2. DS9481R-3C7+ USB-to-1-Wire/iButton Adapter

Evaluates: DS28E80

DS28E80 EV Kit Files

[FILE	DESCRIPTION
	DS28E80_Evaluation_Program_Setup.exe	Evaluation Software (Installation file)

Quick Start

Required Equipment

- DS28E80 EV kit
- DS9481R-3C7+ USB-to-1-Wire/iButton adapter (included)
- USB type-A to Mini-USB type-B cable (included)
- DS9120Q+ TDFN, TO-92 socket board (included)
- RJ11 cable assembly (included)
- Computer with a Windows 8, Windows 7, Windows Vista, or Windows XP operating system (32-bit or 64-bit version) and a spare USB 2.0 or higher port
- DS28E80Q+ (five devices included)
- DS28E80 EV kit software (go to <u>http://www.maxi-mintegrated.com/en/design/tools/applications/evkit-software</u>, search for the DS28E80, and click the link to download files).

Note: In the following sections, software-related items are identified by bolding. Text in **bold** refers to items directly from the adapter software. Text in **bold and underlined** refers to items from the Windows operating system.

Procedure

The EV kit is fully assembled and tested. Follow the steps below to verify board operation:

- 1) Install the 1-Wire drivers. See the *Installing 1-Wire Drivers* section for instructions.
- 2) The PL-2303 Prolific Driver needs to be installed. See the *Installing the Prolific Device Driver for the DS9481R-3C7+* section for instructions.
 - a) Insert the DS28E80 in the TDFN-EP socket on the DS9120Q+ EV board. See Figure 3 for the pin configuration.

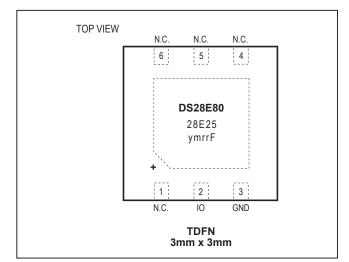


Figure 3. DS28E80 Chip Orientation for TDFN-EP Socket

Note: Pin 1 is indicated by a white dot on the DS9120Q+ PCB.

- 3) Orientation in the socket
 - a) Insert one of the DS28E80 devices into the TDFN-EP socket on the DS28E80 EV kit board.
 - b) Close the burn-in socket.
- 4) Connect the DS9481R-3C7+ with the DS9120Q+ using the RJ11 cable assembly.
- 5) Connect the DS9481R-3C7+ in a vacant USB port on the PC using the USB type-A to Mini-USB type-B cable.

a) Unzip the **DS28E80_EVKit_Rev2.zip** file to a known location.

- 6) Open the **Installation** folder and double-click the **Setup.exe** file.
- When properly installed, the graphical user interface (GUI) window shown in Figure 4 should appear.

Evaluates: DS28E80

Memory	1-Wire Com	mands	Raw 1-Wire		
-	I				
Comma	nas Memory				
	ameter Byte			Number of Bytes	
PB		- Read	Write	1 -	Execute
	v Data				
	cted Memory ameter Byte			Number of Blocks	
PB			Protect	1 💌	Execute
	Remining Cycl	es			
Para PB	ameter Byte			Number of Blocks	Fuenda
L D		•		1 •	Execute
)
Log (Co	lor Meaning: Bl	ue write, gree	n reading)		
]

Figure 4. Initial Screen

Evaluates: DS28E80

- 8) To start communicating with the part, it is necessary to have the DS9481R adapter connected and have a part placed into the DS9120 socket. Once adapter is connected, select Device → Connect.
 - a) The Connect feature can also be used to refresh the connection settings. This is helpful for the user if a new device is attached to the DS9120 adapter. The software needs to reconfigure the number of devices in the network and be ready to work with the new device. Clicking the Connect option refreshes the selections (Figure 5).
- Under Device → Setup, the user can review the settings, change the device, port, 1-Wire speed, or review the selected device ROM ID (Figure 6).
- 10) If more than one device is in the 1-Wire network, it will be displayed under Device → Setup → Device Selection (Figure 7). From there, the user can select any DS28E80 in the 1-Wire network. The new selected device ROM ID will be displayed under Device → Setup → Selected Device ROM ID.

🔯 DS28E	80 EV Software				
File	Device Tools Help	1			
Mei	Setup 🕨	Raw 1-Wire			
	 Connect 				
	Disconnect				
	Parameter Byte		Number of Bytes		
ſ	PB 👻	Read 🕥 Write	1 👻	Execute	
	New Data				

Figure 5. Connect

🔞 DS28	8E80 EV Software	- 1		
File	Device Tools	Help		
Mei	Setup	•	Adapter Port 🔹	COM10 -
	Connect		Device Selection	
ĬĬĬ	 Disconnect 		Speed Selection	
	Parameter Byte		Selected Device ROM ID	f Bytes
	PB	•	Read Write 1	▼ Execute

Figure 6. Review/Change Settings

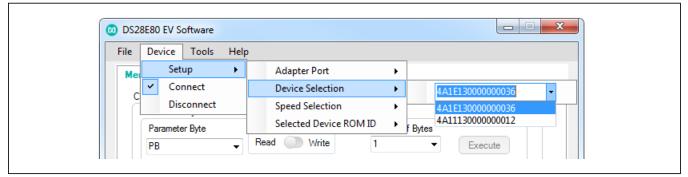


Figure 7. Device Selection

Detailed Description of Software

The DS28E80 evaluation program user interface (Figure 8) has three tabs used to write, read, or protect memory contents by selecting memory blocks or a memory range. Under this tab, the user has the option to write to memory using a programming file instead of typing each memory block content. The **1-Wire Commands** tab is the main tool to evaluate the DS28E80-specific functions. The **Raw 1-Wire** tab provides 1-Wire communication

functions down to the bit level plus power-delivery and program-pulse options. These functions allow the user to manually create communication sequences beyond those found on the **Memory** tab, or to communicate with other 1-Wire devices that are electrically compatible to the DS9481R-3C7+ adapter.

By default, the software opens in simulation mode to provide the user the opportunity to navigate the software and get familiar with the options and software interface.

Memory 1-V	Vire Com	mands	Raw 1	-Wire						
Command Code		Block (55			Paramete	er Bvte	Block A	ddress Oh	•	
Write Block										
Command C	ode	55h								
Restrictions				block must				e bleek		
Protocol Va	riations	None	ere must s	till be at lea	st one writ	te access	left for the	e block		
Error condit	ions		alid param		te d					
		 Wr 	ite access	vrite protec es exhaust	ed					
CS Byte				amming er		he numb	er of remo	aining write		
Co byte		accesses	5.							
				d failed bed d failed bed						
		EEh = Th	e comman	d failed be	cause of a	n interna	program	ming error		
CRCS comp	utation			hifting (leas byte PB inf				nand code a	and	
		Subsequ	ent occurre	ences: Shif	ting the new	w block d	ata (8 byt	es) into the		
				nerator. Th ne byte and						
Parameter B	yte Bitmap									
	Bit 7							Bit 0		
	0	0	0	0	0	0	0	0		
	_		-					1 -		
Data										_
	Byte 8							Byte 1		
	_									
					l					
	CS Byte	e		(Hex)			Ex	ecute		
							_			
Log (Blue: w	riting to pa	art, Green:	reading	from part)						
										_

Figure 8. Main GUI Interface

Evaluates: DS28E80

Menu Bar → File Menu Item

- Save Current Log: Takes the current data contained in the Log group box and gives the user the option to save the log to a specific disk location.
- View Log Files: Opens the interface and shows the current saved logs in the computer so the user can review or print the data.
- Exit: Closes the 1-Wire connection and exits applications.

Menu Bar \rightarrow Device Menu Item

Provides all the connection information and settings required for 1-Wire transactions.

- Setup
 - Adapter Port: COM port used by the DS9481R.
 - **Device Selection:** Indicates the selected DS28E80 device, if currently placed in the DS9490 adapter. If more than one DS28E80, the user can change the current device and interact with a different DS28E80 in the list.

- **Speed Selection:** Option to select the speed the software will connect with the part. The software works by default in standard speed and provides the user an option to change to overdrive or regular speed.
- Selected Device ROM ID: Provides the selected DS28E80 ROM ID number.

Note: Before being able to read this information, the **Connect** option must be selected. In simulation mode, the boxes do not show real information.

- **Connect:** Connects software with the DS9481R, enabling software to interact with the DS28E80 hardware (Figure 9). It is important to have the DS9481R connected to the software before initial start. If the DS9481R is not attached, or the 1-Wire drivers are not installed, the software prompts the user to connect the adapter and closes automatically (Figure 10).
- **Disconnect:** Switches back to simulation mode and disconnects the hardware connection to the DS28E80. To connect back to hardware manipulation, the user must select the **Connect** option again.

File Device Tools Help			
Memory 1-Wire Commands	Raw 1-Wire		
Commands			
User Memory			
Parameter Byte	Write Number of Bytes	Events .	
		Execute	
New Data			
Protected Memory			
Parameter Byte	Number of Blocks		
	Protect 1 -	Execute	
Read Remaining Cycles	Number of Blocks		
Parameter Byte PB		Execute	
FD •	1 👻	Execute	
Log (Blue: writing to part, Green: rea	ding from part)		
Ebg (blue, whiling to part, creek, rea	oing iroin party		

Figure 9. Adapter Connected

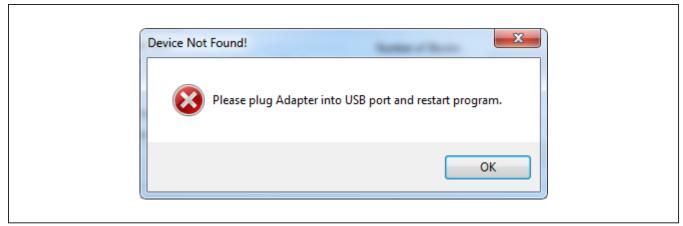


Figure 10. Adapter Not Connected

Menu Bar \rightarrow Tools Menu Item

- Auto Save Data Log: When selected, it automatically saves the log box information into a preloaded default path selected by the software. If this option is not selected, the log will not be saved until the user chooses to save the log by using the File Menu Item → Save Current Log option.
- Select Programming File: The EV kit includes an option under the Memory tab when writing, to select the data from a file instead of typing manually the bytes to write to the memory. To use the From File option under the user memory block, the user must manually make a text file and following the 8-bytes-per-block requirement write the data in the file representing hex values. This provides an option to load contents into the New Data edit box under the Memory tab.

Menu Bar → Help Menu Item

- **About:** Splash screen disable option and Company contact information.
- General: Provides basic DS28E80 information.
- Log Path: Displays the current path where the Auto Save Data Log option stores the log files.

Loading Text File to Software

Using any text editor, the user can create a programming file that can be input to the **New Data** edit

	Prog	ramr	ning	Exan	nple.	txt -	[x	
File	E	dit	Forn	nat	Vie	w I	Help			l
00 00 00 00 00 00	01 01 01 01 01 01 01	02 02 02 02 02 02 02 02	A0 A0 A0 A0 A0 A0	BC BC BC BC BC BC BC	FF FF FF FF FF	08 08 08 08 08 08 08	8D 8D 8D 8D 8D 8D 8D 8D			
•									•	

Figure 11. Programming File Example

box. This helps the user program the device memory faster using a preconfigured file (Figure 11). The only requirement is to follow the format for each block page, with 8 bytes per block in hex format. To do this, click the **Tools** menu item, and select the **Select Programming** File \rightarrow No file selected option. A second popup screen gives the user the option to select the text file (Figure 12) and Figure 13).

DS28E80 EV Software	
ile Device Tools Help	
Memory 1 Auto Save Data Log	
Commands Select Programming File	No file selected
User Memory	
Parameter Byte	Number of Bytes
PB Read Write	1 Execute
New Data	

Figure 12. Import Programming File

🕘 🔵 🗢 📃 Desktop 🕨		•	✓ Search Desktop	
Organize 🔻 New folder				
★ Favorites	File folder	File folder		
■ Desktop Downloads	Medical Demo File folder	parselogic_sp_v1p31 File folder		
 Recent Places Pictures 	pdfbox-1.8.4 File folder	SVN Projects File folder		
➢ Libraries ☑ Documents ▲ Music	Cisco Anyconnect VPN Instructions.txt Text Document	DS2465EV_markup.pdf Adobe Acrobat Document 1.98 MB		
 Pictures Subversion Videos 	instructions.docx Microsoft Word Document 58.5 KB	ProgrammingExample.txt Text Document 200 bytes		
Second Se	sdd_template_v1_0.doc Microsoft Word 97 - 2003 Do 67.5 KB	Coum Type: Text Document Size: 200 bytes Date modified: 5/14/2014 14:17		
 System (C:) OneTouch4 Plus (F:) AutoInfo (\MAXDALFS01) () ∑ 	WINTITIE BIN File 15.4 MB			
File name: SUB30	0(SCS300.7) JBL active subwoofer(1).pdf			-
			Open 🔽 🔿	ancel

Figure 13. File Selection

User Interface \rightarrow Memory Tab

The **Memory** tab has four group boxes: **User Memory**, **Protected Memory**, **Read Remaining Cycles**, and **Log** (Figure 14). Table 1 explains the elements in the **User Memory** group box. Table 2 provides details on the elements in the **Protected Memory** group box, and <u>Table</u> $\underline{3}$ explains the elements in the **Read Remaining Cycles** group box. The **Log** group box displays the 1-Wire communication commands.

_	e Device Tools Help	
M	lemory 1-Wire Commands Raw 1-Wire	
	Commands	
	User Memory	
	Parameter Byte	Number of Bytes
	Block Address 0h 🗸 Read 🕥 Write	1 • Execute
	New Data	
	Protected Memory	
	Parameter Byte	Number of Blocks
	PB Read Protect	Evaputa
	Read Remaining Cycles	
	Parameter Byte	Number of Blocks
	PB	
		1 • Execute
	Log (Blue: writing to part, Green: reading from part)	

Figure 14. Memory Tab Functions

Evaluates: DS28E80

Table 1. Memory Tab (User Memory Group Box)

ELEMENT NAME (TYPE)	PURPOSE	USAGE, SETTINGS
Parameter Byte (drop-down list)	Starting Block Address selection	Indicates the starting address to start writing or reading data from user memory.
Read/Write (toggle button)	Toggles between read or write from selected starting block	Switches between read or write option. The write option opens a second option to input data into the New Data edit box.
Number of Bytes (drop-down list)	Option displays only under reading	Indicates how many bytes will be read from user memory, starting from the selected parameter byte address.
New Data (edit box)	Field in which to type data to write to DS28E80 memory	Disabled under Read option. Open when Write is selected.
Manual/From File (drop-down list) (Figure 15)	Select when user wants to type data into New Data or load data from file	If Manual is selected, the user must type data in hex representation in the field and have at least 8 bytes for an acceptable transaction. If From File is selected, program uses the selected input file and populates data into the edit box. If a file is not selected yet, the program prompts to open file and load contents to edit box.
Execute (button)	Executes the selected options	Sends command to DS28E80 to write or read bytes depending on all selections made.

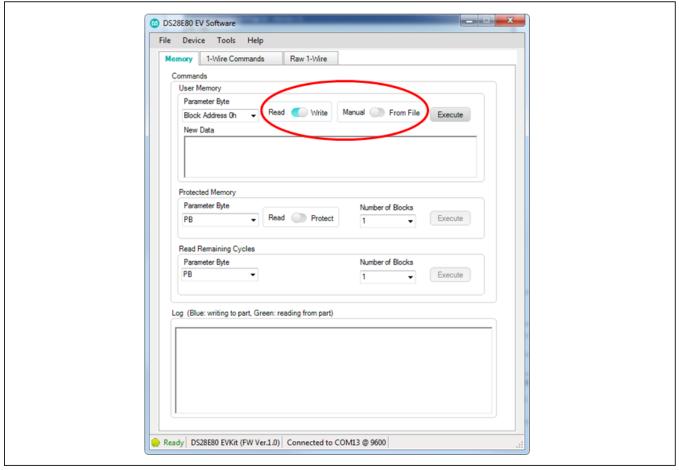


Figure 15. Write Toggle Selected

Table 2. Memory Tab (Protected Memory Group Box)

ELEMENT NAME (TYPE)	PURPOSE	USAGE, SETTINGS
Parameter Byte (drop-down list)	Starting Block Address selection	Indicates the starting address to start protecting or reading protection status.
Read/Protect (toggle-button)	Toggles between read or write protect from selected starting block	The Protect option disables the Number of Blocks and protects only the block selected in the Parameter Byte drop-down list.
Manual/From File (toggle-button)	Select if data is going to be typed in to the New Data edit box or if the data is coming from a preselected file	When Manual is selected, the user must type the byte values in the New Data edit box. When From File is selected, the New Data edit box is populated using the data contained in the loaded text file. To see how to load file, see the Select Programming File option under the Tools menu item.
Number of Blocks (drop-down list)	Option enabled only under reading	Indicates how many blocks are read from user memory, starting from the selected parameter byte address.
Execute (button)	Execute the selected options	Sends command to DS28E80 to protect or read the protection status from the selected Parameter Byte drop-down list.

Table 3. Memory Tab (Read Remaining Cycles Group Box)

ELEMENT NAME (TYPE)	PURPOSE	USAGE, SETTINGS
Parameter Byte (drop-down list)	Starting Block Address selection	Indicates the starting address to read the remaining writing cycles.
Number of Blocks (drop-down list)	Option enabled only under reading	Indicates how many blocks are read from user memory, starting from the selected parameter byte address.
Execute (button)	Executes the selected options	Sends command to DS28E80 to read remaining cycles.

User Interface \rightarrow 1-Wire Commands Tab

The **1-Wire Commands** tab (Figure 16) helps the user understand the process between each command and shows the memory contents separated by memory blocks. Table 4 provides details on each control.

Note: A read of the memory is performed each time the command from the **Command Code** drop-down list is selected, as shown in Figure 17, or each time the block address from the **Parameter Byte** drop-down list is selected, as shown in Figure 18.

Command Code Write Block (55h) Parameter Byte Block Address 1Eh Write Block 55h Restrictions • The memory block must not be write-protected Protocol Variations • There must still be at least one write access left for the block Protocol Variations • Invalid parameter byte • • The block is write protected. • • Internal programming error CS Byte xhe = The command failed because of write accesses exhausted accesses. Eh = The command failed because of an internal programming error ERCS computation First Diffing (least-significant the step) into the cleared CRC-16 generator. Subsequent occurrence: Shifting the new block data (8 bytes) into the cleared CRC-16 generator. The new data is shifted into the CRC-16 generator. Subsequent occurrence: Shifting the new block data (8 bytes) into the cleared CRC-16 generator. The new data is shifted into the CRC-16 generator. Bit 7 Bit 0 0 0 1 1 Data Bit 8 Byte 1 0 0 0 0 0 CS Byte 00 00 00 00 00 Data	-	1-Wire Comm	ands	Raw 1-	-Wire						
Command Code 55h Restrictions The memory block must not be write-protected There must still be at least one write access left for the block Protocol Variations None Error conditions Invalid parameter byte The block is write protected. Write accesses exhausted Internal programming error CS Byte xAh = Success; the upper nibble reports the number of remaining write accesses. 55h = The command failed because of write accesses exhausted EEh = The command failed because of an internal programming error CRCS computation First occurrence: Shifting (least-significant bit first) the command code and then the parameter byte PB into the cleared CRC-16 generator. Subsequent occurrences: Shifting the new block data (8 bytes) into the cleared CRC-16 generator. The new data is shifted into the CRC-16 generator in the same byte and bit sequence as sent by the master. Parameter Byte Bitmap	Command	Code Write B	lock (5	ōh)	•	Paramet	ter Byte	Block Ad	dress 1Eh	-	
Restrictions The memory block must not be write-protected Protocol Variations None Error conditions Invalid parameter byte • The block is write protected. • Write accesses exhausted • Internal programming error CS Byte xAh = Success; the upper nibble reports the number of remaining write accesses exhausted Eb = The command failed because of write accesses exhausted Eb = The command failed because of write accesses exhausted Eb = The command failed because of an internal programming error CRCS computation First occurrence: Shifting (least-significant bit first) the command code and then the parameter byte PB into the cleared CRC-16 generator. Subsequent occurrences: Shifting the new block data (8 bytes) into the cleared CRC-16 generator. Subsequent occurrences: Shifting the new block data (8 bytes) into the cleared CRC-16 generator. Parameter Byte Bitmap Bit 7 Bit 0 0 0 1 1 1 Data Byte 8 Byte 1 00 00 00 00 00 CS Byte 00 00 00 00 00 00 00 00			EEb								
Protocol Variations None Error conditions • Invalid parameter byte • The block is write protected. • Write accesses exhausted • Internal programming error • SAh = Success; the upper nibble reports the number of remaining write accesses. S5h = The command failed because of write accesses exhausted • EEh = The command failed because of write accesses exhausted EEh = The command failed because of an internal programming error • CRCS computation First occurrence: Shifting (least-significant of generator. • Subsequent occurrences: Shifting the new block data (8 bytes) into the cleared CRC-16 generator. Subsequent occurrences: Shifting the new block data (8 bytes) into the cleared CRC-16 generator. • Bit 7 Bit 7 Bit 0 0 O 0 1 1 1 Data Byte 8 Byte 1 0 O 0 0 0 0 0 CS Byte 00 00 00 00 00			 Th 								
The block is write protected. Write accesses exhausted Internal programming error XAh= Success; the upper nibble reports the number of remaining write accesses. S5h = The command failed because the block is write protected 33h = The command failed because of write accesses exhausted EEh = The command failed because of an internal programming error CRCS computation First occurrence: Shifting (least-significant bit first) the command code and then the parameter byte PB into the cleared CRC-16 generator. Subsequent occurrence: Shifting the new block data (8 bytes) into the cleared CRC-16 generator. Subsequent occurrence: Shifting the new block data (8 bytes) into the cleared CRC-16 generator. But 7 Bit 7 Bit 0 0 0 1 1 1 0 Data Byte 8 Byte 1 00 00 00 00 00 00 00 00 00 00 00 00 CS Byte 00 (Hex) Execute	Protoco	Variations		ere must st	ill be at le	ast one wri	te access	left for the	block		
CS Byte xAh = Successe; the upper nibble reports the number of remaining write accesses. 55h = The command failed because the block is write protected 33h = The command failed because of write accesses exhausted EEh = The command failed because of an internal programming error CRCS computation First occurrence: Shifting (least-significant bit first) the command code and then the parameter byte PB into the cleared CRC-16 generator. Subsequent occurrences: Shifting the new block data (8 bytes) into the cleared CRC-16 generator. Subsequent occurrences: Shifting the new block data (8 bytes) into the cleared CRC-16 generator. The new data is shifted into the CRC-16 generator in the same byte and bit sequence as sent by the master. Parameter Byte Bitmap Bit 7 Bit 7 Bit 0 0 0 1 1 1 Data Byte 8 Byte 1 00 00 00 00 00 00 CS Byte 00 00 00 00 00 00	Error co	onditions	• Th	e block is w	rite prote es exhaus	ted					
CRCS computation First occurrence: Shifting (least-significant bit first) the command code and then the parameter byte PB into the cleared CRC-16 generator. Subsequent occurrences: Shifting the new block data (8 bytes) into the cleared CRC-16 generator. The new data is shifted into the CRC-16 generator in the same byte and bit sequence as sent by the master. Parameter Byte Bitmap Bit 0 0 0 1 1 1 0 Data Byte 8 Byte 1 00 00 00 00 00 00 CS Byte 00 0 0 0 0 0 0 0	CS Byte	,	xAh = S accesse 55h = Th 33h = Th	uccess; the es. he command he command	upper nibl d failed be d failed be	cause the cause of w	block is w	rite protect sses exhau	ed sted		
Bit 7 Bit 0 0 0 1 1 1 0 Data Byte 8 Byte 1 00 00 00 00 00 00 00 00 00 Execute CS Byte 00 (Hex) Execute Execute Execute	CRCS o	omputation	First occ then the Subsequ cleared	urrence: Sh parameter t lent occurre CRC-16 ger	ifting (lea byte PB ir inces: Shi nerator. Ti	st-significa ito the clea fting the ne ne new dat	nt bit first) red CRC- w block d a is shifte) the comm 16 generat ata (8 byte d into the C	and code a or. s) into the :RC-16	nd	
0 0 0 1 1 1 0 Data Byte 8 Byte 1 00 00 00 00 00 00 00 00 00 00 00 00 00 Execute	Parame	ter Byte Bitmap									_
Data Byte 8 Byte 1 00 00 00 00 00 CS Byte 00 (Hex) Execute		Bit 7							Bit 0		
Byte 8 Byte 1 00 00 00 00 00 00 CS Byte 00 (Hex) Execute		0	0	0	1	1	1	1	0		
00 00 00 00 00 00 00 CS Byte 00 (Hex) Execute	Data										_
00 00 00 00 00 00 00 CS Byte 00 (Hex) Execute		Bute 8							Bute 1		
CS Byte 00 (Hex) Execute			00	00	00	00	00	00			
							00		00		
Log (Blue: writing to part, Green: reading from part)		CS Byte	00		(Hex	:)		Exe	cute		
	Log (Blu	ue: writing to par	t, Greer	: reading f	rom part)					7

Figure 16. 1-Wire Commands Tab Selected

Table 4. 1-Wire Commands

ELEMENT NAME (TYPE)	PURPOSE	USAGE, SETTINGS
Command Code (drop-down list)	Selects the 1-Wire command	Displays what the 1-Wire command byte is, provides information on the operation, and the expected results for each transaction (see Figure 19).
Parameter Byte (drop-down list)	Starting memory block address	Selects the memory block to be accessed or manipulated.
Parameter Byte Bitmap	Displays the current memory contents of each parameter byte	Displays the parameter byte in bit format.
Data	Displays the contents of the memory block selected	While writing, the user can manipulate the bytes in each block and write back to the memory block selected. While reading, this shows the contents in the block. Depending on the command selected, this section changes and displays results accordingly (see Figures 20–22).
Log	1-Wire communication commands	Displays all 1-Wire commands and transactions for each selected option.

mory 1-V	Wire Comm	ands	Raw 1	1-Wire						
Command Co	de Read I	Memory (F	F0h)	-	Param	eter Byte	Block A	ddress Oh	•	
Read Memo Command C	ry Write B	lock (55	h)							
Restrictions	Write P	rotect Blo	ock (C3h)		issued at a	iny time.				
Protocol Var	riatio Read F	rotection	Block (A	Ah)						
Error condit C3 Byte	uons Read H	Remaining N/A	g Cycles ((A5h)						
CRCS comp								nand code	and	
					nto the clea ifting the bl			tor. ared CRC-1	16	
		generator	r. The shif	ting takes				sequence a		
		transmitte	ed by the A	AI44.						
Parameter B	yte Bitmap									
Parameter B	byte Bitmap Bit 7							Bit 0		
Parameter B		0	0	0	0	0	0	Bit 0		
Parameter B	Bit 7	0	0	0	0	0	0			
Parameter B	Bit 7	0	0	0	0	0	0			
	Bit 7 0	0	0	0	0	0	0	0		
	Bit 7 0 Byte 8							0 Byte 1		
	Bit 7 0	0 88	0	0	0	0 99	0	0]	
	Bit 7 0 Byte 8							0 Byte 1]	
	Bit 7 0 Byte 8							0 Byte 1		
Data	Bit 7 0 Byte 8 55	88	66	77				0 Byte 1		
	Bit 7 0 Byte 8 55	88	66	77				0 Byte 1		
Data	Bit 7 0 Byte 8 55	88 e write, g	66 green rea	77				0 Byte 1		
Data Log (Color M RP CC F0 [55] [88] [Bit 7 0 Byte 8 55	88 e write, g BB] (FF)	66 green rea	77 ding)	88			0 Byte 1		
Data Log (Color M RP CC F0 [55] [88] [R	Bit 7 0 Byte 8 55 Meaning: Blue 00 //CRC [66] [77] [88	88 e write, g BB] (FF)	66 green rea	77 ding)	88			0 Byte 1		
Data Log (Color M RP CC F0 [55] [88] [R Read User	Bit 7 0 Byte 8 55 Meaning: Blue 00 //CRC [66] [77] [88 Memory	88 e write, g BB] [FF] 3] [99] [1	66 green rea	77 ding)	88			0 Byte 1		
Data Log (Color M RP CC F0 [55] [88] [R Read User RP CC F0	Bit 7 0 Byte 8 55 Ieaning: Blue 00 //CRC [66] [77] [88 Memory 00 //CRC [88 e write, g BB] [FF] 3] [99] [1 BB] [FF]	66 preen rea	77 ding) //CRC [3	88 8] [E6]			0 Byte 1		
Data Log (Color M RP CC F0 [55] [88] [R Read User RP CC F0	Bit 7 0 Byte 8 55 Meaning: Blue 00 //CRC [66] [77] [88 Memory	88 e write, g BB] [FF] 3] [99] [1 BB] [FF]	66 preen rea	77 ding) //CRC [3	88 8] [E6]			0 Byte 1		
Data Log (Color M RP CC F0 [55] [88] [R Read User RP CC F0 [55] [88] [Bit 7 0 Byte 8 55 Ieaning: Blue 00 //CRC [66] [77] [88 Memory 00 //CRC [88 e write, g BB] [FF] 3] [99] [1 BB] [FF]	66 preen rea	77 ding) //CRC [3	88 8] [E6]			0 Byte 1		•

Figure 17. Read Memory Occurs When Command Code Selected

	1-Wire Com			1-Wire	/					_
Comman	d Code Read	Memory	(F0h)	-	Paran	ieter Byte		ddress 1h	-	
Read M		1 ===:						ddress 1h	^	
Restric	Ind Code	F0h	he comma	and can be	issued at a	any time		ddress 2h ddress 3h		
	Variations		one	ind can be	issued at	any une.	_	ddress 4h		
	onditions		valid para	meter byte				JJ CL		
CS Byte	omputation	N/A		Ohio da						
CRUST	omputation	then the Subseq generat	paramete	rrences: Sl ifting takes	into the cle hifting the b	ared CRC lock data	-16 gener into the cl	mand code ator. eared CRC- sequence a	16	
	Bit 7 0	0	0	0	0	0	0	Bit 0]	
Data										5 II
								Byte 1		
	Byte 8				0.0	00	00	00		
	Byte 8 00	00	00	00	00	00	00	00		
		00	00	00	00	00	00	00		
Log (Col		1			00	00		00		
	00 lor Meaning: Bl	ue write,	green rea		00	00		00		
RP CC [55] [8 R Read U	00	ue write, [BB] [FF 38] [99]	green rea] [11] [00] ;	ading)		00		00	*	

Figure 18. Read Memory Occurs When Parameter Byte Selected

Mei	nory 1-W	ire Comr	nands	Raw 1	-Wire						
	Command Cod	e Read	Memory (F	⁻ 0h)	-	Parame	eter Byte	Block Ad	ldress Oh	-	
	Read Memory		FOL								
	Command Co Restrictions	ode	F0h None, Th	e comman	nd can be i	ssued at a	nv time.				
	Protocol Varia	ations	 No 		la carroc i	00000000	ny anto:				
	Error condition	ons		alid param	eter byte						
	CS Byte CRCS compu	utation	N/A First occu	urrence: SI	hifting (lea	se-signific	ant bit first) the comm	and code	and	
	citee compe	lauon				to the clea					
									ared CRC-1		
				r. The shift ed by the A		place in the	e same bit	and byte s	equence a	5	
	Parameter By	Bit 7							Bit 0	1	
	Parameter By	Bit 7 0	0	0	0	0	0	0	0		
		Bit 7 0 Byte 8							0 Byte 1		
		Bit 7 0	0	66	77	0	0 99	0	0		
		Bit 7 0 Byte 8 55 eaning: Blu	88 Je write, g	66 preen read	77				0 Byte 1		

Figure 19. Read Memory Selected

	[ools]	mands Raw 1-Wire
Command Code	Read	Protection Block (AAh) Parameter Byte Block Address 1Eh
Read Block	Protection	
Command C		AAh
Restrictions		None. The command can be issued at any time.
Protocol Var		None
Error conditi CS Byte	ons	Invalid parameter byte N/A
CRCS comp	utation	Shifting (least-significant bit first) the command code and then the parameter
		byte PB into the cleared CRC-16 generator.
	/te Bitmap Bit 7 0	Bit 0 0 0 1 1 1 1 0
Data	Bit 7 0	Bit 0 0 0 1 1 1 1 0
	Bit 7 0 Byte 8	Bit 0 0 0 1 1 1 1 0 Byte 1
	Bit 7 0	Bit 0 0 0 1 1 1 1 0
	Bit 7 0 Byte 8 00	Bit 0 0 0 1 1 1 1 0 Byte 1 00 00 00 00 00 00
	Bit 7 0 Byte 8 00	Bit 0 0 0 1 1 1 1 0 Byte 1
Data	Bit 7 0 Byte 8 00 Protecti	Bit 0 0 0 1 1 1 1 0 Byte 1 00 00 00 00 00 00
Data Log (Blue: w	Bit 7 0 Byte 8 00 Protecting to partiting to partiting to partiting to partiting to partition (0) 1E //CRC 00] [00] [0] (0)	Bit 0 0 0 1 1 1 1 0 Byte 1 00 00 00 00 00 00 00 ion Status 0F art, Green: reading from part) [3B] [F7] D0] [00] [00] //CRC [FF] [FF] Satus

Figure 20. Read Protection Selected

Evaluates: DS28E80

Memory	1-Wire Com	nmands Raw 1-Wire
Comman	d Code Read	Remaining Cycles (A5h) Parameter Byte Block Address 1Eh
Read	Remaining Cycle	05
	nand Code	A5h
	ictions	None. The command can be issued at any time.
	col Variations	None
	conditions	Invalid parameter byte
CS B		N/A
CRCS	computation	Shifting (least-significant bit first) the command code and then the parameter byte PB into the cleared CRC-16 generator.
Data	Bit 7 0	Bit 0 0 0 1 1 1 1 0
Data	0	
<	0 Remaini	0 0 1 1 1 1 0
	0 Remaini Blue: writing to pa	0 0 1 1 1 1 0
Log (I RP C [00] R Read	0 Remaini Blue: writing to pa C F0 1E //CRC [00] [00] [00] B Remaining Writi	0 0 1 1 1 1 0 hig Writing Cycles 08 wart, Green: reading from part) C [38] [F7] [00] [00] [00] //CRC [FF] [FF]
Log (I RP C [00] R Reac Para	0 Remaini Blue: writing to pa IC F0 1E //CRC [00] [00] [00] [00]	0 0 1 1 1 1 0 hig Writing Cycles 08 bart, Green: reading from part) C [38] [F7] [00] [00] [00] //CRC [FF] [FF] ing Cycles

Figure 21. Read Remaining Cycles Selected

Evaluates: DS28E80

Write Protect Block Parameter Byte Block Address 1Eh Write Protect Block Command Code C3h Restrictions None. The command can be issued at any time. Protocol Variations Protocol Variations None Invalid parameter byte Error conditions Invalid parameter byte Invalid parameter byte CS Byte AAh = Success 55h = The command failed because the block is already write protected EEh = The command failed because of an internal programming error CRCS computation Shifting (least-significant bit first) the command code and then the parameter byte PB into the cleared CRC-16 generator.
Command Code C3h Restrictions None. The command can be issued at any time. Protocol Variations None Error conditions Invalid parameter byte • Invalid parameter byte • The block is already write protected • Internal programming error CS Byte AAh = Success 55h = The command failed because the block is already write protected EEh = The command failed because of an internal programming error CRCS computation Shifting (least-significant bit first) the command code and then the parameter
Restrictions None. The command can be issued at any time. Protocol Variations None Error conditions Invalid parameter byte The block is already write protected Internal programming error CS Byte AAh = Success 55h = The command failed because the block is already write protected EEh = The command failed because of an internal programming error CRCS computation Shifting (least-significant bit first) the command code and then the parameter
Protocol Variations None Error conditions Invalid parameter byte • Invalid parameter byte • The block is already write protected • Internal programming error CS Byte AAh = Success 55h = The command failed because the block is already write protected EEh = The command failed because of an internal programming error CRCS computation Shifting (least-significant bit first) the command code and then the parameter
Error conditions • Invalid parameter byte • The block is already write protected • Internal programming error CS Byte AAh = Success 55h = The command failed because the block is already write protected EEh = The command failed because of an internal programming error CRCS computation Shifting (least-significant bit first) the command code and then the parameter
The block is already write protected Internal programming error CS Byte AAh = Success 55h = The command failed because the block is already write protected EEh = The command failed because of an internal programming error CRCS computation Shifting (least-significant bit first) the command code and then the parameter
Internal programming error AAh = Success S5h = The command failed because the block is already write protected EEh = The command failed because of an internal programming error CRCS computation Shifting (least-significant bit first) the command code and then the parameter
CS Byte AAh = Success 55h = The command failed because the block is already write protected EEh = The command failed because of an internal programming error CRCS computation Shifting (least-significant bit first) the command code and then the parameter
55h = The command failed because the block is already write protected EEh = The command failed because of an internal programming error CRCS computation Shifting (least-significant bit first) the command code and then the parameter
EEh = The command failed because of an internal programming error CRCS computation Shifting (least-significant bit first) the command code and then the parameter
CRCS computation Shifting (least-significant bit first) the command code and then the parameter
0 0 0 1 1 1 0
Data
Data
Byte 8 Byte 1
Byte 8 Byte 1
Byte 8 Byte 1 00 00 00 00 00 00 00 00
Byte 8 Byte 1
Byte 8 Byte 1 00 00 00 00 00 00 CS Byte 00 (Hex) Execute Execute
Byte 8 Byte 1 00 00 00 00 00 00 00 00
Byte 8 Byte 1 00 00 00 00 00 00 CS Byte 00 (Hex) Execute Execute
Byte 8 Byte 1 00 00 00 00 00 00 CS Byte 00 (Hex) Execute
Byte 8 Byte 1 00 00 00 00 00 00 CS Byte 00 (Hex) Execute Execute Log (Blue: writing to part; Green: reading from part) RP CC F0 1E //CRC [38] [F7] 1
Byte 8 Byte 1 00 00 00 00 00 00 00 CS Byte 00 (Hex) Execute Execute Log (Blue: writing to part, Green: reading from part) Execute France Fran
Byte 8 Byte 1 00 00 00 00 00 00 00 CS Byte 00 (Hex) Execute Exec
Byte 8 Byte 1 00 00 00 00 00 00 00 CS Byte 00 (Hex) Execute Execute Log (Blue: writing to part, Green: reading from part) Execute Image: CC F0 1E //CRC [38] [F7] [00] [00] [00] [00] [00] [00] //CRC [FF] [FF] Image: CC F0 1E //CRC [38] [F7]
Byte 8 Byte 1 00 00 00 00 00 00 00 CS Byte 00 (Hex) Execute Execute Log (Blue: writing to part, Green: reading from part) Execute Execute Execute RP CC F0 1E //CRC [38] [F7] [00] [00] [00] [00] [00] [00] //CRC [FF] [FF] Image: Comparison of the secure of the secu
Byte 8 Byte 1 00 00 00 00 00 00 00 CS Byte 00 (Hex) Execute Execute Log (Blue: writing to part, Green: reading from part) Execute Execute Execute RP CC F0 1E //CRC [3B] [F7] [00] [00] [00] [00] [00] [00] //CRC [FF] [FF] Image: Comparison of the part o
Byte 8 Byte 1 00 00 00 00 00 00 00 CS Byte 00 (Hex) Execute Execute Log (Blue: writing to part, Green: reading from part) Execute Execute Execute RP CC F0 1E //CRC [38] [F7] [00] [00] [00] [00] [00] [00] //CRC [FF] [FF] Image: Comparison of the secure of the secu

Figure 22. Write Protect Block Selected

User Interface \rightarrow Raw 1-Wire Tab

The **Raw 1-Wire** tab (Figure 23) has two main group boxes: **Low Level** and **ROM Level**. Table 5 details the elements in the **Low Level** group box and Table 6 details the **ROM Level** elements. The **Low Level** group box provides the low-level 1-Wire primitives that can be used to construct any 1-Wire communication sequence. Choices made in the **Low Level** group box (e.g., speed, which toggles between standard and overdrive) apply also to ROM-level functions. The **ROM Level** group box has functions that implement the 1-Wire ROM function commands that use the 64-bit ROM ID that each 1-Wire slave device has for device discovery and selection.

Memor	ory 1-Wire Commands Raw 1-Wire
Lov	w Level
	Reset Standard Overdrive Write Bit Logic 1 Logic 0
	Strong Pullup after Next Byte
	Strong Pullup after Next Byte Strong Pullup after Next Bit
	Set Power Normal Read Bytes 1 Number of Bytes
	Deliver Power Read Bit Receives Least Significant Bit
RO	DM Level
	ROM Used with ROM Commands 4AC93100000000C
	Reset Search ROM (first) Reset Search ROM (next)
	Reset-Resume Reset Match ROM Reset-OD Match ROM
	Reset Read ROM Reset-Skip ROM Reset-OD Skip ROM
Log	g (Blue: writing to part, Green: reading from part)
Г	

Figure 23. Raw 1-Wire Tab Functions

Table 5. Raw 1-Wire Tab (Low Level Group Box)

ELEMENT NAME (TYPE)	DESCRIPTION
Reset (button)	Generates a 1-Wire reset at the speed specified by the toggle button.
Standard/Overdrive (toggle button)	Defines the speed to be used for 1-Wire communication.
Strong Pullup after Next Byte (radio button)	Starts the 1-Wire strong-pullup power delivery after the next byte (either read or write).
Strong Pullup after Next Bit (radio button)	Starts the 1-Wire strong-pullup power delivery after the next communication bit (either read or write).
Set Power Normal (radio button)	Ends the 1-Wire strong-pullup power-delivery mode. Returns the 1-Wire bus from power-down state to normal power.
Deliver Power (button)	Delivers power to the device at the power selected by the radio buttons.
Write Bit (button)	Generates a write time slot on the 1-Wire bus. Bit sent is specified in the Logic 1 Logic 0 toggle button.
Write Byte(s) (button)	Transmits the bytes displayed in the Write Byte(s) edit box/drop-down list on the 1-Wire bus.
Read Bytes (button)	Reads as many bytes from the 1-Wire bus, specified by the Number of Bytes to read counter.
Number of Bytes (drop-down list)	Specifies the number of bytes to be read when using the Read Bytes button.
Read Bit (button)	Generates a read-data time slot on the 1-Wire bus.

Table 6. Raw 1-Wire Tab (ROM Level Group Box)

ELEMENT NAME (TYPE)	DESCRIPTION	
ROM Used with ROM Commands (edit box)	The data in this field is used with the ROM function buttons to select a particular 1-Wire slave device. This field can be manually edited to input the ROM ID of the device to be selected. It is automatically filled when using one of the following buttons: Reset Search-ROM (first) , Reset Search-ROM (next) , or Reset Read ROM .	
Reset Search ROM (first) (button)	Sends a 1-Wire reset and performs the search-ROM sequence to discover the first 1-Wire slave device on the network. The ROM IDs and binary search sequence, not physical location, determine the order of the devices discovered. Refer to Application Note 187: <i>1-Wire Search Algorithm</i> for details.	
Reset Search-ROM (next) (button)	Sends a 1-Wire reset and continues the search-ROM sequence from where the last binary search left off and finds the next 1-Wire slave device on the bus.	
Reset-Resume (button)	Sends a 1-Wire reset, followed by the Resume command, A5 (hex).	
Reset Match ROM (button)	Sends the 1-Wire reset followed by the Match ROM command (55h), followed by the 8 bytes of the ROM ID in the ROM Used with ROM Commands . A warning is displayed if the ROM Used with ROM Commands edit box is empty.	
Reset-OD Match ROM (button)	Sends the 1-Wire reset, followed by the Overdrive Match ROM command (69h), followed by the 8 bytes of the ROM ID in the ROM Used with ROM Commands edit box.	
Reset Read ROM (button)	Sends the 1-Wire reset followed by the Read ROM command (33h), and then reads the 64- bit ROM ID of the 1-Wire slave device. The CRC8 within the number is checked to verify the ROM ID's validity. A warning is logged if the CRC8 is not valid. The ROM ID is also loaded into the ROM Used with ROM Commands edit box.	
Reset-Skip-ROM (button)	Sends a 1-Wire reset followed by the CCh Skip ROM command. This selects all devices on the 1-Wire bus.	
Reset-OD Skip ROM (button)	Sends a 1-Wire Reset followed by the 3Ch Overdrive Skip ROM command. This sets all devices on the 1-Wire bus to Overdrive speed.	

CODE	1-WIRE ACTIVITY	DESCRIPTION
R	Reset Pulse	The 1-Wire master generates a reset pulse.
RP	Presence Pulse	The 1-Wire master generates a reset pulse and receives a presence pulse.
RN	No Presence Pulse	The 1-Wire master generates a reset pulse and does not receive a presence pulse.
НН	Writing 1 byte	The 1-Wire master transmits a single byte. The byte value (a pair of hex digits) is shown in place of HH.
[HH]	Reading 1 byte	The 1-Wire master reads a single byte. The byte value (a pair of hex digits) is shown in place of HH.
1b	Write a 1 bit	The 1-Wire master generates a write-one time slot.
0b	Write a 0 bit	The 1-Wire master generates a write-zero time slot.
[1b]	Read 1 bit	The 1-Wire master generates a read-data time slot and reads 1.
[0b]	Read 0 bity	The 1-Wire master generates a read-data time slot and reads 0.
//	(none)	Lines beginning with // are comments.
<ovd></ovd>	Overdrive Speed	Overdrive speed mode.
<std></std>	Standard Speed	Standard speed mode.

Table 7. Communication Log Decoder

Table 8. RJ11 Pin Assignment

PIN NUMBER	SIGNAL NAME	DESCRIPTION
1	VCC	Not used for the DS28E80
2	VCC GND	Not used for the DS28E80
3	1W	1-Wire data
4	GND	1-Wire ground geference
5	PULSE	Not used for the DS28E80
6	(Reserved)	Not used for the DS28E80

Detailed Description of Hardware

RJ11 Pin Assignment

<u>Table 8</u> shows the pin assignment at RJ11 on the DS9120Q+ socket board and the DS9481R-3C7+ USB-to-1-Wire/iButton adapter.

Evaluation Socket Schematic and Layout

 $\underline{\mbox{Figure 24}}$ and $\underline{\mbox{Figure 25}}$ show the schematic and layout of the socket board.

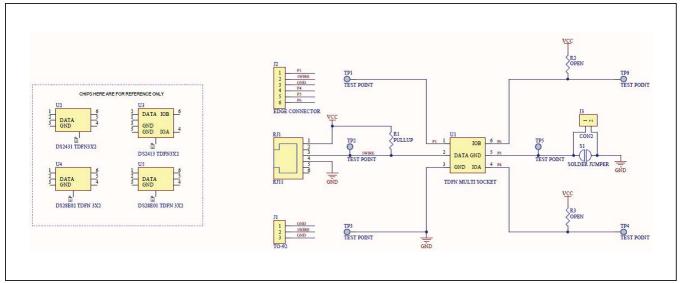


Figure 24. DS9120Q+ Socket Board Schematic

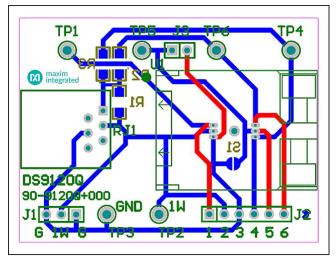


Figure 25. DS9120Q+ Socket Board Composite Layout

Table 9. PCB Color Legend

COLOR	DESCRIPTION
Red	Top metal layer
Green	Top silk screen
Brown	Bottom silk screen
Blue	Bottom metal layer
Grey	All metal layers
Purple	Board outline

Evaluates: DS28E80

Appendix A: Extended Setup Guide

Installing the Prolific Device Driver for the DS9481R-3C7+

The DS9481R-3C7+ USB-to-1-Wire/iButton adapter uses both the Prolific PL-2303HXD and a microcontroller to provide a 1-Wire port on the computer. Many Microsoft operating systems have a version of the PL-2303 Prolific Driver preloaded. Plugging in the DS9481R-3C7+ device for the first time often completes the installation. If the Microsoft operating system in question cannot install the device driver, then do the following:

- 1) Unplug the DS9481R-3C7+.
- Download the driver file labeled PL2303_Prolific_ DriverInstaller_v1_9_0.zip or newer from http://files. maximintegrated.com/sia_bu/public/PL2303_Prolific_ DriverInstaller.zip.

- Unzip the downloaded archive and run the executable file that begins with PL2303_Prolific_DriverInstaller. Follow the directions of the Install Wizard until the PL-2303 USB-to-serial driver install is finished.
- 4) Close the installation by clicking the **Finish** button.
- 5) Verify correct installation of the virtual COM port by inserting the DS9481R-3C7+ into a spare USB port on the computer. Check the COM port by looking in Control Panel → System Hardware → Device Manager and expand Ports (COM & LPT). If the driver installed correctly it should be displayed, as in the example shown in <u>Figure 26</u>. Note that your COM port number may be different. **Warning:** The 1-Wire drivers only support COM ports up to COM15.
- 6) Once installed, return to the *Procedure* section and execute Step 5.

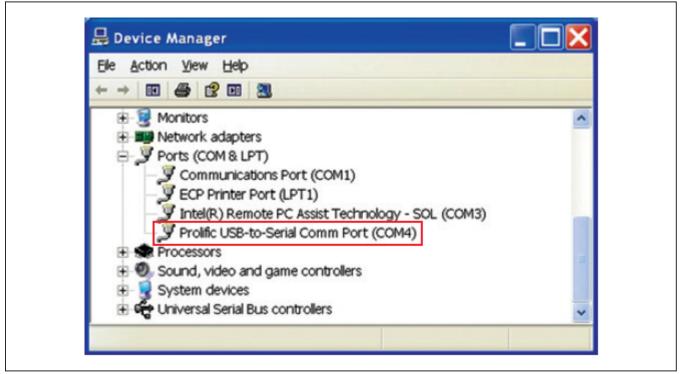


Figure 26. DS9481R-3C7+ "Prolific" COM Port

Installing 1-Wire Drivers

The DS9481R-3C7+ USB-to-1-Wire/iButton adapter also requires 1-Wire drivers to operate the 1-Wire port. Unless you have been using 1-Wire devices before (e.g., with a different adapter), the drivers need to be installed before the EV kit software can function. Follow the steps below to install the 1-Wire Drivers software package. For expanded installation details, refer to Tutorial 4373: *OneWireViewer and iButton Quick Start Guide*, which is available at www.maximintegrated.com/app-notes/index.mvp/id/4373.

To download the 1-Wire Drivers software package, do the following:

 Go to <u>www.maximintegrated.com/1-wiredrivers</u>. Click on the button that takes you to the download page. Select the applicable version of the Windows operating system, and then select the file that corresponds to your computer's configuration (i.e., 32-Bit 1-Wire Drivers or 64-Bit 1-Wire Drivers), and finally, click **Download**.

- 2) When prompted with the question **Do you want to run or save this file?**, select **Run**.
- 3) When you get a security warning that states **Do you** want to run the software?, select **Run**.
- 4) Read and check the box if you accept the license agreement and click **Install**.
- 5) Click Finish to exit the Setup Wizard.
- 6) Return to the *Procedure* section and execute Step 5.

Installing Microsoft .NET Framework

The DS28E80 EV kit software requires the Microsoft .NET Framework Version 3.5 SP1 for the program to run. To check whether it is installed, look in the Control Panel under Add/Remove Programs. If Microsoft.NET is not listed, it is not installed. For installation instructions and download, go to <u>http://msdn.microsoft.com/en-us/vstudio/aa496123</u>.

Evaluates: DS28E80

Ordering Information

PART	TYPE	
DS28E80EVKIT#	EV System	

#Denotes RoHS compliant.

Evaluates: DS28E80

Revision History

REVISION NUMBER	REVISION DATE	DESCRIPTION	PAGES CHANGED
0	11/14	Initial release	—

For pricing, delivery, and ordering information, please contact Maxim Direct at 1-888-629-4642, or visit Maxim Integrated's website at www.maximintegrated.com.

Maxim Integrated cannot assume responsibility for use of any circuitry other than circuitry entirely embodied in a Maxim Integrated product. No circuit patent licenses are implied. Maxim Integrated reserves the right to change the circuitry and specifications without notice at any time.





Общество с ограниченной ответственностью «МосЧип» ИНН 7719860671 / КПП 771901001 Адрес: 105318, г.Москва, ул.Щербаковская д.З, офис 1107

Данный компонент на территории Российской Федерации

Вы можете приобрести в компании MosChip.

Для оперативного оформления запроса Вам необходимо перейти по данной ссылке:

http://moschip.ru/get-element

Вы можете разместить у нас заказ для любого Вашего проекта, будь то серийное производство или разработка единичного прибора.

В нашем ассортименте представлены ведущие мировые производители активных и пассивных электронных компонентов.

Нашей специализацией является поставка электронной компонентной базы двойного назначения, продукции таких производителей как XILINX, Intel (ex.ALTERA), Vicor, Microchip, Texas Instruments, Analog Devices, Mini-Circuits, Amphenol, Glenair.

Сотрудничество с глобальными дистрибьюторами электронных компонентов, предоставляет возможность заказывать и получать с международных складов практически любой перечень компонентов в оптимальные для Вас сроки.

На всех этапах разработки и производства наши партнеры могут получить квалифицированную поддержку опытных инженеров.

Система менеджмента качества компании отвечает требованиям в соответствии с ГОСТ Р ИСО 9001, ГОСТ РВ 0015-002 и ЭС РД 009

Офис по работе с юридическими лицами:

105318, г.Москва, ул.Щербаковская д.3, офис 1107, 1118, ДЦ «Щербаковский»

Телефон: +7 495 668-12-70 (многоканальный)

Факс: +7 495 668-12-70 (доб.304)

E-mail: info@moschip.ru

Skype отдела продаж: moschip.ru moschip.ru_4

moschip.ru_6 moschip.ru_9