



Cyclone PRO User Manual

CYCLONEPROUM
Rev. 1.10
January 2008



Purchase Agreement

P&E Microcomputer Systems, Inc. reserves the right to make changes without further notice to any products herein to improve reliability, function, or design. P&E Microcomputer Systems, Inc. does not assume any liability arising out of the application or use of any product or circuit described herein.

This software and accompanying documentation are protected by United States Copyright law and also by International Treaty provisions. Any use of this software in violation of copyright law or the terms of this agreement will be prosecuted.

All the software described in this document is copyrighted by P&E Microcomputer Systems, Inc. Copyright notices have been included in the software.

P&E Microcomputer Systems authorizes you to make archival copies of the software and documentation for the sole purpose of back-up and protecting your investment from loss. Under no circumstances may you copy this software or documentation for the purpose of distribution to others. Under no conditions may you remove the copyright notices from this software or documentation.

This software may be used by one person on as many computers as that person uses, provided that the software is never used on two computers at the same time. P&E expects that group programming projects making use of this software will purchase a copy of the software and documentation for each user in the group. Contact P&E for volume discounts and site licensing agreements.

P&E Microcomputer Systems does not assume any liability for the use of this software beyond the original purchase price of the software. In no event will P&E Microcomputer Systems be liable for additional damages, including any lost profits, lost savings or other incidental or consequential damages arising out of the use or inability to use these programs, even if P&E Microcomputer Systems has been advised of the possibility of such damage.

By using this software, you accept the terms of this agreement.

MS-DOS & Windows are registered trademarks of Microsoft Corporation. IBM is a registered trademark of IBM corporation.

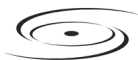
Freescale™ and the Freescale logo are trademarks of Freescale Semiconductor, Inc. All other product or service names are the property of their respective owners. © Freescale Semiconductor, Inc. 2004.

P&E Microcomputer Systems, Inc.
P.O. Box 2044
Woburn, MA 01888
617-353-9206
<http://www.pemicro.com>

Manual version 1.10

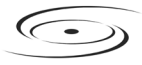


1	INTRODUCTION	1
2	CYCLONE PRO HARDWARE	3
2.1	Cyclone PRO Power Supply	3
2.2	RS232 Communication	3
2.3	Ethernet Communication	4
2.4	USB Communications	4
2.5	Electromechanical Relays	5
2.6	Power Connectors	6
2.7	Jumper Settings	6
2.8	Optional Oscillator (MON08 Only)	7
2.9	Target BDM Connector	7
2.10	Target MON08 Connector	8
2.11	Ribbon Cable	10
2.12	Target Power Management	10
2.13	CompactFlash Port	17
3	STAND-ALONE PROGRAMMER CONFIGURATION	19
3.1	Target Architecture	19
3.2	Specify Programming Script	28
3.3	Script Wizard	31
3.4	Programming Operations	33
3.5	Image Description	34
3.6	Store Image To Cyclone	34
3.7	Save Image/Cyclone Configuration	36
3.8	Configuration Via LCD Menu	36
4	STAND-ALONE PROGRAMMER MANUAL CONTROL	43
4.1	Operation Via Cyclone PRO Buttons	43
4.2	Operation Via LCD Menu (Rev. C Only)	45
5	STAND ALONE PROGRAMMER AUTOMATED CONTROL	50
5.1	Automated .EXE Control Of The Cyclone	50
5.2	Automated .DLL Control Of The Cyclone	54
5.3	Control Of Multiple Cyclone Units	63



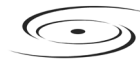
6	TARGET MON08 HEADER PINOUTS	65
6.1	68HC908AB	65
6.2	68HC908AP	66
6.3	68HC908AS	66
6.4	68HC908AT	67
6.5	68HC908AZ	68
6.6	68HC908BD	69
6.7	68HC908EY	69
6.8	68HC908GP	70
6.9	68HC908GR16/32	71
6.10	68HC908GR4/8	72
6.11	68HC908GT	72
6.12	68HC908GZ	73
6.13	68HC908JB1/8	74
6.14	68HC908JB12	75
6.15	68HC908JB16	75
6.16	68HC908JG	76
6.17	68HC908JK	77
6.18	68HC908JL	78
6.19	68HC908JR	78
6.20	68HC908JW	79
6.21	68HC908KX	80
6.22	68HC908LB	81
6.23	68HC908LD	82
6.24	68HC908LJ	82
6.25	68HC908LK	83
6.26	68HC908LT	84
6.27	68HC908LV	85
6.28	68HC908MR4/8	86
6.29	68HC908MR16/32	86
6.30	68HC908QL	87

6.31	68HC908QB	88
6.32	68HC908QC	89
6.33	68HC908QT	90
6.34	68HC908QY	91
6.35	68HC908RF	92
6.36	68HC908RK	92
6.37	68HC908SR	93
7	PC-HOSTED DEBUG/PROGRAMMING SOFTWARE.....	95
7.1	P&E Microcomputer Systems Software	95
7.2	Freescall Software	98
7.3	HC08 P&E Multilink/Cyclone PRO Connections	102
7.4	HCS08 P&E Multilink/Cyclone PRO Connections	121
8	ETHERNET CONFIGURATION	129
8.1	Network Architectures	129
8.2	Network Parameters.....	130
8.3	Internet Protocol	131
8.4	Connecting The Cyclone Device	131
8.5	Cyclone IP Setup Via LCD Menu	133
8.6	Cyclone IP Setup Utility User Interface (ConfigureIP).....	133
8.7	Using ConfigureIP.exe To Configure The Cyclone PRO.....	135
8.8	Managing Multiple Images	137
9	SERIAL PORT CONFIGURATION	140
10	USB PORT CONFIGURATION	141



CYCLONE PRO



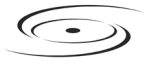


1 INTRODUCTION

The Cyclone PRO is part of P&E's line of stand-alone programming hardware. The Cyclone PRO is able to debug and program Motorola 68HC908 targets in-circuit via the Monitor ROM (MON08 Port). In addition, the Cyclone PRO is able to program and debug the ColdFire V1, HC(S)12(X), HCS08, and RS08 families of microcontrollers through the on-board Background Debug Modules of those processors. The Cyclone PRO acts as a debugging/programming tool. It can communicate with a PC through either RS-232, Ethernet, or USB interfaces. In addition, the Cyclone PRO can function as a stand-alone programmer - once configured properly, it can program target devices independent of a PC.

Some of the features that make the Cyclone PRO so versatile are:

- Software-configurable port-pin settings for Monitor ROM entrance.
- Works with 1.6V to 5V targets with internal bus frequency ranges from 1MHz to 8MHz (MON08) or 16KHz to 50MHz (BDM).
- Automatically detects target internal bus frequency and sets communications speed.
- Automatically cycles target power using electromechanical relays during security protocol (MON08).
- Automatically provides Vpp voltage for RS08 targets during flash erasing and programming operations.
- Provides 9.8304 MHz oscillator signal to overdrive target crystal and RC clock circuitry (MON08).
- PC connection can be achieved with a variety of convenient interfaces: RS-232, USB, or Ethernet.
- Jumper-settable power management schemes for MON08 and BDM targets.
- CompactFlash storage available with purchase of CompactFlash



activation license.

2 CYCLONE PRO HARDWARE

The following is an overview of the features and interfaces of the Cyclone PRO unit.

2.1 Cyclone PRO Power Supply

The Cyclone PRO requires a regulated 6V DC Center Positive power supply with 2.5/5.5mm female plug. The Cyclone PRO derives its power from the Power Jack located on the side of the unit.



Figure 2-1: Cyclone PRO Power Jack

2.2 RS232 Communication

The Cyclone PRO provides a DB9 Female connector to communicate with a host computer through the RS232 communication (115200 Baud, 8 Data bits, No parity, 1 Stop bit).



Figure 2-2: Cyclone PRO RS232 Connector

2.3 Ethernet Communication

The Cyclone PRO provides a standard RJ45 socket to communicate with a host computer through the Ethernet Port (10/100 BaseT).



Figure 2-3: Cyclone PRO Ethernet Connector

2.4 USB Communications

The Cyclone PRO provides a USB connector for Universal Serial Bus communications between the Cyclone PRO and the host computer. The Cyclone PRO is a USB 1.1 compliant device.



Figure 2-4: Cyclone PRO USB Connector

2.5 Electromechanical Relays

Inside the Cyclone PRO, two electromechanical relays are used to cycle target power. The specifications of the relays are as following:

- Maximum switched power:** 30W or 125 VA
- Maximum switched current:** 1A
- Maximum switched voltage:** 150VDC or 300VAC
- UL Rating:** 1A at 30 VDC
1A at 125 VAC

P&E only recommends switching DC voltages up to 30 Volts.

2.6 Power Connectors

The Cyclone PRO provides a Target Power Supply Input Jack and a Target Power Supply Output Jack with 2.5/5.5 mm Pin Diameter. The power jacks are connected or disconnected by two electromechanical relays. When connected, the Center Pin of the Target Power Supply Input Jack is connected to the Center Pin of the Target Power Supply Output Jack. When disconnected, both terminals of the Target Power Supply Output Jack are connected to GND via a 1W, 100 Ohm resistor.



Figure 2-1: Power Connector Locations

2.7 Jumper Settings

Please note that Rev. C of the Cyclone PRO uses different jumper settings than previous versions. The jumpers must be set differently for various power management options that the Cyclone PRO offers. If the Cyclone PRO is not being used to manage the target's power, only Jumper 5 needs to be installed. Please see **Section 2.12 - Target Power Management** for the correct jumper settings for other power management options.

2.8 Optional Oscillator (MON08 Only)

The Cyclone PRO provides a software configurable 9.8304MHz or 4.1952 MHz oscillator clock signal to Pin 13 of the MON08 Connector. The user may use this clock signal to overdrive the target RC or crystal circuitry. If this signal is not used, just leave Pin 13 of the target MON08 header unconnected.

Please note that if the target already uses an oscillator as its clock, the Cyclone PRO will NOT be able to overdrive it. The clock should have sufficient drive to be used with a target system even if the target system has an RC circuit or crystal connected.

2.9 Target BDM Connector

The Cyclone PRO provides a standard 6 pin 0.100-inch pitch dual row 0.025-inch square header for ColdFire V1, 68(S)12(X), 68HCS08, and RS08 targets. The BDM Header uses the Motorola standard pin configuration, listed here for reference:

PIN 1 - BKGD	GND - PIN 2
PIN 3 - NC	RST/Vpp - PIN 4
PIN 5 - NC	Vdd - PIN 6

Please see **Figure 2-2** for the physical location of the BDM connector on the Cyclone PRO.

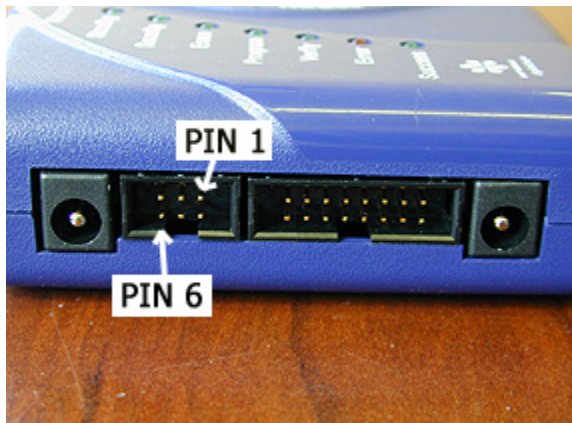


Figure 2-2: 6 Pin BDM Connector Pin Connections

The mechanical drawing is shown in **Figure 2-3**.

2.10 Target MON08 Connector

The Cyclone PRO provides a standard 16-position 0.100-inch pitch dual row 0.025-inch square header for the HC08 targets. The mechanical drawing is shown below.

0.100" Dual Row, 0.025" Square Header

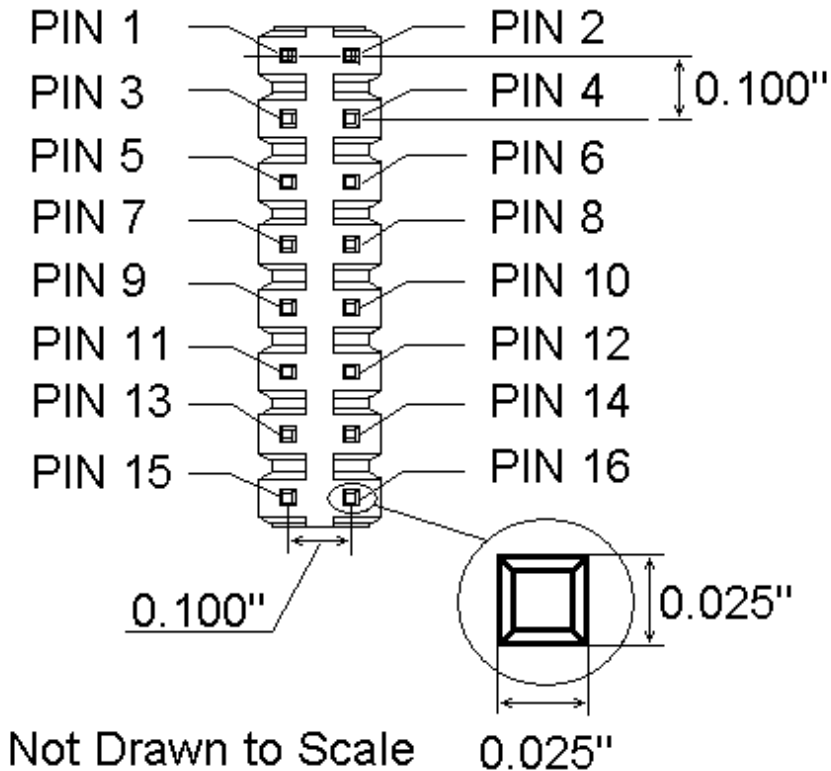


Figure 2-3: 16-Pin Header Mechanical Drawing

The MON08 Connector adopts the standard pin-out from MON08 debugging (as used on different ICS boards) with some modifications. The general pin-out is as follows:

PIN 1 -	NC	GND	- PIN 2
PIN 3 -	NC	RST	- PIN 4
PIN 5 -	NC	IRQ	- PIN 6
PIN 7 -	NC	MON4	- PIN 8
PIN 9 -	NC	MON5	- PIN10
PIN11 -	NC	MON6	- PIN12
PIN13 -	OSC	MON7	- PIN14
PIN15 -	Vout	MON8	- PIN16

The physical locations of the pins are displayed below:

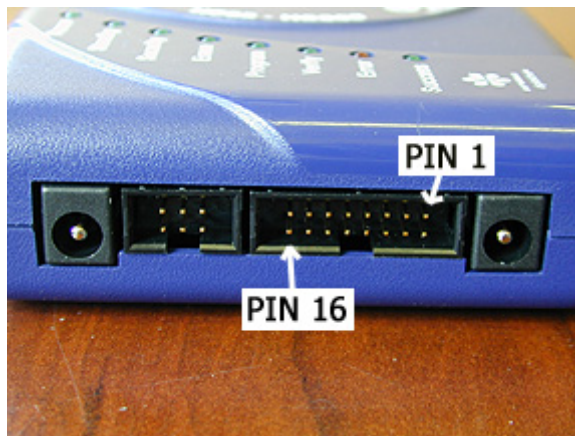


Figure 2-4: 16-Pin MON08 Connector Pin Locations

Please note that **NC** designates that these pins are reserved for future P&E use. **Make sure you do not connect any signal to these lines.**

The **MON4-MON8** signals are software configurable to support connections to different HC08 devices. Depending upon the device, either the MON4 or MON5 pin is the single-wire communications line (which usually corresponds to PORTA0 or PORTB0). The rest of the lines are either No Connect or are port lines which must be put to particular values upon reset. The Cyclone PRO software lists the target processor types and their corresponding pin-outs for user references. The software also selects the single-wire communications line according to the target processor type. The specific

pinouts for different HC08 devices are specified in the configuration software and also in **CHAPTER –** .

2.11 Ribbon Cable

The Cyclone PRO communicates with the target through either a 16-pin ribbon cable (MON08) or 6-pin ribbon cable (BDM). Both have a 0.100-inch centerline dual row socket IDC assembly (not keyed). The ribbon cable is designed such that the Cyclone PRO MON08/BDM Connector and the Target Header have the same pinout, i.e., Pin 1 of the Cyclone PRO MON08 Connector is connected to Pin 1 of the target MON08 Header. **Figure 2-5** sketches the connection mechanism (looking down into the sockets) for the MON08 ribbon cable. The 6-pin BDM ribbon cable uses a similar scheme with fewer pins.

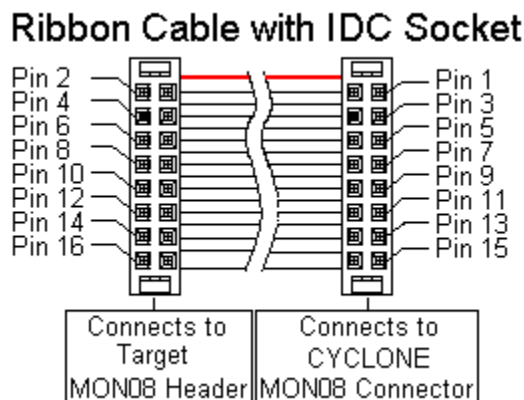


Figure 2-5: Ribbon Cable Diagram When Looking Into IDC Socket

2.12 Target Power Management

Note: Cyclone PRO Rev. C implements different jumper settings than previous revisions of the hardware. Rev. C has added a fifth jumper position, which was necessary for the “Power In Jack ->Power Out Jack” power management setting. Rev. C has also added new target management schemes, including one for BDM Pin6, which are explained below.

The Cyclone PRO provides three target power management schemes for

BDM users and four schemes for MON08 users. The target board may derive power by the use of power jacks, or by the use of MON08 connector Pin 15, or by the use of BDM connector Pin 6, or a combination of both. The options are explained in detail below. Each configuration will have an associated jumper setting that **MUST** be set on the Cyclone PRO. The jumper header is found on the long side of the unit, opposite from the side where the power jack is located.

Note: In order for the Cyclone PRO to handle power sequencing for MON08 targets automatically, power **MUST** be provided to the target system by the Cyclone PRO.

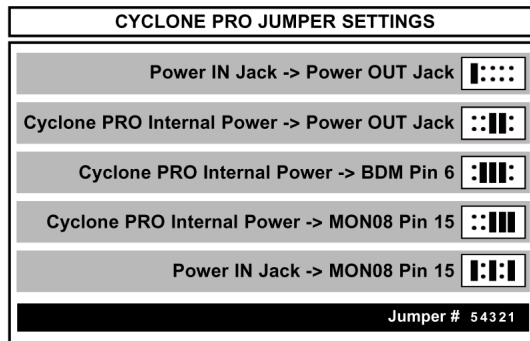


Figure 2-6: Jumper Settings (Rev. C)

2.12.1 Using Power In Jack and Power Out Jack

The target power supply is connected to the Power In Jack of the Cyclone PRO. The Power Out Jack of the Cyclone PRO is connected to the target system. **Figure 2-7** shows the connections.

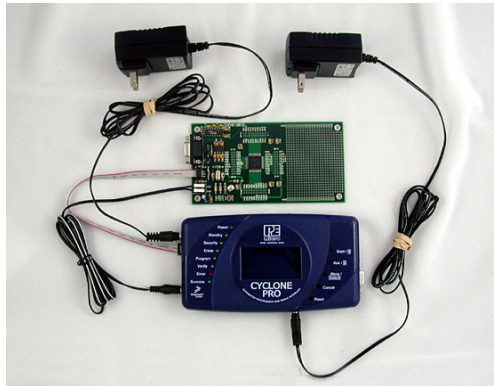


Figure 2-7: Via Power In and Power Out Jacks of Cyclone PRO

All of the jumpers except Jumper 5 should be left open for this mode, as shown in **Figure 2-8**.

Note: This setting differs from the Rev. B Cyclone PRO, which requires no jumpers.



Figure 2-8: Jumper Settings for Target Power Connection via Power In and Power Out Jacks Of Cyclone PRO

Only Jumper 5 is installed.

2.12.2 Using Cyclone PRO Board Power and Power Out Jack

The target power supply is not needed. The Power Out Jack of the Cyclone PRO will act as a center positive power supply to the target system. **Figure 2-9** shows the connections.

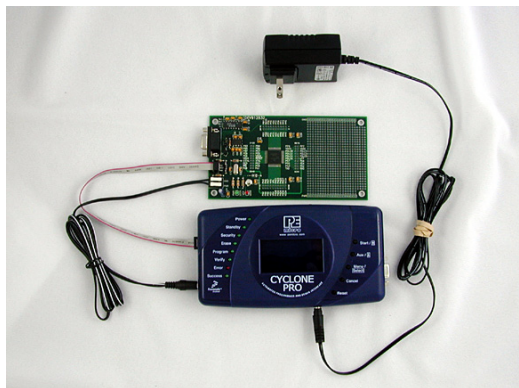


Figure 2-9: Via Cyclone PRO Board Power and Power Out Jack of Cyclone PRO

Jumpers 2 and 3 are enabled. Jumpers 1, 4 and 5 are left open, as shown in **Figure 2-10**.



Figure 2-10: Jumper Settings for Target Power Connection via Cyclone PRO Board Power and Power Out Jack of Cyclone PRO

Jumpers 2 and 3 are installed. Jumpers 1, 4, and 5 are left open.

2.12.3 Using Cyclone PRO Board Power and Pin 15 (for MON08), or Pin 6 (For BDM)

The target power supply is not needed. Pin 15 of the Cyclone PRO MON08 connector provides the appropriate voltage for MON08 targets. Pin 6 of the Cyclone PRO BDM header provides the appropriate voltage for BDM targets. The Power Out jack of the Cyclone PRO connector is not needed. **Figure 2-11** shows the connections.

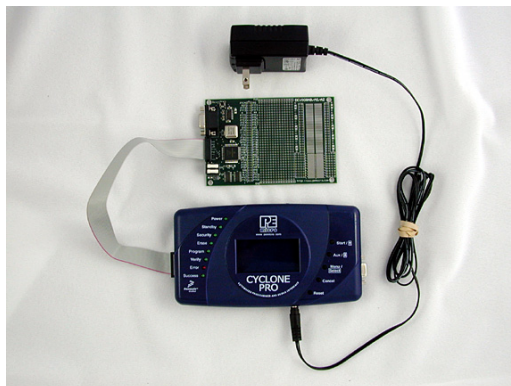


Figure 2-11: Via Cyclone PRO Board Power and Pin 15 of Cyclone PRO MON08 Connector

Jumpers 1, 2, and 3 are all enabled, as shown in **Figure 2-12**



Figure 2-12: Jumper Settings for Target Power Connection via Cyclone PRO Board Power and Pin 15 of Cyclone PRO MON08 Connector

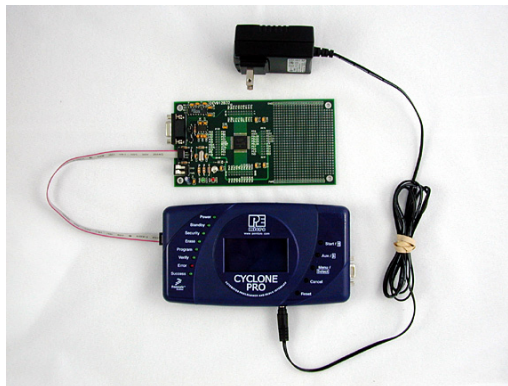


Figure 2-13: Via Cyclone PRO Board Power and Pin 6 of Cyclone PRO BDM Connector

Jumpers 2, 3, and 4 are enabled as shown in **Figure 2-14**.



Figure 2-14: Jumper Settings for Target Power Connection via Cyclone PRO Board Power and Pin 6 of Cyclone PRO BDM Connector

2.12.4 Using Power IN Jack And MON08 Header Pin 15

A center positive power supply is connected to the Power IN Jack of the Cyclone PRO. Pin 15 of the Cyclone PRO MON08 header provides the appropriate voltage for MON08 targets.

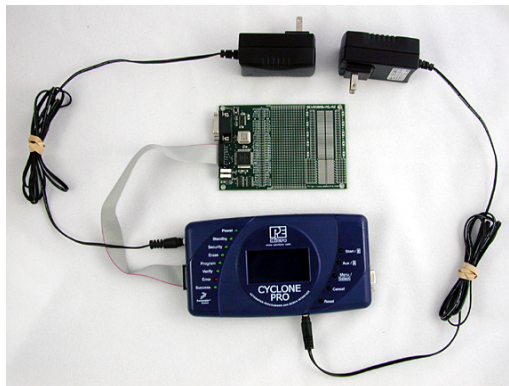


Figure 2-15: Using Power IN Jack And MON08 Header Pin 15



Figure 2-16: Jumper Settings For Power IN Jack And MON08 Header Pin 15

Jumpers 1, 3, and 5 are enabled. Jumpers 2 and 4 are left open.

2.12.5 Target Powered Independently Of Cyclone PRO

For ColdFire V1/HC(S)12(X)/HCS08/RS08 devices, the target may be powered independently of the Cyclone PRO. The user should remove all jumpers on the Cyclone PRO except jumper 5 if they elect to provide a separate power source for the target.

2.13 CompactFlash Port

The CompactFlash port can be activated by purchasing a CompactFlash Activation License from P&E. The CompactFlash port accepts Type I CompactFlash cards. This allows the user to store programming images that are, individually or collectively, larger than the Cyclone's internal memory (7MB). It also makes swapping programming images much more quick and convenient.

Purchase of the Compact Flash activation license unlocks the CompactFlash panels in the included Manage Images utility. Programming images may then be managed on the CompactFlash in exactly the same way as they are in the Cyclone's internal memory. Please see section for more information about

using Manage Images.



Figure 2-17: CompactFlash Port

3 STAND-ALONE PROGRAMMER CONFIGURATION

The Cyclone PRO may act as a Stand-Alone In-Circuit Programmer for the HC08, HCS08, HC(S)12(X), and ColdFire V1 targets. A simple user interface, CREATEIMAGE.EXE, is provided for configuring the Cyclone PRO.

This section describes how to configure the Cyclone PRO for stand-alone programming. The Cyclone PRO does not require a target to be connected when it is being configured. However, the power of the Cyclone PRO must be turned on (indicated by the “Standby” LED), and the PC serial port or Ethernet Port or USB port must be connected to the Cyclone PRO. **Figure 3-1:** shows the configuration dialog with an example configuration. The configuration details are explained in detail below.

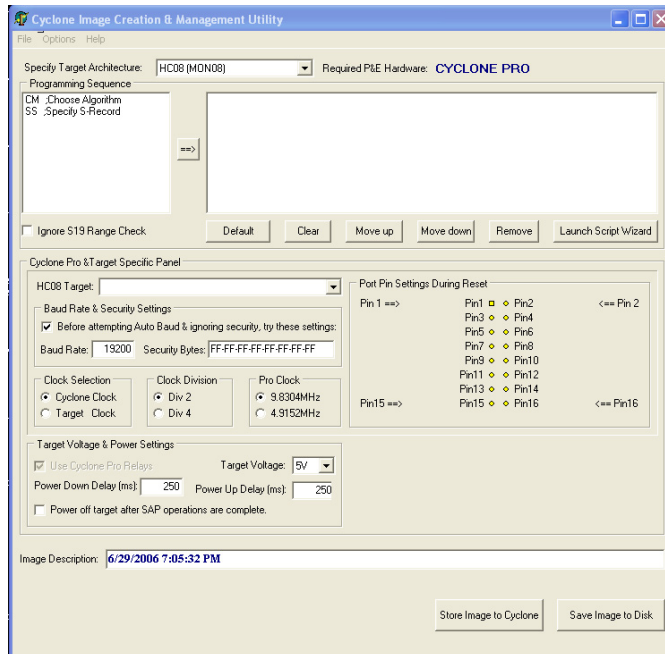


Figure 3-1: Cyclone Image Creation And Management Utility

3.1 Target Architecture

The Cyclone Pro supports ColdFire V1, HC9(S)12(X), HCS08, RS08, and HC08 targets.

The user may select the target architecture by clicking the corresponding tab: ColdFire V1, HC9(S)12(X), HCS08, RS08, or HC08 (MON08).

Alternately, the user may select the target architecture through the File menu. **Figure 3-2** shows this selection in the File menu.

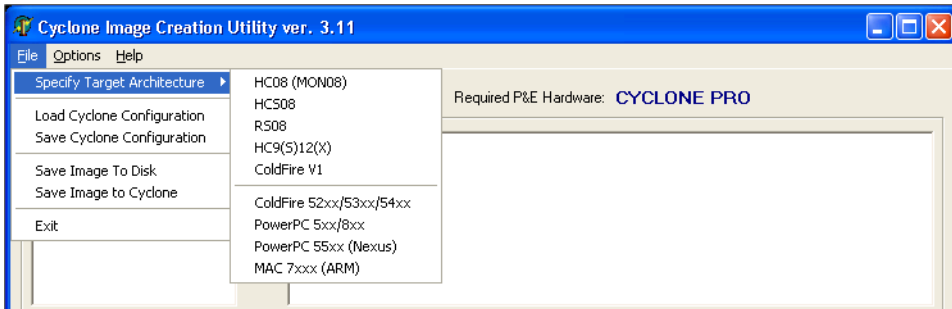


Figure 3-2: Target Class Selection via File Menu

3.1.1 ColdFire V1

The user may configure the Cyclone PRO to operate on a ColdFire V1 target by selecting ColdFire V1 from the Specify Target Architecture drop-down menu.

Alternatively, the user may select the ColdFire V1 target through the File menu.

Figure 3-3 shows the ColdFire V1 specification configuration.

The target BDM header connections are shown on the right-hand side for user reference. The Cyclone PRO will automatically detect the communications speed.

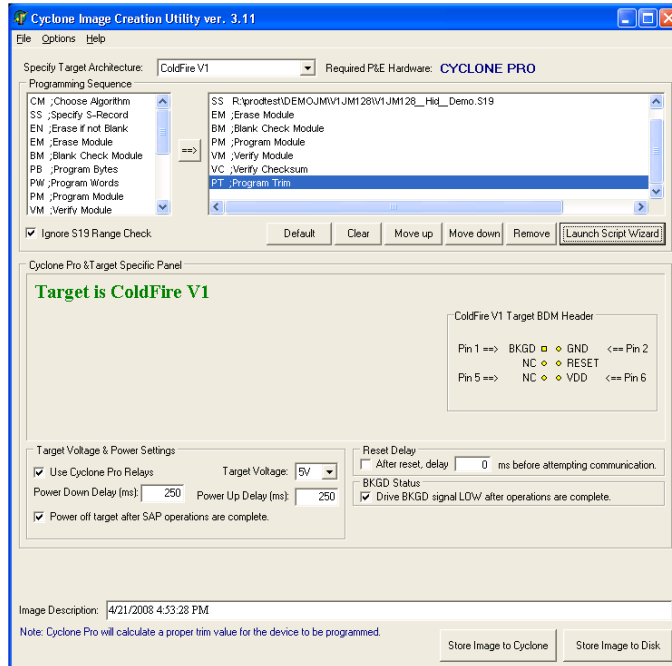


Figure 3-3: ColdFire V1 Class Settings

3.1.2 HC9(S)12(X)

The user may configure the Cyclone PRO to operate on a HC9(S)12(X) target by selecting HC9(S)12(X) from the Specify Target Architecture drop-down menu.

Alternatively, the user may select the HC9(S)12(X) target through the File menu.

Figure 3-4 shows the HC9(S)12(X) specification configuration.

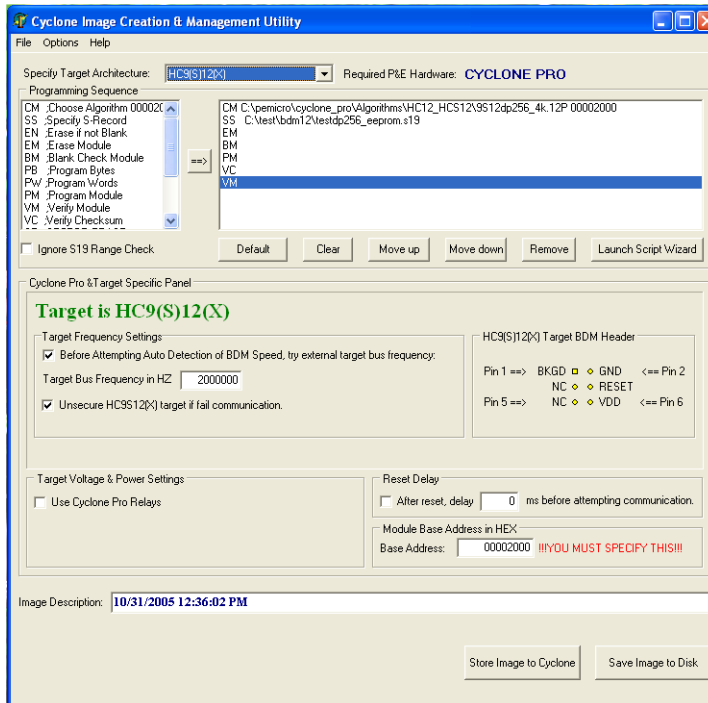


Figure 3-4: HC9(S)12(X) Class Settings

The target BDM header connections are shown on the right-hand side for user reference.

The user may specify the target internal bus frequency in the edit box provided. If the checkbox in the Target Frequency Settings is checked, the Cyclone PRO will first contact the target with the specified frequency. If the Cyclone PRO contacts the target successfully with the provided frequency, the Cyclone PRO will continue executing the rest of the commands. If the operation is unsuccessful, or if the checkbox is not checked, the Cyclone PRO will automatically detect the correct communication speed and begin executing the commands. The autodetection process will add time to the programming process.

Depending on the programming algorithm, the user may be required to enter the base address for the module. If the programming algorithm requires a

base address, the Module Base Address Settings will be displayed. The user must enter the correct base address. Otherwise, the default value (0) will be used. The base address is the address of the start of the flash module on the device.

3.1.3 HCS08

The user may configure the Cyclone PRO to operate on an HCS08 target by selecting HCS08 from the Specify Target Architecture drop-down menu.

Alternatively, the user may select the HCS08 target through the File menu.

Figure 3-5 shows the HCS08 specification configuration.

The target BDM header connections are shown on the right-hand side for user reference. The Cyclone PRO will automatically detect the communications speed.

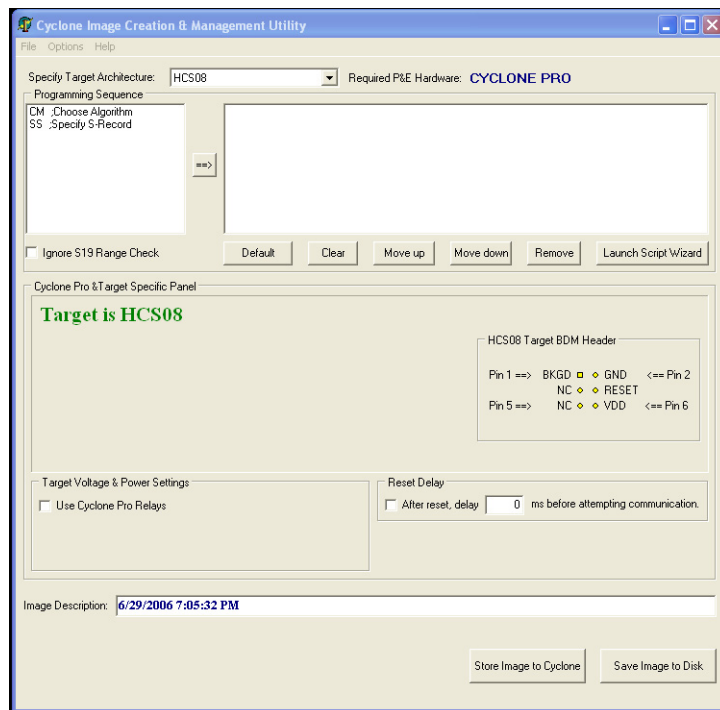


Figure 3-5: HCS08 Class Settings

3.1.4 RS08

The user may configure the Cyclone PRO to operate on an RS08 target by selecting RS08 from the Specify Target Architecture drop-down menu. The Cyclone PRO will automatically detect the target frequency and voltage. The Cyclone PRO provides all necessary voltage for programming.

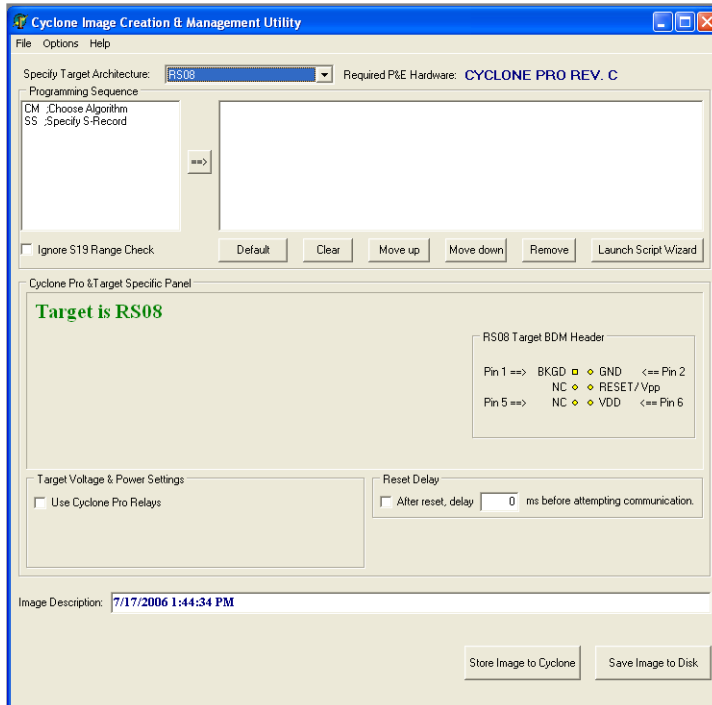


Figure 3-6: RS08 Class Settings

3.1.5 HC08

The user may configure the Cyclone PRO to operate on an HC08 target by selecting HC08 from the Specify Target Architecture drop-down menu.

Alternatively, the user may select the MON08 target through the File menu. Figure 5.6 shows the HC08 specification configuration.

The target MON08 header connections are shown on the right-hand side for

user reference.

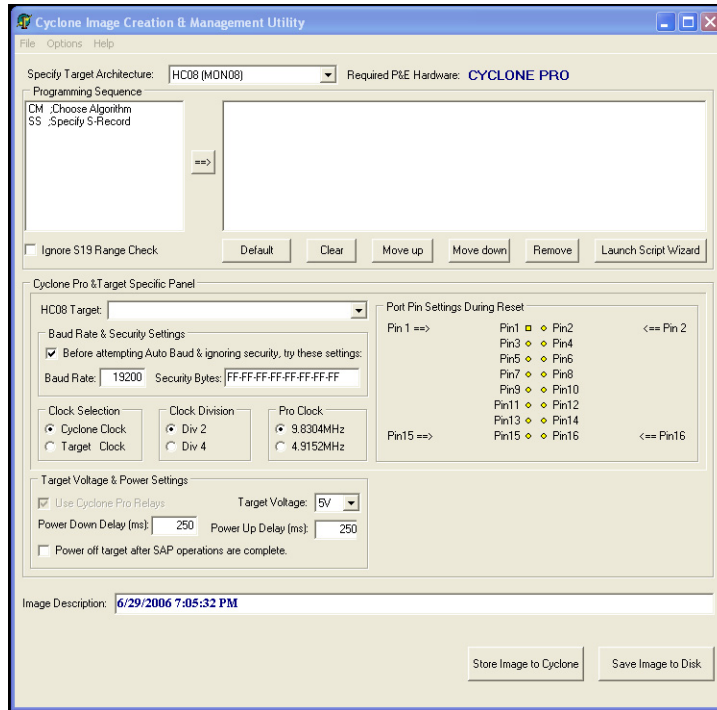


Figure 3-7: HC08 Class Settings

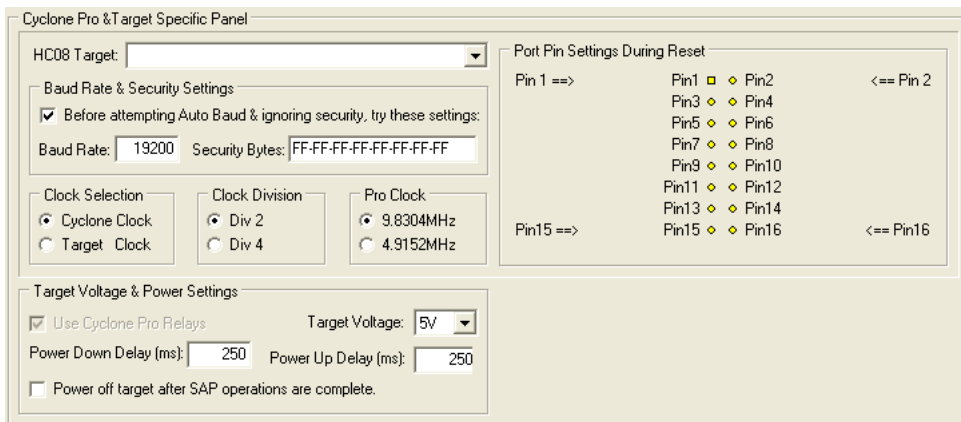


Figure 3-8: Port Pin Settings - 68HC908AB Family

3.1.5.1 HC08 Target

The user should choose the HC08 target that best describes the target MCU to be programmed. For example, choose “AB” for a 68HC908AB32 device, and choose “JB16” for a 68HC908JB16 device.

The MON08 Header connections are shown on the right hand side for user reference. Please refer to Chapter 4 for detailed information.

3.1.5.2 Clock Selection

The user may choose to use the “Cyclone Clock” to overdrive the target RC or Crystal circuitry. In this case either the 9.8304 MHz or 4.9152 MHz oscillator signal of the Cyclone PRO is connected to the target MON08 Header Pin 13 through the ribbon cable. The user should connect the target MCU OSC1 pin to Pin 13 of the MON08 Header.

If the user chooses to use the “Target Clock”, they must NOT connect anything to Pin 13 of the target MON08 Header. The 9.8304 MHz oscillator signal of the Cyclone PRO will be disabled.

3.1.5.3 Clock Division

The user may freely choose the clock division (if supported by the target) as long as the target bus frequency stays within specification. If the port

pin that determines the clock division is not connected to the target MON08 Header, the user can safely ignore this selection.

Typically the clock division does not affect the quality of communications between the Cyclone PRO and the target. Normally a smaller clock division leads to faster target bus frequencies, and faster communications between the Cyclone PRO and the target, which in turn leads to a shorter programming time.

3.1.5.4 Baud Rate and Security Settings

This configuration field provides faster monitor ROM entrance for known baud rate and security bytes.

If the “Before attempting auto-baud and ignoring security, try these settings” checkbox is checked, the Baud Rate and Security Bytes that are specified in this field are tried first when the Cyclone PRO contacts the target. If the target enters the Monitor Mode successfully, the Cyclone PRO will execute the instructed programming operations immediately.

If “Before attempting auto-baud and ignoring security, try these settings” is NOT checked, or if the target fails to enter the Monitor Mode using the specified baud rate and security bytes, the Cyclone PRO will automatically detect the communications baud rate, ignore the security, enter the Monitor Mode, erase the target device, power cycle the target, and re-enter the Monitor Mode using blank security bytes.

3.1.5.5 Target Voltage and Power Settings

The target voltage specifies the target MCU I/O voltage level.

The user needs to take into account the power discharge time for the Power Down delay. The reset driver delays, power stabilization time, and the target clock stabilization time should be considered for the Power Up delay.

3.2 Specify Programming Script

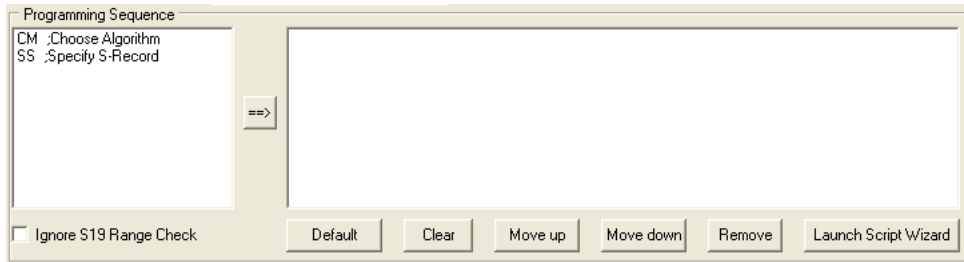


Figure 3-9: Specify Programming Script

This is a two-panel interface. The left panel provides a list of available programming functions. The right panel displays the ordering of the functions.

To specify the programming algorithm for the target, double-click on the Choose Algorithm (CM) function in the left panel. Or, you may highlight it and add it to the right panel using the arrow (->). This opens the Load Programming Algorithm dialog.

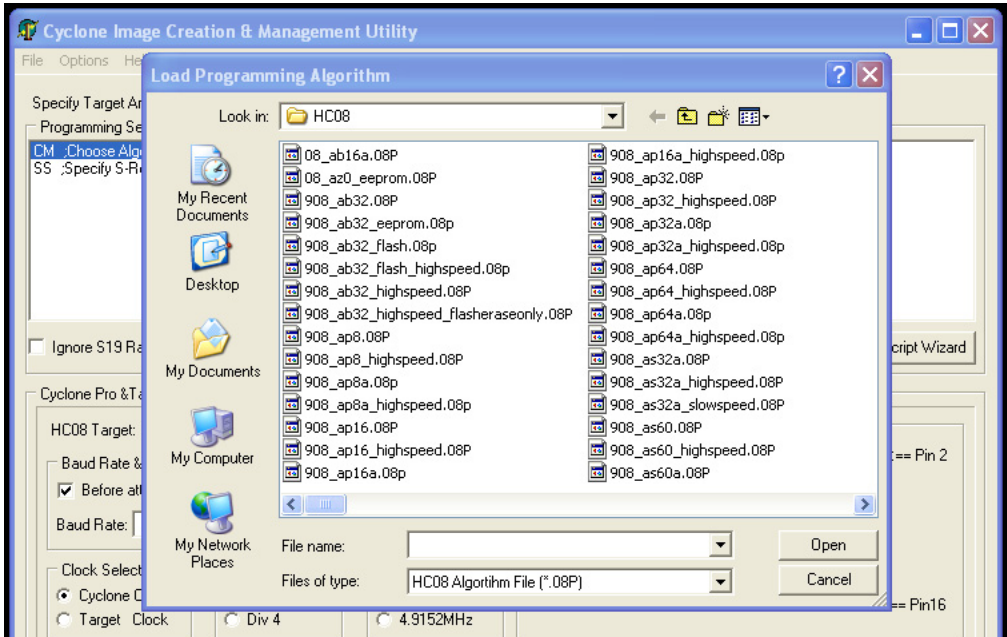


Figure 3-10: Load Programming Algorithm Dialog

Select the programming algorithm that you wish to use.

Similarly, to specify the S-Record to be programmed into the target, double-click on Specify S-Record (SS) in the left panel. This opens a dialog which allows you to select the appropriate S-Record.

Once both the algorithm and S-Record are selected, the full list of programming functions becomes available in the left panel.

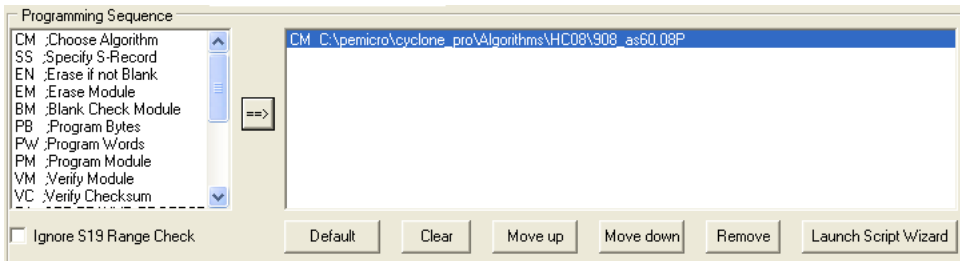


Figure 3-11: Programming Functions Enabled

Next, the user should add additional programming functions to complete the programming script.

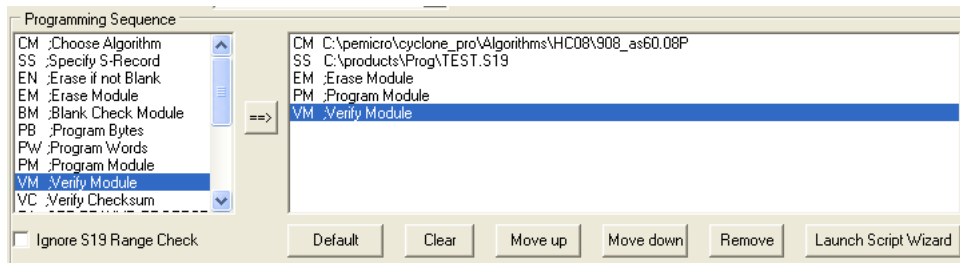


Figure 3-12: Programming Functions Complete

The Default button prompts the user for a programming module, followed by an S-Record, and creates a default programming script.

The Clear button will remove all programming commands from the right panel.

The Move Up and Move Down buttons allow the user to manually re-sequence the order of the programming commands.

The Remove button can be used to remove a selected command from the right panel.

For information about the Launch Script Wizard button, please reference **Section 3.3 - Script Wizard**.

At this point the image can be saved to a disk or to the Cyclone PRO device. For more information, please see **Section 3.6 - Store Image To Cyclone**.

3.3 Script Wizard

The Script Wizard is a convenient addition to the Cyclone Image Creation And Management Utility. In contrast to script creation method referenced in **Section 3.2 - Specify Programming Script**, the Script Wizard allows the user to simply select the programming functions. The Script Wizard then orders these functions automatically.

To begin, click the Launch Script Wizard button on the right side of the Cyclone Image Creation And Management Utility. This opens the Script Wizard.

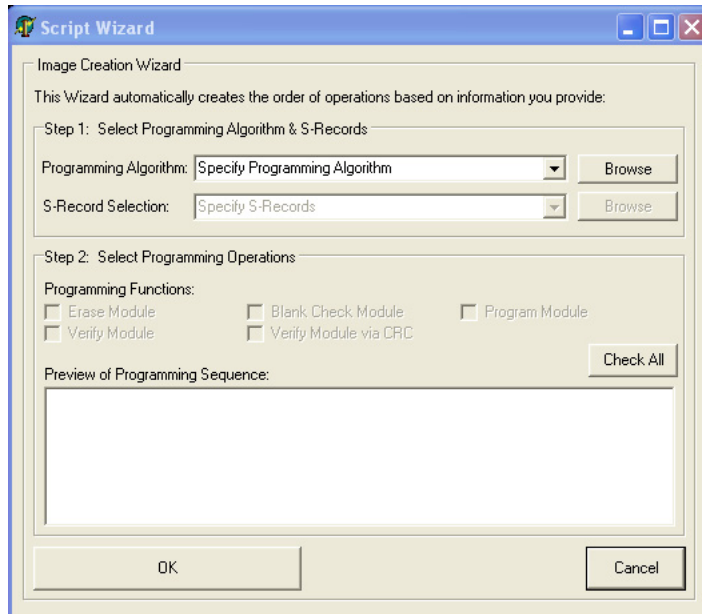


Figure 3-13: Script Wizard Dialog

The user must first select a programming algorithm and an S-Record. Click the corresponding Browse buttons to select a file, or use the drop-down list to choose a previously selected file.

Once this is complete, the programming functions are enabled. At this point the user has only to select the programming steps that are required, with no further attention to the order of operations.

For example, in **Figure 3-14**, three programming functions have been selected and sequenced by the Script Wizard.

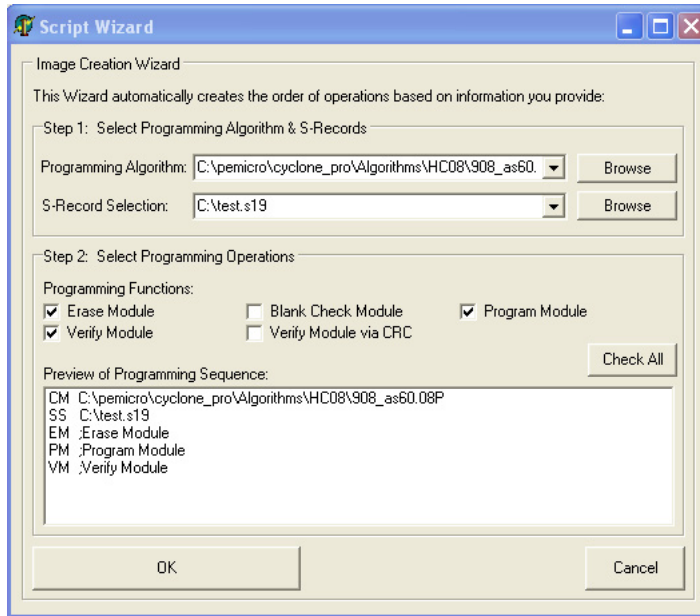


Figure 3-14: Script Wizard Dialog - Programming Functions Selected

Upon completion, click OK and the script information will appear in the Specify Programming Script dialog.

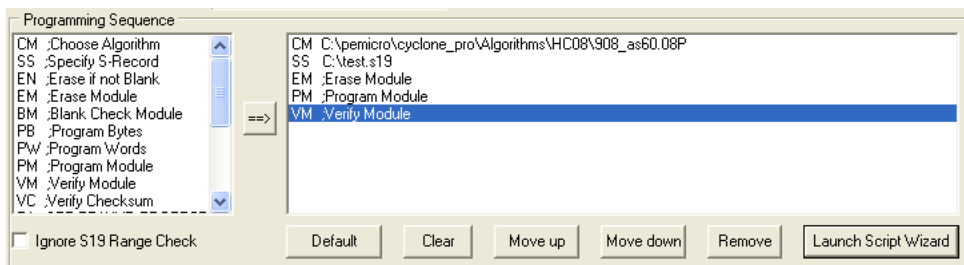


Figure 3-15: Results Of Script Wizard Dialog

3.4 Programming Operations

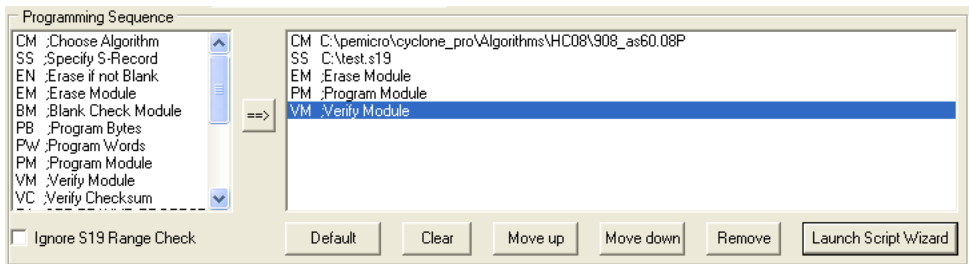


Figure 3-16: Programming Operations Dialog Section

In the Programming Sequence field, the user may specify the algorithm, S-Record, and operations to be carried out.

Choose Module

Presents a list of available programming files. Each programming file contains information on how to program a particular module. Usually, the name of the file indicates what kind of module it relates to.

Specify S-Record

Asks for the name (and/or path) to a file of Freescale S-records to be used in programming or verifying a module. If the file is not found, an error message is given. The currently-selected file is shown in the S19 file selected window. The programmer accepts S1, S2, and S3 records. All other file records are treated as comments. If you do not specify a file-name extension, a default of .S19 is used. The programmer also supports ELF/Dwarf 2.0 object files.

Your S19 file may contain data for both EEPROM and flash. If you know that your S19 file contains the correct data, “Ignore S19 Range” may be checked. This will cause any out of range errors to be ignored.

Erase If Not Blank

This command performs a blank check of the module and erases it if it is not blank.

Erase Module

If “Erase Module” is specified, the Cyclone PRO will perform an “Erase Module” on the target device after entering the Monitor Mode or BDM mode.

Blank Check Module

If “Blank Check Module” is checked, the Cyclone PRO will perform a “Blank Check Module” on the target device.

Program Bytes

Prompts for a starting address, which must be in the module. You are then asked to enter in hexadecimal a byte to be programmed into the current location. Clicking the OK button will automatically advance to the next data byte location.

Program Words

Prompts for a starting address, which must be in the module. You are then asked to enter in hexadecimal a word to be programmed into the current location. Clicking the OK button will automatically advance to the next data word location.

Program Module

For this command to work, you must have previously selected an S-record file.

Verify Module

For this command to work, you must have previously selected an S-record file.

Verify Checksum

This command verifies the module content via a CRC calculation.

3.5 Image Description

The Cyclone PRO Configuration Utility allows the user to summarize the purpose of current configuration for future reference. The description will be either programmed into the Cyclone PRO or saved into an encrypted file.

This field will not affect the Cyclone PRO’s operations with the target.

3.6 Store Image To Cyclone

“Store Image to Cyclone” allows the current configuration to be programmed

into the Cyclone PRO. The Cyclone PRO will then be ready for operations.

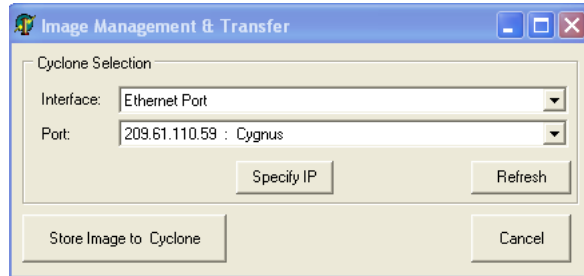


Figure 3-17: Image Management And Transfer Dialog

The Interface drop-down list allows the user to select one of three serial, USB, or ethernet interfaces. The Port drop-down list allows the user to select from one of the Cyclones available on that interface. In the case of a Cyclone present on a different network (i.e., not displayed automatically in the Port drop-down list), the user may specify its IP address by using the Specify IP button.

“Store Image to Cyclone” will then store the image on the selected Cyclone.

3.7 Save Image/Cyclone Configuration

“Store Image To Disk” allows the current configuration to be saved onto the hard drive. The image can then be transferred to the Cyclone PRO via the Manage Images Utility.

“Save Cyclone Configuration,” in the file menu, allows the user to save the configuration into a file, which may be used for future reference, e.g., comparing the Cyclone PRO contents with the file to see if they are the same.

3.8 Configuration Via LCD Menu

The following section describes configuration of the Cyclone PRO using the LCD menus. **Figure 3-18** shows an overview of the menu structure.

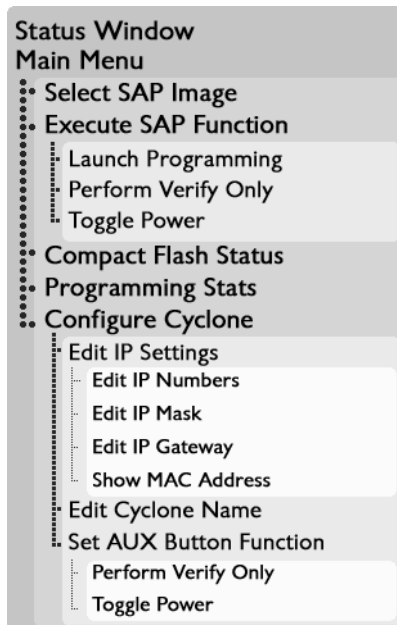


Figure 3-18: LCD Menu Overview

3.8.1 Status Window

```
Cyclone PRO v8.04-1.2
IP: 200.0.0.0
Name: Manufacturing
PC: none
Images: 2 (IN)/3 (EX)
IMG1: HC908GZ60
Idle...
Programming completed.
```

Figure 3-19: Status Window

The status window appears when the Cyclone PRO is powered on. This window lists the following information:

1. Firmware version of the PRO.
2. IP address assigned to the PRO.
3. Name assigned to the PRO.
4. Name of the PC connected to the PRO.
5. Number of programming images in the PRO's memory.
6. Name of the selected programming image.
7. Current status.
8. Results of the last operation performed.

3.8.2 Main Menu

```
Main Menu
-----
Select SAP Image
Execute SAP Function
Compact Flash Status
Programming Stats
Configure Cyclone
```

Figure 3-20: Main Menu

The Main Menu is accessible by pressing the “Menu” button when the status window is displayed. The Main Menu contains the following selections:

3.8.2.1 Select SAP Image

Select SAP Image brings up a display listing the images that are stored in the Cyclone PRO’s memory. You may select the appropriate image by using the Up/Down arrows to highlight it, and then pressing the “Select” button. The image name shown is the one specified in the Cyclone configuration utility when programming the image to the Cyclone.



```
IN1 : HC908GZ60
IN2 : MC9S08QE128
EX1 : HC912B32
EX2 : HC9S12DP256B
EX3 : RS08KA2
```

Figure 3-21: Select SAP Image

3.8.3 Execute SAP Function

The Execute SAP Function menu selection is discussed in **Section 4.2 - Operation Via LCD Menu (Rev. C Only)**. Please refer to that section for additional menu information.

3.8.4 Compact Flash Status

The Compact Flash Status menu selection is discussed in **Section 4.2 - Operation Via LCD Menu (Rev. C Only)**. Please refer to that section for additional menu information.

3.8.5 Programming Stats

The Programming Stats menu selection is discussed in **Section 4.2 - Operation Via LCD Menu (Rev. C Only)**. Please refer to that section for additional menu information.

3.8.6 Configure Cyclone

```

Configure Cyclone
-----
Edit IP Settings
Edit Cyclone Name
Set AUX Button Func
  
```

Figure 3-22: Configure Cyclone

Configure Cyclone brings up a submenu with three options from which to choose.

3.8.6.1 Configure Cyclone: Edit IP Settings

```

Edit IP Settings
-----
Edit IP Numbers
Edit IP Mask
Edit IP Gateway
Show MAC Address
  
```

Figure 3-23: Configure Cyclone: Edit IP Settings

Edit IP Settings brings up a submenu with four options from which to choose.

Edit IP Settings: Edit IP Numbers

Edit IP Numbers allows the user to set an IP number for the Cyclone PRO. The current IP number is displayed on the second line. Use the Up/Down buttons to scroll through the characters. To select a character, hit the Select button. When you are finished, scroll through the characters until you reach the -> (right-arrow) character. Selecting this character will complete the process. Press the Cancel button at any time to leave the IP

number as is and return to the Main Menu.

```
Edit IP Number
(200.0.0.0)
-----
0
Up/Down to scroll
Select to choose
Choose -> when done
```

Figure 3-24: Edit IP Settings: Edit IP Number

Edit IP Settings: Edit IP Mask

Edit IP Mask allows the user to set an IP Mask for the Cyclone PRO. The current IP Mask is displayed on the second line. Use the Up/Down buttons to scroll through the characters. To select a character, hit the Select button. When you are finished, scroll through the characters until you reach the -> (right-arrow) character. Selecting this character will complete the process. The default IP mask is 255.255.255.0.

```
Edit IP Mask
(0.0.0.0)
-----
0
Up/Down to scroll
Select to choose
Choose -> when done
```

Figure 3-25: Edit IP Settings: Edit IP Mask

Edit IP Settings: Edit IP Gateway

Edit IP Gateway allows the user to set the IP Gateway for the Cyclone PRO. The current IP Gateway is displayed on the second line. Use the Up/Down buttons to scroll through the characters. To select a character, hit the Select button. When you are finished, scroll through the characters until you reach the -> (right-arrow) character. Selecting this character will complete the

process.

```

Edit IP Gateway
(209.10.112.201)
-----
0
Up/Down to scroll
Select to choose
Choose -> when done
  
```

Figure 3-26: Edit IP Settings: Edit IP Gateway

Edit IP Settings: Show MAC Address

Show MAC Address displays the current MAC address for the Cyclone PRO.

```

Show MAC Address
-----
10-22-33-44-55-0
  
```

Figure 3-27: Edit IP Settings: Show MAC Address

3.8.6.2 Configure Cyclone: Edit Cyclone Name

Edit Cyclone Name allows the user to set the name for the Cyclone PRO. The current name is displayed on the second line. Use the Up/Down buttons to scroll through the characters. To select a character, hit the Select button. When you are finished, scroll through the characters until you reach the -> (right-arrow) character. Selecting this character will complete the process. Press the Cancel button at any point to leave the

name as is and return to the Main Menu.

```
Edit Cyclone Name  
(Manufacturing)  
-----  
A  
Up/down to scroll  
Select to choose  
Choose -> when done
```

Figure 3-28: Configure Cyclone: Edit Cyclone Name

3.8.6.3 Configure Cyclone: Set AUX Button Func

Set AUX Button Func allows the user to assign a function to the AUX button of the Cyclone PRO. Highlight the function you wish to assign to the AUX button and press the Select button to choose it.

```
Set AUX Button Func  
-----  
Perform Verify Only  
Toggle Power
```



Figure 3-29: Configure Cyclone: Set AUX Button Func

4 STAND-ALONE PROGRAMMER MANUAL CONTROL

The Cyclone PRO must be configured before it may serve as a Stand-Alone Programmer for HC08, HCS08, HC(S)12(X), ColdFire V1 targets. The user may manually control the Cyclone PRO: via the buttons/LEDs, LCD menu, or via PC software. The target power management schemes remain the same for each control method.

4.1 Operation Via Cyclone PRO Buttons

There are five (5) buttons on the top of the Cyclone PRO which are used for stand-alone programming and to navigate the LCD menus. They are specified as follows.

<u>Button</u>	<u>Function</u>
START / 	Start executing the tasks pre-configured into the Cyclone PRO. Menu Mode: Navigate upwards in LCD menu.
AUX / 	Perform auxilliary function (stand-alone verification). Menu Mode: Navigate downwards in LCD menu.
MENU / [SELECT]	Toggles ON/OFF the Target Board Power. Menu Mode: Select highlighted item in LCD menu.
CANCEL	Cancel the tasks being executed and go back to the standby state.
RESET	Hardware reset of the Cyclone PRO.

4.1.1 Cyclone PRO LED Indicators

The Cyclone PRO has eight (8) LEDs to indicate the current operation stage.

<u>LED</u>	<u>FUNCTION</u>
Power	Indicates that the target board power is connected.
Standby	The Cyclone PRO is waiting for instructions.
Security	The Cyclone PRO is trying to pass the target security and enter monitor mode (for MON08 only).
Erase	The Cyclone PRO is erasing the target Flash/EEPROM.
Program	The Cyclone PRO is programming the target Flash/

	EEPROM.
Verify	The Cyclone PRO is verifying that the contents programmed.
Error	The Cyclone PRO failed to execute the functions as instructed.
Success	The Cyclone PRO executed the functions successfully.

4.1.2 Procedure via Buttons and LEDs

The following steps must be followed in order for the Cyclone PRO to operate properly after the Cyclone PRO has been configured:

1. Turn off the target power supply if the “POWER IN” Jack is adopted.
2. Turn off the Cyclone PRO board power.
3. Set the correct Jumper settings.
4. Connect the target power supply to the “POWER IN” Jack, if applicable.
5. Connect the “POWER OUT” Jack to the target board power, if applicable.
6. Connect the MON08 Header Ribbon Cable to the target MON08 port, or the BDM Header Ribbon Cable to the target BDM port.
7. Turn on the Cyclone PRO board power.
8. Turn on the target power supply, if applicable.
9. Press the “START” push button on the Cyclone PRO. You will see the LEDs light up as specific functions are being executed.

When the “Success” LED lights up, you have successfully programmed your target.

4.1.3 Example

When the Cyclone PRO is powered up, the Standby LED is turned on. After the user programs the contents and procedures into the Cyclone PRO on-board flash, the Cyclone PRO may be used as a Stand-Alone Programmer. Suppose the user wants to perform the following instructions for a 68HC908 target:

- 1) Erase Module

- 2) Blank Check Module
- 3) Program Module
- 4) Verify Module.

When the Start Button is pressed, the “Target Power On” LED will turn on, indicating that the Cyclone PRO is powering up the target board.

Then the Standby LED will turn off and the Security LED will turn on. Here, if the target flash needs to be erased first to bypass the security, the Security LED will turn off and the Erasing LED will turn on.

When the Erasing LED turns off, the Cyclone PRO attempts to pass security again.

Then, when the Cyclone PRO starts programming the module, the “Programming” LED is illuminated.

After this is done, the “Programming” LED is turned off and the “Verifying” LED is turned on, designating that the Cyclone PRO is verifying the contents just programmed into the target.

Finally, if these operations have been performed successfully, the “Success” LED and the “Standby” LED are illuminated. One stand-alone programming cycle has just been completed.

4.2 Operation Via LCD Menu (Rev. C Only)

Rev. C of the Cyclone PRO may be operated by making selections from the LCD menu. This section describes the layout of the menus and the functions that each may be used to perform.

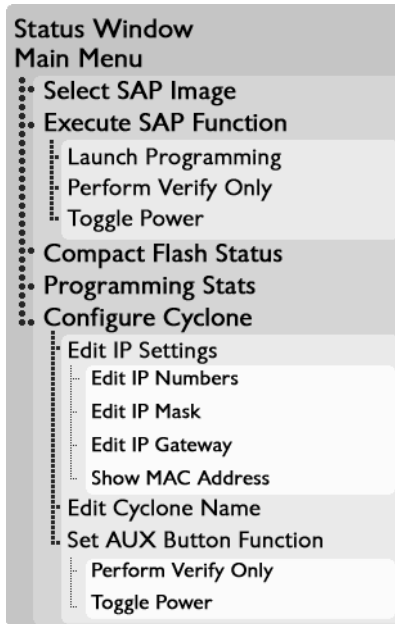


Figure 4-1: LCD Menu Overview

4.2.1 Status Window

```
Cyclone PRO v8.04-1.2
IP: 200.0.0.0
Name: Manufacturing
PC: none
Images: 2 (IN) / 3 (EX)
IMG1: HC908GZ60
Idle...
Programming completed.
```

Figure 4-2: Status Window

The status window appears when the Cyclone PRO is powered on. This window lists the following information:

1. Firmware version of the PRO.

2. IP address assigned to the PRO.
3. Name assigned to the PRO.
4. Name of the PC connected to the PRO.
5. Number of programming images in the PRO's memory.
6. Name of the selected programming image.
7. Current status.
8. Results of the last operation performed.

4.2.2 Main Menu

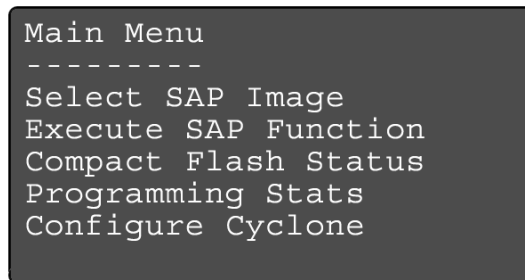


Figure 4-3: Main Menu

The Main Menu is accessible by pressing the “Menu” button when the status window is displayed. The Main Menu contains four selections. This section contains information on Execute SAP Function and Show Statistics. For information on Select SAP Image and Configure Cyclone, please refer to **Section 3.8 - Configuration Via LCD Menu**.

4.2.2.1 Execute SAP Function

Execute SAP Function presents three Stand-Alone Programming functions that you may execute by highlighting the function that you wish

to execute and pressing the “Select” button.

```
Execute SAP Function
-----
Launch Programming
Perform Verify Only
Toggle Power
```

Figure 4-4: Execute SAP Function

4.2.2.2 Compact Flash Status

If a compatible CompactFlash card is loaded into the Cyclone, the Compact Flash Status menu selection displays information about that CompactFlash card, including its name, the number of images it contains, and the amount of free, used, and total memory it contains.

```
Compact Flash Status
-----
Name: Production1
Number Of Images: 3
Memory: 128MB Total
32MB Used/96MB Free
```

Figure 4-5: Compact Flash Status

4.2.2.3 Show Statistics

The fourth line (PCIP:) displays the IP address of the last PC to control the Cyclone PRO. The other categories listed are for future use and are not currently implemented.

```
Show Statistics
-----
SerNum:
NumProgs:
ProgsLeft:
PCIP: 0.0.0.0
```

Figure 4-6: Show Statistics

5 STAND ALONE PROGRAMMER AUTOMATED CONTROL

There are various means of controlling the Cyclone PRO in an automated environment. Included with the Cyclone PRO are:

- 1) The CYCLONEPRO_LAUNCH utility, which allows control via script files.
- 2) The Basic Edition of P&E's Cyclone Automated Control SDK, which includes a .DLL which may be used by a host application to control the Cyclone PRO via Ethernet.

In addition, P&E offers the Professional and Enterprise editions of the Cyclone Automated Control SDK separately, which allow control of multiple Cyclones in parallel via serial, USB, or Ethernet. These editions also include support for and management of multiple programming images. Visit www.pemicro.com for more information about the Cyclone Automated Control SDK.

5.1 Automated .EXE Control Of The Cyclone

The user can use the Cyclone PRO in an automated production environment via PC software called CYCLONEPRO_LAUNCH.EXE. This utility may be invoked in a batch file to carry out pre-configured operations in the Cyclone PRO.

5.1.1 Startup

- a) Connect the Cyclone PRO to the PC via RS232, USB, or Ethernet.
- b) Connect the Cyclone PRO to the target system (Via the 6-pin background mode connector to the HC(S)12(X)/S08 targets, or the 16-pin MON08 connector to the HC908 targets).
- c) Power up the PC and the target system.
- d) Run the software from the DOS prompt. Allowed command line parameters are:

CYCLONEPRO_LAUNCH [port=y] [image=n]

where the value of y is: (See examples section)

USB1 - USB Device #1
USB2 - USB Device #2
USB3 - USB Device #3

USB4 - USB Device #4

COM1 - Serial Port 1

COM2 - Serial Port 2

COM3 - Serial Port 3

COM4 - Serial Port 4

- Ethernet IP address ####. Each # symbol represents a decimal number between 0 and 255.

- and where n designates the index of the image residing on the Cyclone PRO, ranging in value from 1 to 255.

5.1.2 Command-Line Parameter Examples

CYCLONEPRO_LAUNCH Port=COM1

Cyclone PRO is connected to the PC via the COM1 port.

CYCLONEPRO_LAUNCH Port=USB1

Cyclone PRO is connected to the PC via the USB1 port.

CYCLONEPRO_LAUNCH Port=USB1 Image=1

Start the first programming image on the Cyclone which is connected to USB1 port.

CYCLONEPRO_LAUNCH Port=209.61.110.251

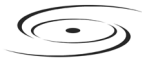
Cyclone PRO is connected to a network with IP address of 209.61.110.251

CYCLONEPRO_LAUNCH Port=209.61.110.251 Image=3

Start the third programming image on the Cyclone which is connected to the network with the IP address of 209.61.110.251.

5.1.3 DOS Error Returns

DOS error returns are provided so they may be tested in .BAT files. The error codes used are:



Application handling related error codes:

- 190: No command line parameters specified.
- 191: Specified COM port is not available.
- 192: Specified USB port is not available. Please make sure the USB port is available, and the USB cable is connected.
- 193: Specified Ethernet IP address is incorrect.
- 199: The Cyclone PRO device is not ready. Please check power and connections.

Execution related error codes:

- \$0001: Unable to detect target communication speed.
- \$0002: Unable to enter debug mode.
- \$0003: Operation cancelled by user.
- \$0004: Error writing data byte block.
- \$0005: Error writing data byte block.
- \$0006: Error during execution.
- \$0007: Error enabling module.
- \$0008: Error enabling module.
- \$0009: Error enabling module.
- \$000A: Error testing target timing.
- \$1001: Blank_Check_Word Algorithm is not supported.
- \$1002: Blank_Check_Byte Algorithm is not available.
- \$1003: Error during blank checking device using blank_check_byte algorithm.
- \$1004: Error during blank checking device using blank_check_word algorithm.
- \$2003: Error during erasing device.
- \$2004: Error during erasing device.
- \$3001: Program_Word Algorithm is not supported.
- \$3002: Program_Byte Algorithm is not available.
- \$3003: Error during programming device.
- \$5003: Error during verifying module.

- \$5004: Error during verifying range.
- \$6003: Error during user functions.
- \$6004: Error during user functions.
- \$7003: Error calculating trim value.
- \$7004: Trim value is not calculated.
- \$7005: Program Word Algorithm is not supported for writing the trim value.
- \$7006: Program Byte Algorithm is not available for writing the trim value.
- \$7007: Error during programming the trim value.

5.1.4 Sample Batch File

Here is an example of calling the CYCLONEPRO_LAUNCH utility and testing its error code return in a simple batch file. Sample batch files are given for both Windows NT/2000 and Windows 95/98/ME/XP.

Windows NT/2000:

```
CYCLONEPRO_LAUNCH PORT=COM1
if errorlevel 1 goto bad
goto good
:bad
ECHO BAD BAD BAD BAD BAD BAD BAD BAD
:good
ECHO done
```

Windows 95/98/ME/XP:

```
START /W CYCLONEPRO_LAUNCH PORT=USB1
if errorlevel 1 goto bad
goto good
:bad
ECHO BAD BAD BAD BAD BAD BAD BAD BAD
:good
ECHO done
```

5.1.5 Information

For more information on CYCLONEPRO_LAUNCH and how to construct detailed executions for your own devices, please contact P&E.

5.2 Automated .DLL Control Of The Cyclone

Another automated means of controlling a Cyclone PRO or Cyclone MAX unit is via a Windows .DLL. The .DLL allows a host application to connect to a Cyclone unit via Ethernet and control stand-alone programming operations.

Included with the Cyclone PRO is a .DLL which allows control of a Cyclone unit. Example code and documentation is included which demonstrates use of the .DLL with a Microsoft Visual C/C++ application and a Borland Delphi application, although the .DLL is not limited to these development environments. Any application which can load and use a .DLL can be used for automated control of a single Cyclone unit.

Automated .DLL Control Of Multiple Cyclones

For controlling multiple Cyclone units in parallel, the required .DLL is available in the “Cyclone Automated Control SDK.” Please visit www.pemicro.com for more information about this product.

5.2.1 Connecting/Disconnecting from the Cyclone

In order to control a Cyclone PRO or Cyclone MAX unit from a host computer, the host computer must initiate a connection to a Cyclone unit. The *connect_to_cyclonepromax_by_ip* call opens a connection to the Cyclone unit via Ethernet. The return handle is used by the calling application while controlling the Cyclone unit. The host PC should disconnect from the Cyclone unit when the programming process has completed.

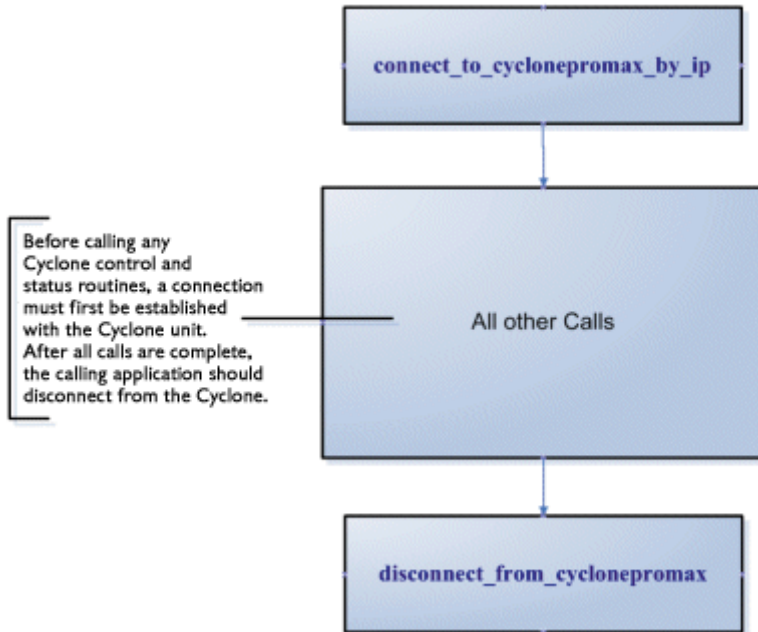


Figure 5-7: Call Sequence Detail

5.2.1.1 Call: `connect_to_cyclonepromax_by_ip`

C/C++ Prototype:

```
unsigned int connect_to_cyclonepromax_by_ip(char *port_identifier);
```

Description:

This function opens a session with a Cyclone PRO or Cyclone MAX by its IP address. The *port_identifier* parameter is a pointer to a null-terminated character string in the format xxx.xxx.xxx.xxx, where xxx = 0..255. The return value is the handle to the open Cyclone PRO or Cyclone MAX, which must be passed to other routines as a parameter. A return value of 0 indicates a failure to connect to the Cyclone PRO or Cyclone MAX at that address.

5.2.1.2 Call: `disconnect_from_cyclonepromax`

C/C++ Prototype:

```
bool disconnect_from_cyclonepromax(unsigned long cyclonepromaxhandle);
```

Description:

When processing is complete, the session with the Cyclone PRO or Cyclone MAX should be terminated. Calling this routine with the appropriate handle terminates the session and allows opening of a different Cyclone PRO/MAX unit.

5.2.2 Controlling Cyclone Programming

There are two main programming commands which start the programming process on a Cyclone.

The *START_execute_all_commands* call is used to launch a programming image which is stored on the Cyclone unit and has been pre-configured in terms of all the steps to be accomplished during programming. Once started, the *check_STARTED_cyclonepromax_status* routine may be called until it indicates that the Cyclone programming has completed. At this point the *get_last_error_code* routine should be called to determine if an error has occurred or if the programming was successful.

Another command, *START_dynamic_program_bytes*, allows data which has not been pre-configured into the Cyclone unit to be programmed to the target. This is useful for programming the date, a serial number, MAC address, or other dynamically generated data to the target. The *check_STARTED_cyclonepromax_status* routine is used to wait for programming completion.

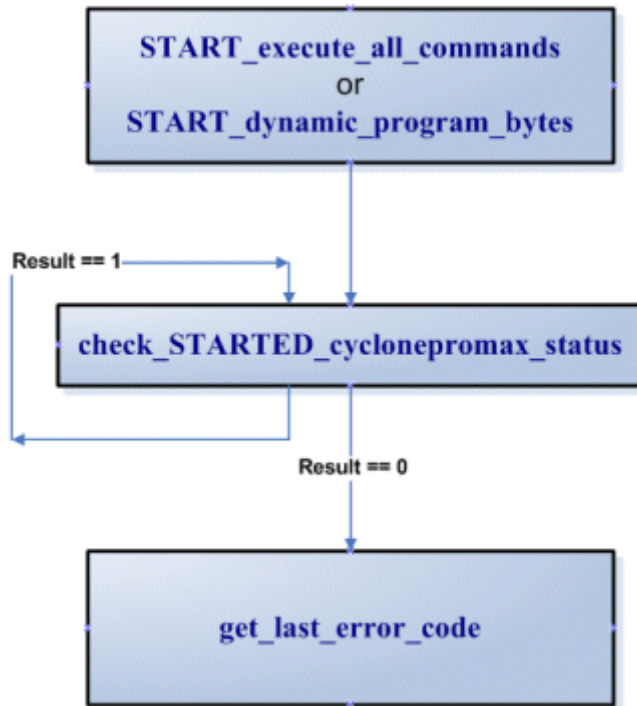


Figure 5-8: Call Sequence Detail

5.2.2.1 Call: START_execute_all_commands

C/C++ Prototype:

```
bool START_execute_all_commands(unsigned long cyclonepromaxhandle,
unsigned long imagenumber);
```

Description:

A Cyclone unit may have several independent programming images in non-volatile memory. A programming image contains the programming algorithms, binary data, and programming sequence. Calling this routine instructs the Cyclone unit to start execution of a particular image. If the Cyclone unit only holds one image, the *imagenumber* should be set to a “1”. If the Cyclone unit

holds multiple programming images, the valid values for image number range from one up to the number of images in the unit. A result of true indicates that the image has started the programming process. Call the *Check_STARTED_CycloneProMax_Status* command to wait for completion.

5.2.2.2 Call: **START_dynamic_program_bytes**

C/C++ Prototype:

```
bool START_dynamic_program_bytes(unsigned long cyclonepromaxhandle,
unsigned long target_address, unsigned short data_length, char *buffer);
```

Description:

Sometimes, in addition to the large amount of static data being programmed into a target from the Cyclone, it will be advantageous for the calling application to program small sections of unique data dynamically. Examples of this include: date/time, serial number, MAC addresses, and lot numbers. This routine allows the user to specify an address, data length, and pointer to a buffer of data to be programmed into the target. It is valid to be called only after a programming image has been programmed into the target (once *START_execute_all_commands* has completed). A result of true indicates that the image has started the programming process. Call the *Check_STARTED_CycloneProMax_Status* command to wait for completion.

5.2.2.3 Call: **check_STARTED_cyclonepromax_status**

C/C++ Prototype:

```
unsigned int check_STARTED_cyclonepromax_status(unsigned long
cyclonepromaxhandle);
```

Description:

Checks to see if the Cyclone PRO or Cyclone MAX has completed a programming operation started with either the *START_execute_all_commands* or *START_dynamic_program_bytes* routines. The results are:

- 1 = Currently Programming
- 0 = Completed (with or without error)

After this call returns the completed value, *get_last_error_code* should be

called to determine the programming result.

5.2.2.4 Call: `get_last_error_code`

C/C++ Prototype:

```
unsigned short get_last_error_code(unsigned long cyclonepromaxhandle);
```

Description:

Returns the last error code of the Cyclone PRO or Cyclone MAX.

0 = No Error.

5.2.2.5 Call: `get_last_error_addr`

C/C++ Prototype:

```
unsigned int get_last_error_addr(unsigned long cyclonepromaxhandle);
```

Description:

If the *get_last_error_code* call returned a non-zero value (i.e., there was an error), this routine can be used to query the address where the error occurred.

5.2.3 Configuration / Image Maintenance

These calls are used to perform operations on the Cyclone units such as: reset, check image, erase image, and add new image to the Cyclone. Due to security concerns, the calling application will not read or upload images from the Cyclone unit. All images which are to be verified or downloaded onto the Cyclone must have been generated by the Cyclone Configuration utility.

5.2.3.1 Call: `reset_cyclonepromax`

C/C++ Prototype:

```
bool reset_cyclonepromax(unsigned long cyclonepromaxhandle, unsigned long reset_delay_in_ms);
```

Description:

Used to reset the Cyclone PRO/MAX hardware. The delay should be at least 2500ms. This routine does not need to be called by the calling application and

exists for historical reasons.

5.2.3.2 Call: `get_image_description`

C/C++ Prototype:

```
char *get_image_description(unsigned long cyclonepromaxhandle, unsigned long imagenumber);
```

Description:

Returns the description string of the selected image. For a Cyclone unit with only one programming image, the *imagenumber* value should be set to 1. If the Cyclone unit holds multiple programming images, the valid values for image number range from one up to the number of images in the unit. The result is a pointer to a null-terminated string containing the image description. This description is valid until the next call into the .DLL is made.

5.2.3.3 Call: `compare_image_with_file`

C/C++ Prototype:

```
bool compare_image_with_file(unsigned long cyclonepromaxhandle, char *aFile, unsigned long imagenumber);
```

Description:

Allows the calling application to compare a stand-alone programming image stored on the PC with an image stored in the Cyclone. This can be used to validate that a programming image in the PC is exactly the same as the one on the Cyclone. The Cyclone images automatically self-verify, but this may be useful to make sure that the version of the image in the Cyclone is what the calling application is expecting. For a Cyclone unit with only one programming image, the *imagenumber* value should be set to 1. If the Cyclone unit holds multiple programming images, the valid values for image number range from one up to the number of images in the unit. A “true” result indicates that the images are the same. The *aFile* pointer should point to a null terminated filename, which is the full pathname of the file.

5.2.3.4 Call: `erase_all_cyclone_images`

C/C++ Prototype:

```
bool erase_all_cyclone_images(unsigned long cyclonepromaxhandle);
```

Description:

This call erases all stand-alone programming images on the Cyclone unit. Generally this call is made before a new image is placed onto the Cyclone by the `add_image_to_cyclone` call. A “true” result indicates that the erasure was successful.

5.2.3.5 Call: `add_image_to_cyclone`

C/C++ Prototype:

```
unsigned int add_image_to_cyclone(unsigned long cyclonepromaxhandle,  
char *aFile);
```

Description:

This call adds the stand-alone programming image specified with the *aFile* parameter into the Cyclone’s onboard memory. This image must have been created with the Cyclone configuration utility. This call is useful for adding more images into the Cyclone. The Cyclone has limited onboard memory, so the user may wish to erase existing images with the `erase_all_cyclone_images` command before calling this command. The command will return the image number that was just added, or else return a “0” if the operation failed.

5.2.3.6 Call: `toggle_power_no_background_entrance`

C/C++ Prototype:

```
bool toggle_power_no_background_entrance(unsigned long  
cyclonepromaxhandle);
```

Description:

Causes the RESET line of the target processor to be driven low, then pulled up high to enter normal operation mode.

5.2.3.7 Call: `set_local_machine_ip_number`

C/C++ Prototype:

```
void set_local_machine_ip_number(char *ip_number);
```

Description:

If a machine has more than one network card, this routine may be called to indicate the IP address of the network card which should be used during communications. This should be called prior to calling any other routines.

5.2.4 DLL Information, Interface Code, and Examples

The documented interface routines are available in the CYCLONEPROMAX_SAP_CONTROL.DLL library included in the software install. This .DLL is callable from almost any 32-bit Windows development environment. The method of calling the .DLL varies for different compilers, so .DLL interface code and sample applications are provided for each of the following compilers:

Microsoft Visual C++ 5.0+ - Visual MFC Application

Borland Delphi 4.0+ (Pascal) - Visual Application

The sample applications come with project and workspaces defined for ease of use. Just open the project/workspace in your compiler and you will be able to build the sample application without any modifications.

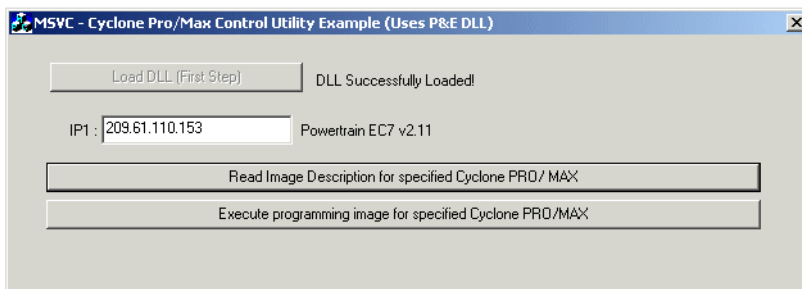


Figure 5-9: Cyclone PRO/MAX Control Utility Example

The sample applications come pre-compiled with ICONS, so you can run

them before jumping into the code. The callable interface routines are defined in:

```
INSTALLDIR\msvc50\cyclonepromax_sap_control.h    - For MSVC 5.0+
INSTALLDIR\Delphi20\cyclonepromax_sap_control.pas - For Delphi 4.0+
```

If you add the above-mentioned interface file (and associated .CPP file for the C++ compilers) to your own application, you will be able to directly access all the Cyclone control routines. Remember that the CYCLONEPROMAX_SAP_CONTROL.DLL file has to be in the executable's directory or system directory.

5.3 Control Of Multiple Cyclone Units

The utilities included with the Cyclone allow the unit to be updated and controlled in an automated fashion. The user can change the image in the Cyclone and execute the programming images resident on the unit under control from a host computer, or without a computer at all. These utilities/drivers focus on controlling a single Cyclone unit.

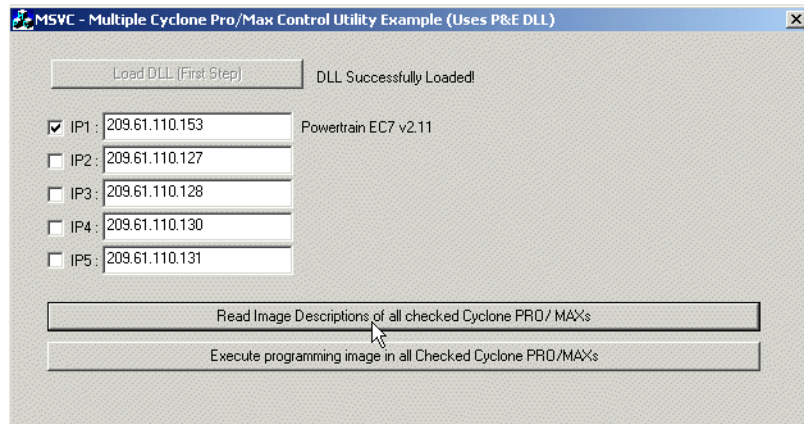
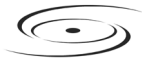


Figure 5-10: Multiple Cyclone PRO/MAX Control Utility Example

However, it may also be desirable to control multiple Cyclone units simultaneously from a host PC. Since the Cyclone is a self-contained computer system, including all programming algorithms, data, and processing



power necessary for programming, the host PC only needs to provide minimal control information. This means that an enormous number of Cyclone units may be controlled from a single PC. P&E offers a software development kit, the Multi-Cyclone Automated Control SDK, which is designed to provide this control information. This kit consists of the following components:

1. A .DLL which allows control of up to 1000 or more Cyclone units in parallel.
2. An .EXE which allows control of up to 1000 or more Cyclone units in parallel.
3. A UDP packet specification for controlling the Cyclone via Ethernet packets.
4. A serial packet specification for controlling the Cyclone via RS232 packets.

This SDK (software development kit) supports both the Cyclone PRO and the Cyclone MAX programmers, meaning that any supported Freescale devices can be programmed in parallel, even if they are different devices with different data. More information may be found at www.pemicro.com.

6 TARGET MON08 HEADER PINOUTS

This chapter details the MON08 connector signals according to the individual target MCU types.

6.1 68HC908AB

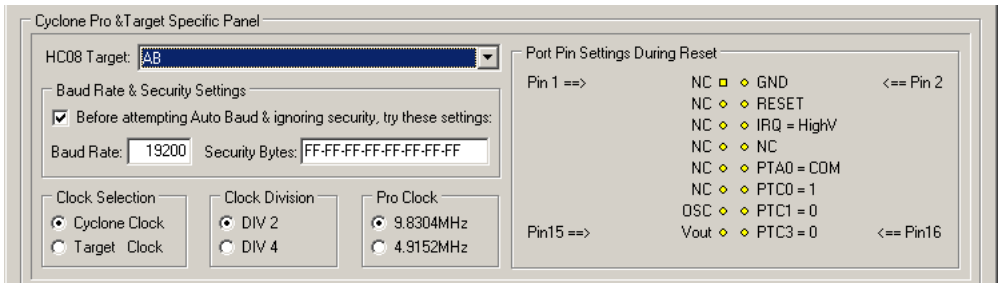


Figure 6-1: 68HC908AB Family MON08 Pinout

The target GND is connected to the Pin 2 of the target MON08 Header.

The target RESET line is directly connected to the Pin 4 of the target MON08 Header.

The target IRQ line is directly connected to the Pin 6 of the target MON08 Header.

PORTA0 from the target processor is connected to the target MON08 Header Pin 10, acting as the communications line. The Cyclone PRO pulls up this signal with a 10K Ohm resistor to the target VDD.

PORTC0, PORTC1 and PORTC3 are used for entering monitor mode. By default the user may directly bring these signals out to the target MON08 Header.

Alternatively, the user may pull up PORTC0 and pull down PORTC1, and pull up/down PORTC3 for clock division. In which case the user does not need to connect these signals to the target MON08 Header.

6.2 68HC908AP

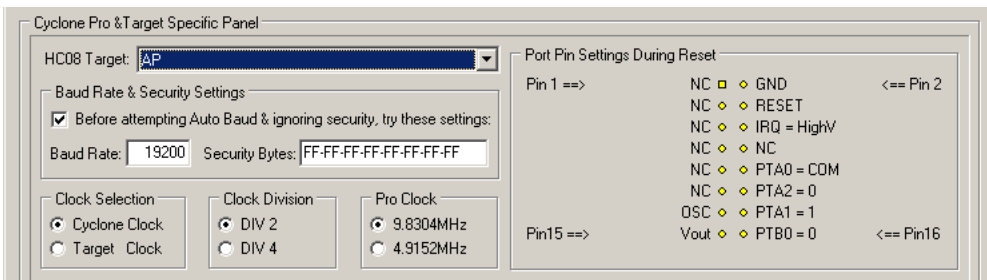


Figure 6-2: 68HC908AP Family MON08 Pinout

The target GND is connected to the Pin 2 of the target MON08 Header.

The target RESET line is directly connected to the Pin 4 of the target MON08 Header.

The target IRQ line is directly connected to the Pin 6 of the target MON08 Header.

PORTA0 from the target processor is connected to the target MON08 Header Pin 10, acting as the communications line. The Cyclone PRO pulls up this signal with a 10K Ohm resistor to the target VDD.

PORTA2, PORTA1 and PORTB0 are used for entering monitor mode. By default the user may bring these signals out to the target MON08 Header.

Alternatively, the user may pull down PORTA2 and pull up PORTA1, and pull up/down PORTB0 for clock division. In which case the user does not need to connect these signals to the target MON08 Header.

6.3 68HC908AS

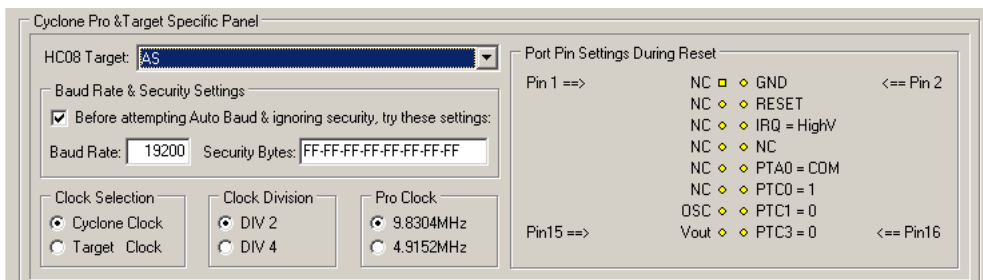


Figure 6-3: 68HC908AS Family MON08 Pinout

The target GND is connected to the Pin 2 of the target MON08 Header.

The target RESET line is directly connected to the Pin 4 of the target MON08 Header.

The target IRQ line is directly connected to the Pin 6 of the target MON08 Header.

PORTA0 from the target processor is connected to the target MON08 Header Pin 10, acting as the communications line. The Cyclone PRO pulls up this signal with a 10K Ohm resistor to the target VDD.

PORTC0, PORTC1 and PORTC3 are used for entering monitor mode. By default the user may directly bring these signals out to the target MON08 Header.

Alternatively, the user may pull up PORTC0 and pull down PORTC1, and pull up/down PORTC3 for clock division. In which case the user does not need to connect these signals to the target MON08 Header.

6.4 68HC908AT

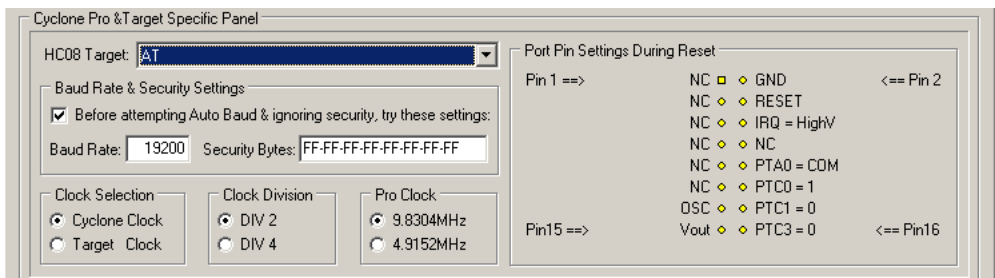


Figure 6-4: 68HC908AT Family MON08 Pinout

The target GND is connected to the Pin 2 of the target MON08 Header.

The target RESET line is directly connected to the Pin 4 of the target MON08 Header.

The target IRQ line is directly connected to the Pin 6 of the target MON08 Header.

PORTA0 from the target processor is connected to the target MON08 Header Pin 10, acting as the communications line. The Cyclone PRO pulls up this

signal with a 10K Ohm resistor to the target VDD.

PORTC0, PORTC1 and PORTC3 are used for entering monitor mode. By default the user may directly bring these signals out to the target MON08 Header.

Alternatively, the user may pull up PORTC0 and pull down PORTC1, and pull up/down PORTC3 for clock division. In which case the user does not need to connect these signals to the target MON08 Header.

6.5 68HC908AZ

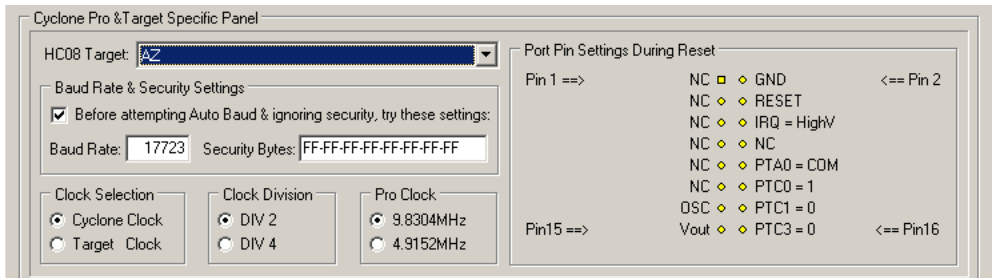


Figure 6-5: 68HC908AZ Family MON08 Pinout

The target GND is connected to the Pin 2 of the target MON08 Header.

The target RESET line is directly connected to the Pin 4 of the target MON08 Header.

The target IRQ line is directly connected to the Pin 6 of the target MON08 Header.

PORTA0 from the target processor is connected to the target MON08 Header Pin 10, acting as the communications line. The Cyclone PRO pulls up this signal with a 10K Ohm resistor to the target VDD.

PORTC0, PORTC1 and PORTC3 are used for entering monitor mode. By default the user may directly bring these signals out to the target MON08 Header.

Alternatively, the user may pull up PORTC0 and pull down PORTC1, and pull up/down PORTC3 for clock division. In which case the user does not need to connect these signals to the target MON08 Header.

6.6 68HC908BD

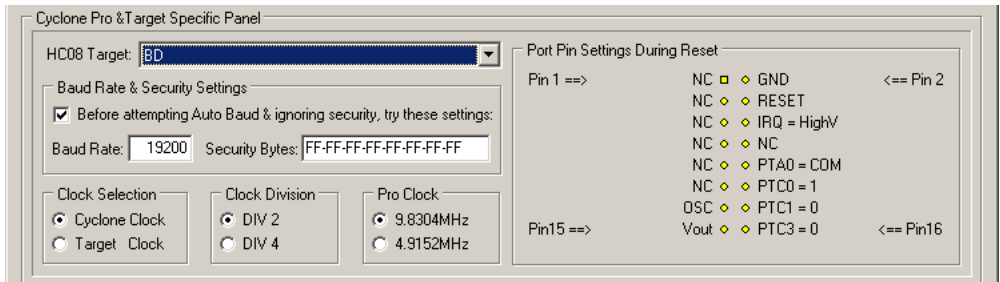


Figure 6-6: 68HC908BD Family MON08 Pinout

The target GND is connected to the Pin 2 of the target MON08 Header.

The target RESET line is directly connected to the Pin 4 of the target MON08 Header.

The target IRQ line is directly connected to the Pin 6 of the target MON08 Header.

PORTA0 from the target processor is connected to the target MON08 Header Pin 10, acting as the communications line. The Cyclone PRO pulls up this signal with a 10K Ohm resistor to the target VDD.

PORTC0, PORTC1 and PORTC3 are used for entering monitor mode. By default the user may directly bring these signals out to the target MON08 Header.

Alternatively, the user may pull up PORTC0 and pull down PORTC1, and pull up/down PORTC3 for clock division. In which case the user does not need to connect these signals to the target MON08 Header.

6.7 68HC908EY

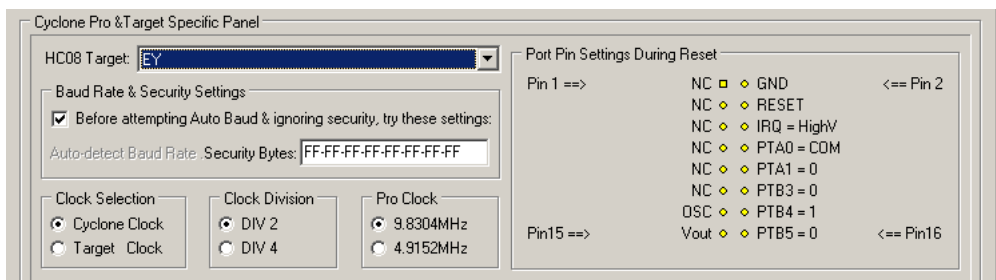


Figure 6-7: 68HC908EY Family MON08 Pinout

The target GND is connected to the Pin 2 of the target MON08 Header.

The target RESET line is directly connected to the Pin 4 of the target MON08 Header.

The target IRQ line is directly connected to the Pin 6 of the target MON08 Header.

PORTA0 from the target processor is connected to the target MON08 Header Pin 10, acting as the communications line. The Cyclone PRO pulls up this signal with a 10K Ohm resistor to the target VDD.

PORTA1, PORTB3, PORTB4 and PORTB5 are used for entering monitor mode. By default the user may directly bring these signals out to the target MON08 Header.

An external clock signal must be provided to the target MCU's OSC pin. The Cyclone PRO provides this signal to MON08 header Pin 13.

Alternatively, the user may pull down PORTA1 and PORTB3, and pull up PORTB4, and pull up/down PORTB5 for clock division. In which case the user does not need to connect these signals to the target MON08 Header.

6.8 68HC908GP

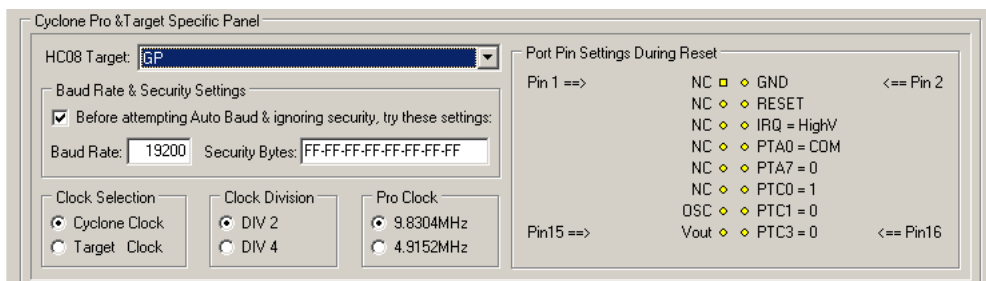


Figure 6-8: 68HC908GP Family MON08 Pinout

The target GND is connected to the Pin 2 of the target MON08 Header.

The target RESET line is directly connected to the Pin 4 of the target MON08 Header.

The target IRQ line is directly connected to the Pin 6 of the target MON08 Header.

PORTA0 from the target processor is connected to the target MON08 Header Pin 8, acting as the communications line. The Cyclone PRO pulls up this signal with a 10K Ohm resistor to the target VDD.

PORTA7, PORTC0, PORTC1 and PORTC3 are used for entering monitor mode. By default the user may directly bring these signals out to the target MON08 Header.

Alternatively, the user may pull down PORTA7 and PORTC1, pull up PORTC0, and pull up/down PORTC3 for clock division. In which case the user does not need to connect these signals to the target MON08 Header.

6.9 68HC908GR16/32

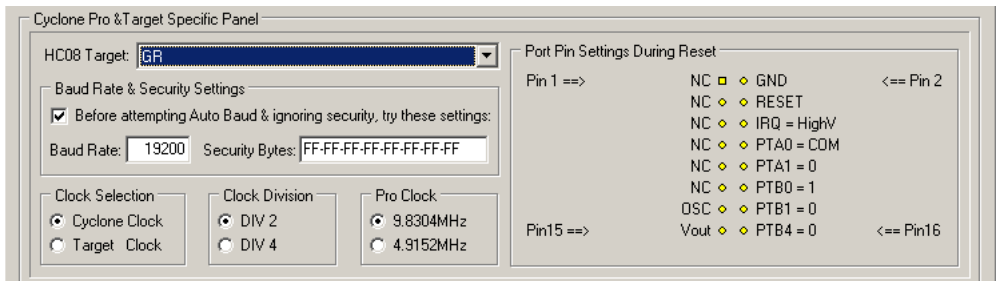


Figure 6-9: 68HC908GR16/32 MON08 Pinout

The target GND is connected to the Pin 2 of the target MON08 Header.

The target RESET line is directly connected to the Pin 4 of the target MON08 Header.

The target IRQ line is directly connected to the Pin 6 of the target MON08 Header.

PORTA0 from the target processor is connected to the target MON08 Header Pin 8, acting as the communications line. The Cyclone PRO pulls up this signal with a 10K Ohm resistor to the target VDD.

PORTA1, PORTB0, PORTB1 and PORTB4 are used for entering monitor mode. By default the user may directly bring these signals out to the target MON08 Header.

Alternatively, the user may pull up PORTB0, pull down PORTA1 and PORTB1, and pull up/down PORTB4 for clock division. In which case the user does not need to connect these signals to the target MON08 Header.

6.10 68HC908GR4/8

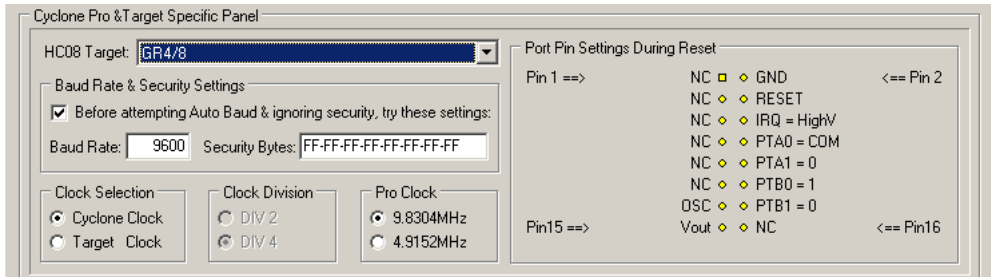


Figure 6-10: 68HC908GR4/8 MON08 Pinout

The target GND is connected to the Pin 2 of the target MON08 Header.

The target RESET line is directly connected to the Pin 4 of the target MON08 Header.

The target IRQ line is directly connected to the Pin 6 of the target MON08 Header.

PORTA0 from the target processor is connected to the target MON08 Header Pin 8, acting as the communications line. The Cyclone PRO pulls up this signal with a 10K Ohm resistor to the target VDD.

PORTA1, PORTB0, and PORTB1 are used for entering monitor mode. By default the user may directly bring these signals out to the target MON08 Header.

Alternatively, the user may pull up PORTB0, pull down PORTA1 and PORTB1. In which case the user does not need to connect these signals to the target MON08 Header. The clock division is fixed Div 4.

6.11 68HC908GT

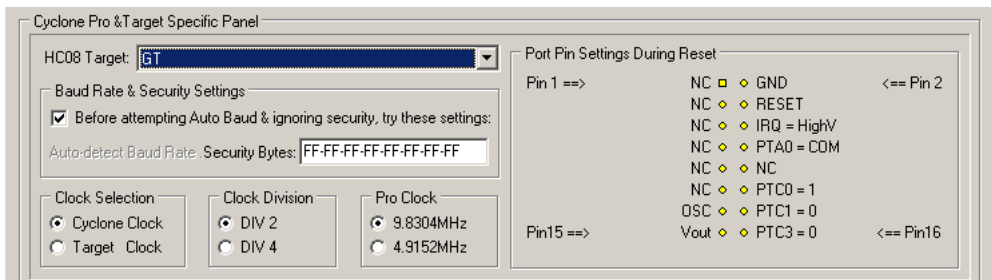


Figure 6-11: 68HC908GT Family MON08 Pinout

The target GND is connected to the Pin 2 of the target MON08 Header.

The target RESET line is directly connected to the Pin 4 of the target MON08 Header.

The target IRQ line is directly connected to the Pin 6 of the target MON08 Header.

PORTA0 from the target processor is connected to the target MON08 Header Pin 8, acting as the communications line. The Cyclone PRO pulls up this signal with a 10K Ohm resistor to the target VDD.

PORTC0, PORTC1 and PORTC3 are used for entering monitor mode. By default the user may directly bring these signals out to the target MON08 Header.

Alternatively, the user may pull up PORTC0 and pull down PORTC1, and pull up/down PORTC3 for clock division. In which case the user does not need to connect these signals to the target MON08 Header.

6.12 68HC908GZ

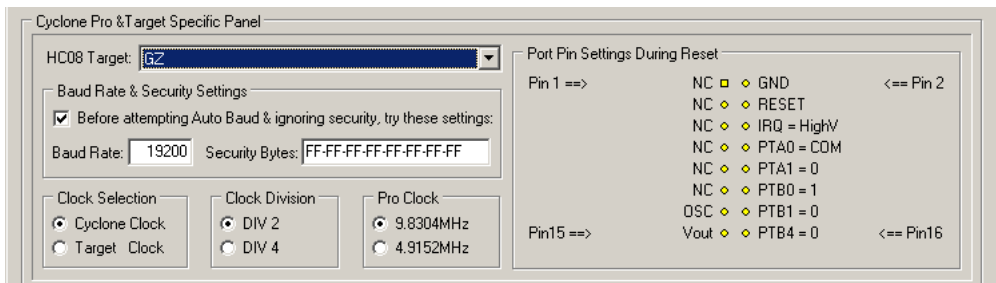


Figure 6-12: 68HC908GZ Family MON08 Pinout

The target GND is connected to the Pin 2 of the target MON08 Header.

The target RESET line is directly connected to the Pin 4 of the target MON08 Header.

The target IRQ line is directly connected to the Pin 6 of the target MON08 Header.

PORTA0 from the target processor is connected to the target MON08 Header Pin 8, acting as the communications line. The Cyclone PRO pulls up this

signal with a 10K Ohm resistor to the target VDD.

PORTA1, PORTB0, PORTB1 and PORTB4 are used for entering monitor mode. By default the user may directly bring these signals out to the target MON08 Header.

Alternatively, the user may pull up PORTB0, pull down PORTA1 and PORTB1, and pull up/down PORTB4 for clock division. In which case the user does not need to connect these signals to the target MON08 Header.

6.13 68HC908JB1/8

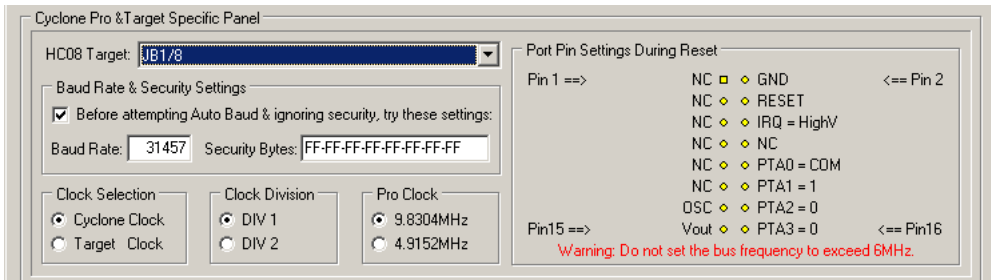


Figure 6-13: 68HC908JB1/8 MON08 Pinout

The target GND is connected to the Pin 2 of the target MON08 Header.

The target RESET line is directly connected to the Pin 4 of the target MON08 Header.

The target IRQ line is directly connected to the Pin 6 of the target MON08 Header.

PORTA0 from the target processor is connected to the target MON08 Header Pin 10, acting as the communications line. The Cyclone PRO pulls up this signal with a 10K Ohm resistor to the target VDD.

PORTA1, PORTA2 and PORTA3 are used for entering monitor mode. By default the user may directly bring these signals out to the target MON08 Header.

Alternatively, the user may pull up PORTA1 and pull down PORTA2, and pull up/down PORTA3 for clock division. In which case the user does not need to connect these signals to the target MON08 Header.

6.14 68HC908JB12

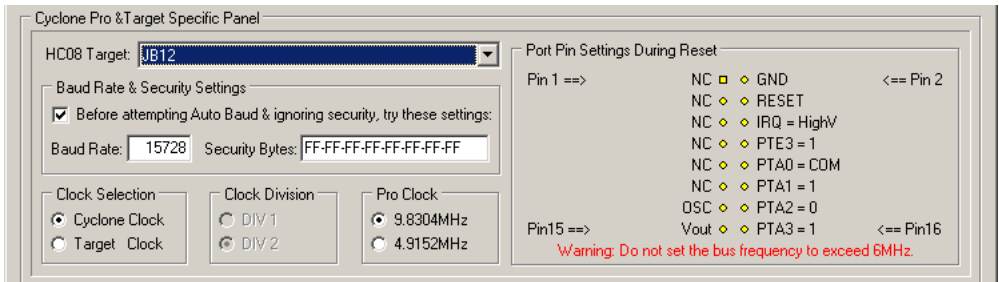


Figure 6-14: 68HC908JB12 MON08 Pinout

The target GND is connected to the Pin 2 of the target MON08 Header.

The target RESET line is directly connected to the Pin 4 of the target MON08 Header.

The target IRQ line is directly connected to the Pin 6 of the target MON08 Header.

PORTA0 from the target processor is connected to the target MON08 Header Pin 10, acting as the communications line. The Cyclone PRO pulls up this signal with a 10K Ohm resistor to the target VDD.

PORTA1, PORTA2, PORTA3 and PORTE3 are used for entering monitor mode. By default the user may directly bring these signals out to the target MON08 Header.

Alternatively, the user may pull up PORTA1 and PORTE3, pull down PORTA2, and pull up/down PORTA3 for clock division. In which case the user does not need to connect these signals to the target MON08 Header.

6.15 68HC908JB16

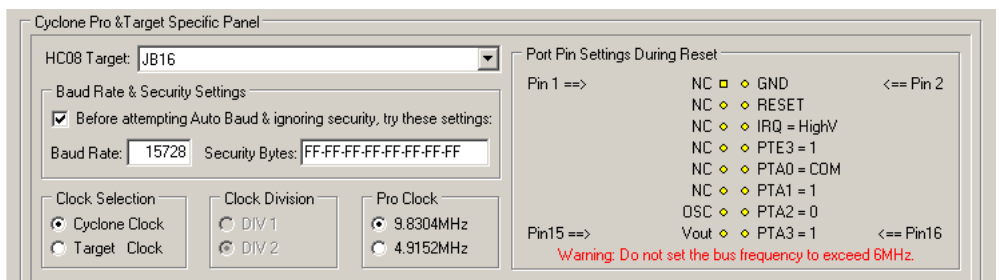


Figure 6-15: 68HC908JB16 MON08 Pinout

The target GND is connected to the Pin 2 of the target MON08 Header.

The target RESET line is directly connected to the Pin 4 of the target MON08 Header.

The target IRQ line is directly connected to the Pin 6 of the target MON08 Header.

PORTA0 from the target processor is connected to the target MON08 Header Pin 10, acting as the communications line. The Cyclone PRO pulls up this signal with a 10K Ohm resistor to the target VDD.

PORTA1, PORTA2, PORTA3 and PORTE3 are used for entering monitor mode. By default the user may directly bring these signals out to the target MON08 Header.

Alternatively, the user may pull up PORTA1 and PORTE3, pull down PORTA2, and pull up/down PORTA3 for clock division. In which case the user does not need to connect these signals to the target MON08 Header.

6.16 68HC908JG

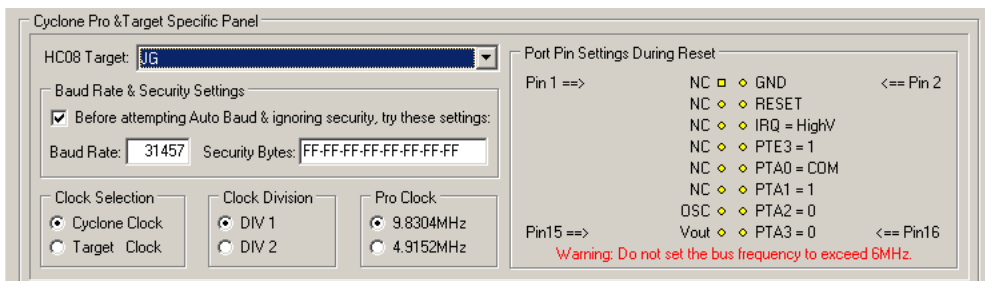


Figure 6-16: 68HC908JG Family MON08 Pinout

The target GND is connected to the Pin 2 of the target MON08 Header.

The target RESET line is directly connected to the Pin 4 of the target MON08 Header.

The target IRQ line is directly connected to the Pin 6 of the target MON08 Header.

PORTA0 from the target processor is connected to the target MON08 Header Pin 10, acting as the communications line. The Cyclone PRO pulls up this

signal with a 10K Ohm resistor to the target VDD.

PORTA1, PORTA2, PORTA3 and PORTE3 are used for entering monitor mode. By default the user may directly bring these signals out to the target MON08 Header.

Alternatively, the user may pull up PORTA1 and PORTE3, pull down PORTA2, and pull up/down PORTA3 for clock division. In which case the user does not need to connect these signals to the target MON08 Header.

6.17 68HC908JK

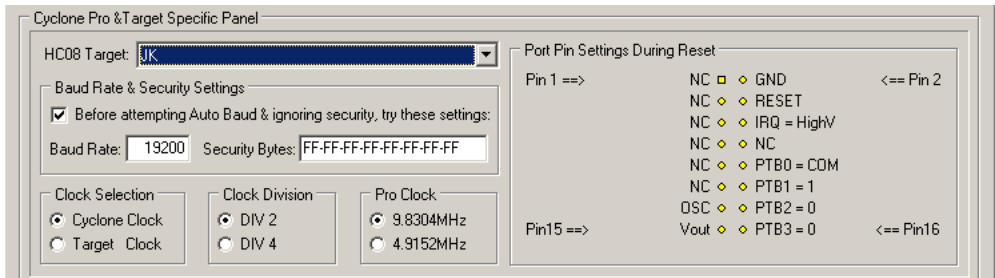


Figure 6-17: 68HC908JK Family MON08 Pinout

The target GND is connected to the Pin 2 of the target MON08 Header.

The target RESET line is directly connected to the Pin 4 of the target MON08 Header.

The target IRQ line is directly connected to the Pin 6 of the target MON08 Header.

PORTB0 from the target processor is connected to the target MON08 Header Pin 10, acting as the communications line. The Cyclone PRO pulls up this signal with a 10K Ohm resistor to the target VDD.

PORTB1, PORTB2 and PORTB3 are used for entering monitor mode. By default the user may directly bring these signals out to the target MON08 Header.

Alternatively, the user may pull up PORTB1 and pull down PORTB2, and pull up/down PORTB3 for clock division. In which case the user does not need to connect these signals to the target MON08 Header.

6.18 68HC908JL

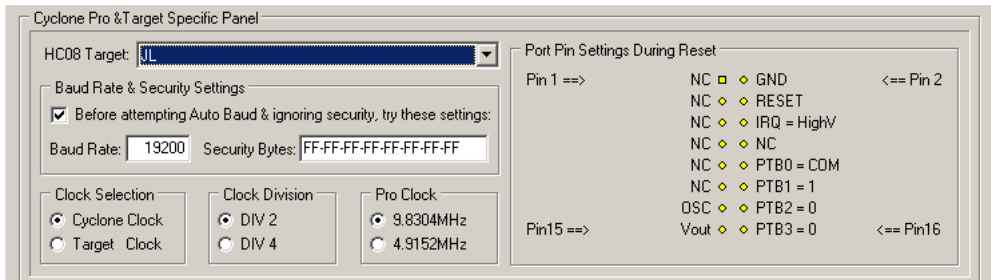


Figure 6-18: 68HC908JL Family MON08 Pinout

The target GND is connected to the Pin 2 of the target MON08 Header.

The target RESET line is directly connected to the Pin 4 of the target MON08 Header.

The target IRQ line is directly connected to the Pin 6 of the target MON08 Header.

PORTB0 from the target processor is connected to the target MON08 Header Pin 10, acting as the communications line. The Cyclone PRO pulls up this signal with a 10K Ohm resistor to the target VDD.

PORTB1, PORTB2 and PORTB3 are used for entering monitor mode. By default the user may directly bring these signals out to the target MON08 Header.

Alternatively, the user may pull up PORTB1 and pull down PORTB2, and pull up/down PORTB3 for clock division. In which case the user does not need to connect these signals to the target MON08 Header.

6.19 68HC908JR

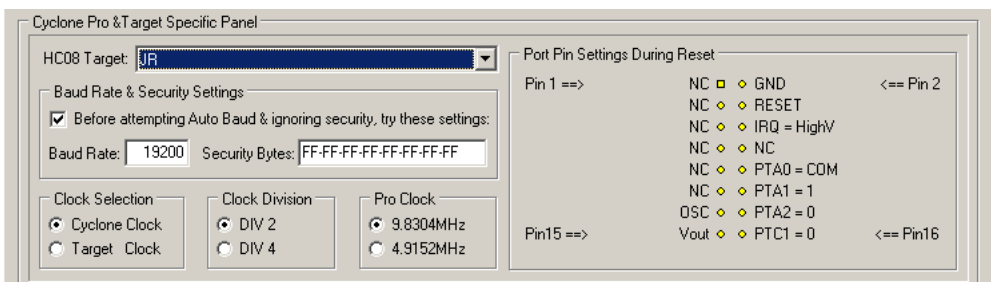


Figure 6-19: 68HC908JR Family MON08 Pinout

The target GND is connected to the Pin 2 of the target MON08 Header.

The target RESET line is directly connected to the Pin 4 of the target MON08 Header.

The target IRQ line is directly connected to the Pin 6 of the target MON08 Header.

PORTA0 from the target processor is connected to the target MON08 Header Pin 10, acting as the communications line. The Cyclone PRO pulls up this signal with a 10K Ohm resistor to the target VDD.

PORTA1, PORTA2 and PORTC1 are used for entering monitor mode. By default the user may directly bring these signals out to the target MON08 Header as shown, or ensure that they are pulled up/down according to the image.

6.20 68HC908JW

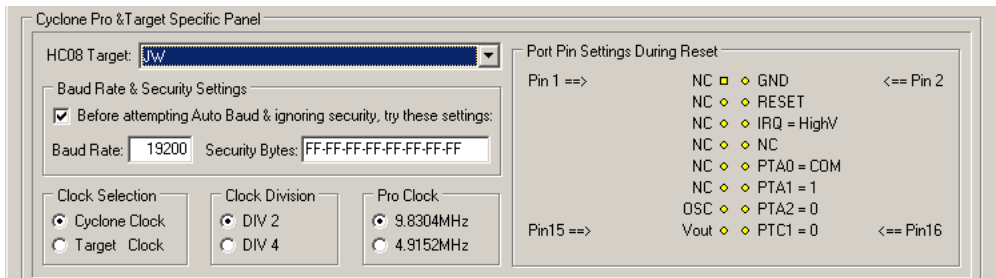


Figure 6-20: 68HC908JW Family MON08 Pinout

The target GND is connected to the Pin 2 of the target MON08 Header.

The target RESET line is directly connected to the Pin 4 of the target MON08 Header.

The target IRQ line is directly connected to the Pin 6 of the target MON08 Header.

PORTA0 from the target processor is connected to the target MON08 Header Pin 10, acting as the communications line. The Cyclone PRO pulls up this signal with a 10K Ohm resistor to the target VDD.

PORTA1, PORTA2 and PORTC1 are used for entering monitor mode. By

default the user may directly bring these signals out to the target MON08 Header, or ensure that they are pulled up/down according to the image.

6.21 68HC908KX

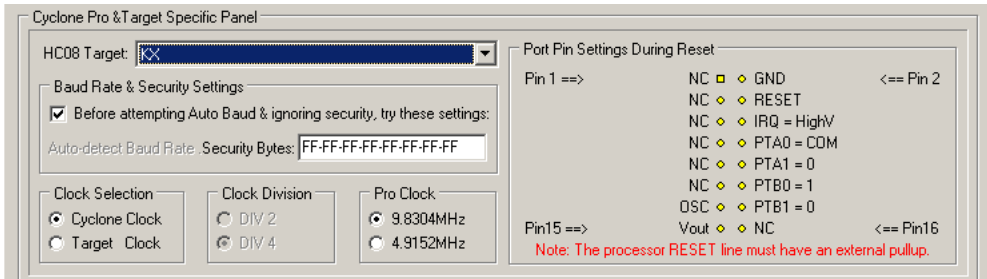


Figure 6-21: 68HC908KX Family MON08 Pinout

The target GND is connected to the Pin 2 of the target MON08 Header.

The user must pull up the RESET line to target VDD with an external resistor.

The target IRQ line is directly connected to the Pin 6 of the target MON08 Header.

PORTA0 from the target processor is connected to the target MON08 Header Pin 8, acting as the communications line. The Cyclone PRO pulls up this signal with a 10K Ohm resistor to the target VDD.

PORTA1, PORTB0 and PORTB1 are used for entering monitor mode. By default the user may directly bring these signals out to the target MON08 Header.

Alternatively, the user may pull up PORTB0 and pull down PORTA1 and PORTB1. In which case the user does not need to connect these signals to the target MON08 Header. The clock division is fixed Div 4.

An external clock signal must be provided to the target MCU's OSC pin. The Cyclone PRO provides this signal to MON08 header Pin 13.

This device has several different algorithms, each of which will program the calculated trim value to a different non-volatile memory location. Each algorithm is named to reflect the non-volatile trim location to which it corresponds, e.g., `908_kx98trim$fdff.08p` will program the trim value to location \$FDFF. Simply select the appropriate algorithm when configuring the Cyclone PRO to ensure that trim is programmed to the desired location.

6.22 68HC908LB

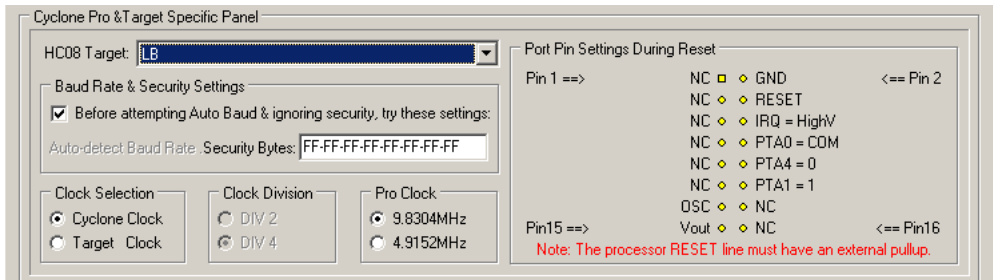


Figure 6-22: 68HC908LB Family MON08 Pinout

The target GND is connected to the Pin 2 of the target MON08 Header.

The user must pull up the RESET line to target VDD with an external resistor.

The target IRQ line is directly connected to the Pin 6 of the target MON08 Header.

PORTA0 from the target processor is connected to the target MON08 Header Pin 8, acting as the communications line. The Cyclone PRO pulls up this signal with a 10K Ohm resistor to the target VDD.

PORTA1 and PORTA4 are used for entering monitor mode. By default the user may directly bring these signals out to the target MON08 Header.

Alternatively, the user may pull up PORTA1 and pull down PORTA4. In which case the user does not need to connect these signals to the target MON08 Header. The clock division is fixed Div 4.

An external clock signal must be provided to the target MCU's OSC pin. The Cyclone PRO provides this signal to MON08 header Pin 13.

Please note that the Cyclone PRO will calculate the proper trim value for the device being programmed. The user has the option to program this trim value to \$FFC0.

6.23 68HC908LD

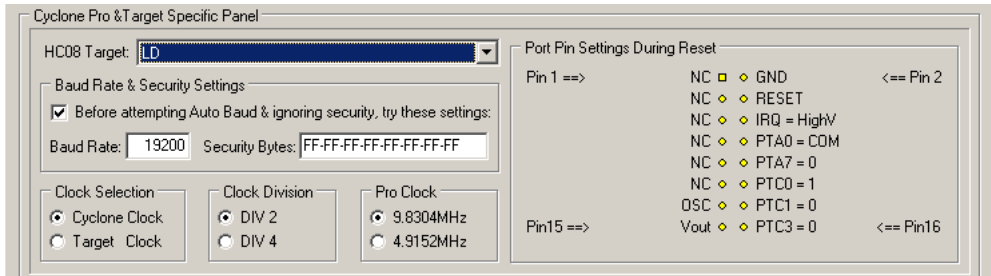


Figure 6-23: 68HC908LD Family MON08 Pinout

The target GND is connected to the Pin 2 of the target MON08 Header.

The target RESET line is directly connected to the Pin 4 of the target MON08 Header.

The target IRQ line is directly connected to the Pin 6 of the target MON08 Header.

PORTA0 from the target processor is connected to the target MON08 Header Pin 8, acting as the communications line. The Cyclone PRO pulls up this signal with a 10K Ohm resistor to the target VDD.

PORTA7, PORTC0, PORTC1 and PORTC3 are used for entering monitor mode. By default the user may directly bring these signals out to the target MON08 Header.

Alternatively, the user may pull up PORTC0, pull down PORTA7 and PORTC1, and pull up/down PORTC3 for clock division. In which case the user does not need to connect these signals to the target MON08 Header.

6.24 68HC908LJ

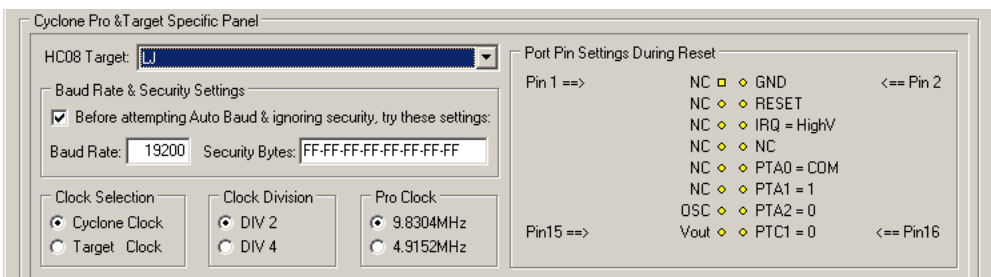


Figure 6-24: 68HC908LJ Family MON08 Pinout

The target GND is connected to the Pin 2 of the target MON08 Header.

The target RESET line is directly connected to the Pin 4 of the target MON08 Header.

The target IRQ line is directly connected to the Pin 6 of the target MON08 Header.

PORTA0 from the target processor is connected to the target MON08 Header Pin 10, acting as the communications line. The Cyclone PRO pulls up this signal with a 10K Ohm resistor to the target VDD.

PORTA1, PORTA2 and PORTC1 are used for entering monitor mode. By default the user may directly bring these signals out to the target MON08 Header.

Alternatively, the user may pull up PORTA1 and pull down PORTA2, and pull up/down PORTC1 for clock division. In which case the user does not need to connect these signals to the target MON08 Header.

6.25 68HC908LK

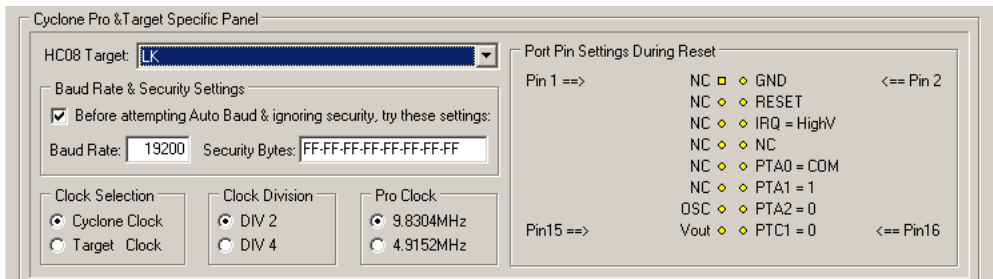


Figure 6-25: 68HC908LK Family MON08 Pinout

The target GND is connected to the Pin 2 of the target MON08 Header.

The target RESET line is directly connected to the Pin 4 of the target MON08 Header.

The target IRQ line is directly connected to the Pin 6 of the target MON08 Header.

PORTA0 from the target processor is connected to the target MON08 Header Pin 10, acting as the communications line. The Cyclone PRO pulls up this

signal with a 10K Ohm resistor to the target VDD.

PORTA1, PORTA2 and PORTC1 are used for entering monitor mode. By default the user may directly bring these signals out to the target MON08 Header.

Alternatively, the user may pull up PORTA1 and pull down PORTA2, and pull up/down PORTC1 for clock division. In which case the user does not need to connect these signals to the target MON08 Header.

6.26 68HC908LT

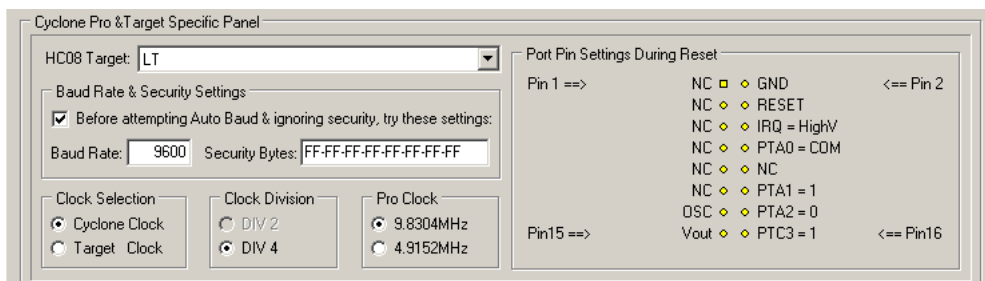


Figure 6-26: 68HC908LT Family MON08 Pinout

The target GND is connected to the Pin 2 of the target MON08 Header.

The user must pull up the RESET line to target VDD with an external resistor.

The target IRQ line is directly connected to the Pin 6 of the target MON08 Header.

PORTA0 from the target processor is connected to the target MON08 Header Pin 8, acting as the communications line. The Cyclone PRO pulls up this signal with a 10K Ohm resistor to the target VDD.

PORTA1 and PORTA2 are used for entering monitor mode. By default the user may directly bring these signals out to the target MON08 Header.

Alternatively, the user may pull up PORTA1 and pull down PORTC3. In which case the user does not need to connect these signals to the target MON08 Header. The clock division is fixed Div 4.

Please note that the Cyclone PRO will calculate the proper trim value for the device being programmed.

6.27 68HC908LV

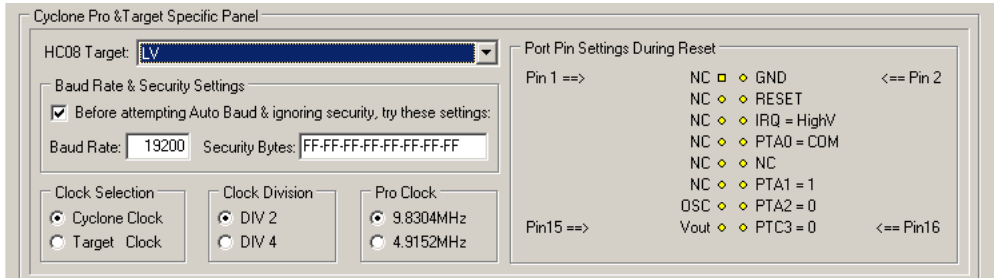


Figure 6-27: 68HC908LV Family MON08 Pinout

The target GND is connected to the Pin 2 of the target MON08 Header.

The user must pull up the RESET line to target VDD with an external resistor.

The target IRQ line is directly connected to the Pin 6 of the target MON08 Header.

PORTA0 from the target processor is connected to the target MON08 Header Pin 8, acting as the communications line. The Cyclone PRO pulls up this signal with a 10K Ohm resistor to the target VDD.

PORTA1 and PORTA4 are used for entering monitor mode. By default the user may directly bring these signals out to the target MON08 Header.

Alternatively, the user may pull up PORTA1 and pull down PORTA4. In which case the user does not need to connect these signals to the target MON08 Header. The clock division is fixed Div 4.

Please note that the Cyclone PRO will calculate the proper trim value for the device being programmed.

6.28 68HC908MR4/8

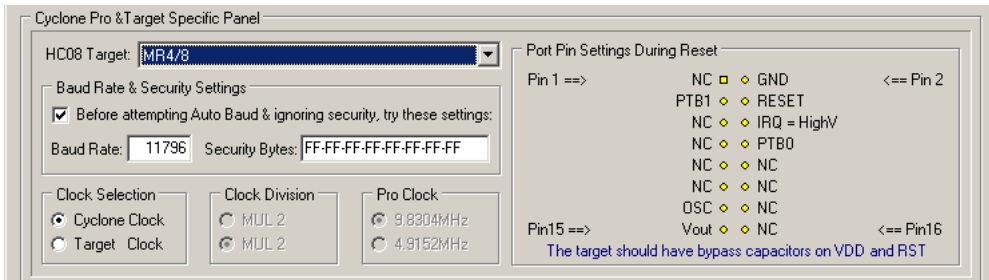


Figure 6-28: 68HC908MR4/8 Family MON08 Pinout

The target GND is connected to the Pin 2 of the target MON08 Header.

The target RESET line is directly connected to the Pin 4 of the target MON08 Header.

The target IRQ line is directly connected to the Pin 6 of the target MON08 Header.

PORTB0 from the target processor is connected to the target MON08 Header Pin 8, acting as the target MR4/8 data receiving line. The Cyclone PRO pulls up this signal with a 10K Ohm resistor to the target VDD.

PORTB1 from the target processor is connected to the target MON08 Header Pin 3, acting as the target MR4/8 data transmitting line. The user need to pull down this signal with a 10K resistor to GND.

6.29 68HC908MR16/32

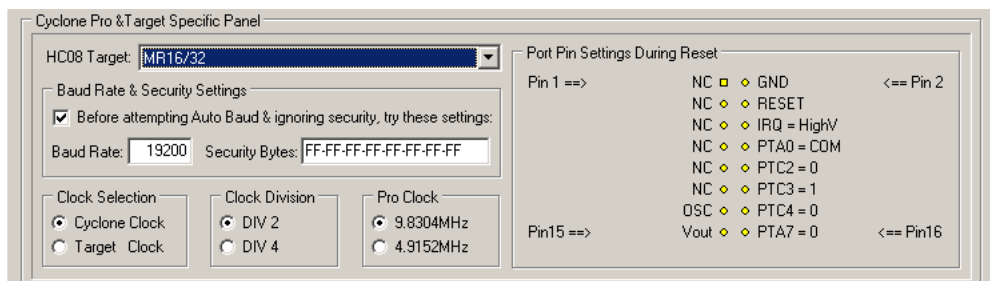


Figure 6-29: 68HC908MR16/32 Family MON08 Pinout

The target GND is connected to the Pin 2 of the target MON08 Header.

The target RESET line is directly connected to the Pin 4 of the target MON08 Header.

The target IRQ line is directly connected to the Pin 6 of the target MON08 Header.

PORTA0 from the target processor is connected to the target MON08 Header Pin 8, acting as the communications line. The Cyclone PRO pulls up this signal with a 10K Ohm resistor to the target VDD.

PORTA7, PORTC2, PORTC3 and PORTC4 are used for entering monitor mode. By default the user may directly bring these signals out to the target MON08 Header.

Alternatively, the user may pull up PORTC3, pull down PORTA7 and PORTC4, and pull up/down PORTC2 for clock division. In which case the user does not need to connect these signals to the target MON08 Header.

6.30 68HC908QL

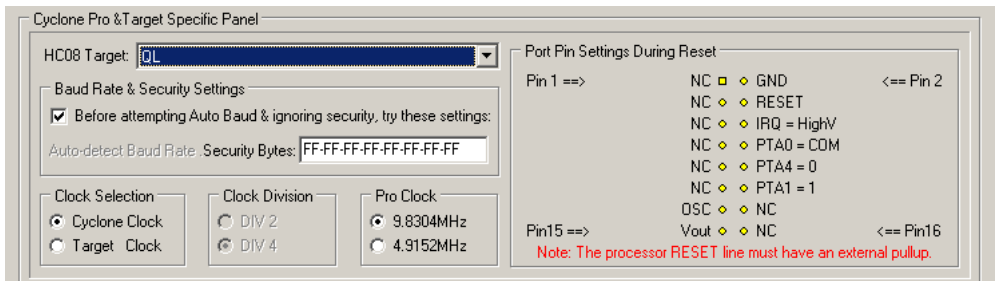


Figure 6-30: 68HC908QL Family MON08 Pinout

The target GND is connected to the Pin 2 of the target MON08 Header.

The user must pull up the RESET line to target VDD with an external resistor.

The target IRQ line is directly connected to the Pin 6 of the target MON08 Header.

PORTA0 from the target processor is connected to the target MON08 Header Pin 8, acting as the communications line. The Cyclone PRO pulls up this signal with a 10K Ohm resistor to the target VDD.

PORTA1 and PORTA4 are used for entering monitor mode. By default the user may directly bring these signals out to the target MON08 Header.

Alternatively, the user may pull up PORTA1 and pull down PORTA4. In which

case the user does not need to connect these signals to the target MON08 Header. The clock division is fixed Div 4.

An external clock signal must be provided to the target MCU's OSC pin. The Cyclone PRO provides this signal to MON08 header Pin 13.

Please note that the Cyclone PRO will calculate the proper trim value for the device being programmed and the user has the option to program this trim value to \$FFC0.

6.31 68HC908QB

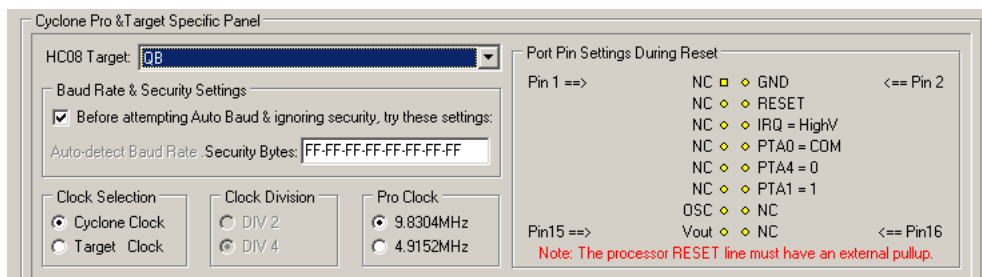


Figure 6-31: 68HC908QB Family MON08 Pinout

The target GND is connected to the Pin 2 of the target MON08 Header.

The user must pull up the RESET line to target VDD with an external resistor.

The target IRQ line is directly connected to the Pin 6 of the target MON08 Header.

PORTA0 from the target processor is connected to the target MON08 Header Pin 8, acting as the communications line. The Cyclone PRO pulls up this signal with a 10K Ohm resistor to the target VDD.

PORTA1 and PORTA4 are used for entering monitor mode. By default the user may directly bring these signals out to the target MON08 Header.

Alternatively, the user may pull up PORTA1 and pull down PORTA4. In which case the user does not need to connect these signals to the target MON08 Header. The clock division is fixed Div 4.

An external clock signal must be provided to the target MCU's OSC pin. The Cyclone PRO provides this signal to MON08 header Pin 13.

The internal clock on this device has several different tap points, each of which will result in different clock frequencies. There are several different

algorithms for the device, each of which is optimized for a specific tap point frequency. To trim to a specified frequency, simply select the corresponding algorithm.

6.32 68HC908QC

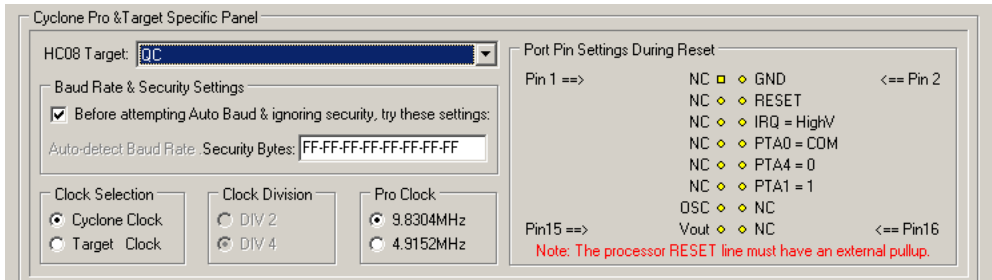


Figure 6-32: 68HC908QC Family MON08 Pinout

The target GND is connected to the Pin 2 of the target MON08 Header.

The user must pull up the RESET line to target VDD with an external resistor.

The target IRQ line is directly connected to the Pin 6 of the target MON08 Header.

PORTA0 from the target processor is connected to the target MON08 Header Pin 8, acting as the communications line. The Cyclone PRO pulls up this signal with a 10K Ohm resistor to the target VDD.

PORTA1 and PORTA4 are used for entering monitor mode. By default the user may directly bring these signals out to the target MON08 Header.

Alternatively, the user may pull up PORTA1 and pull down PORTA4. In which case the user does not need to connect these signals to the target MON08 Header. The clock division is fixed Div 4.

An external clock signal must be provided to the target MCU's OSC pin. The Cyclone PRO provides this signal to MON08 header Pin 13.

The internal clock on this device has several different tap points, each of which will result in different clock frequencies. There are several different algorithms for the device, each of which is optimized for a specific tap point frequency. To trim to a specified frequency, simply select the corresponding algorithm.

6.33 68HC908QT

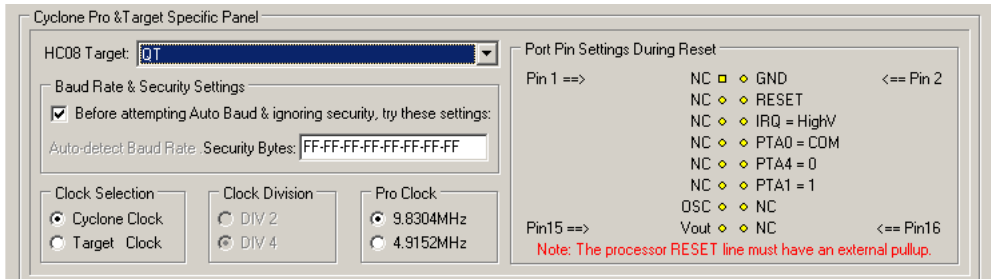


Figure 6-33: 68HC908QT Family MON08 Pinout

The target GND is connected to the Pin 2 of the target MON08 Header.

The user must pull up the RESET line to target VDD with an external resistor.

The target IRQ line is directly connected to the Pin 6 of the target MON08 Header.

PORTA0 from the target processor is connected to the target MON08 Header Pin 8, acting as the communications line. The Cyclone PRO pulls up this signal with a 10K Ohm resistor to the target VDD.

PORTA1 and PORTA4 are used for entering monitor mode. By default the user may directly bring these signals out to the target MON08 Header.

Alternatively, the user may pull up PORTA1 and pull down PORTA4. In which case the user does not need to connect these signals to the target MON08 Header. The clock division is fixed Div 4.

An external clock signal must be provided to the target MCU's OSC pin. The Cyclone PRO provides this signal to MON08 header Pin 13.

Please note that the Cyclone PRO will calculate the proper trim value for the device being programmed and the user has the option to program this trim value to \$FFC0.

6.35 68HC908RF

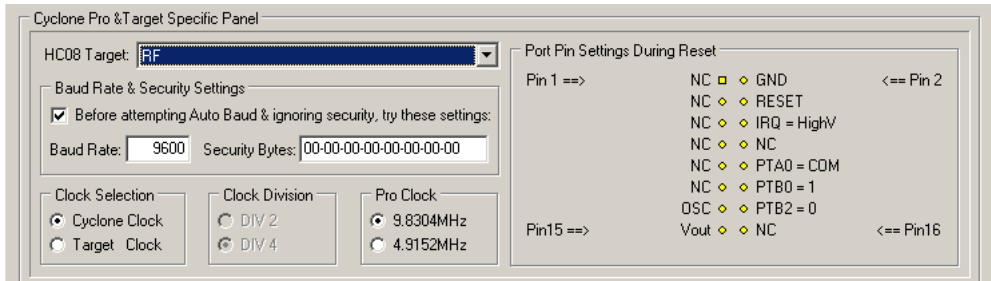


Figure 6-35: 68HC908RF Family MON08 Pinout

The target GND is connected to the Pin 2 of the target MON08 Header.

The target RESET line is directly connected to the Pin 4 of the target MON08 Header.

The target IRQ line is directly connected to the Pin 6 of the target MON08 Header.

PORTA0 from the target processor is connected to the target MON08 Header Pin 10, acting as the communications line. The Cyclone PRO pulls up this signal with a 10K Ohm resistor to the target VDD.

PORTB0 and PORTB2 are used for entering monitor mode. By default the user may directly bring these signals out to the target MON08 Header.

Alternatively, the user may pull up PORTB0 and pull down PORTB2. In which case the user does not need to connect these signals to the target MON08 Header. The clock division is fixed Div 4.

6.36 68HC908RK

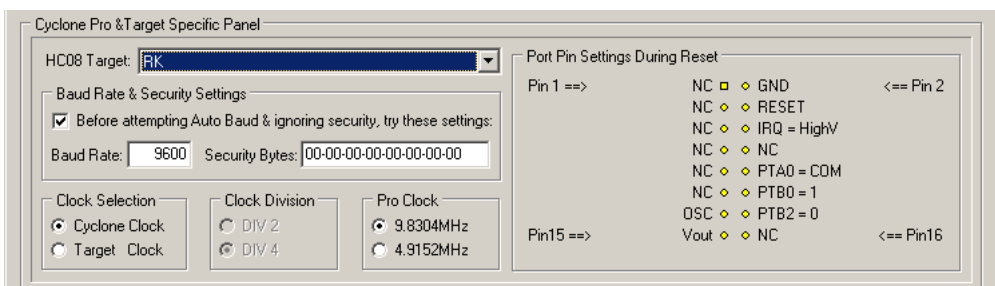


Figure 6-36: 68HC908RK Family MON08 Pinout

The target GND is connected to the Pin 2 of the target MON08 Header.

The target RESET line is directly connected to the Pin 4 of the target MON08 Header.

The target IRQ line is directly connected to the Pin 6 of the target MON08 Header.

PORTA0 from the target processor is connected to the target MON08 Header Pin 10, acting as the communications line. The Cyclone PRO pulls up this signal with a 10K Ohm resistor to the target VDD.

PORTB0 and PORTB2 are used for entering monitor mode. By default the user may directly bring these signals out to the target MON08 Header.

Alternatively, the user may pull up PORTB0 and pull down PORTB2. In which case the user does not need to connect these signals to the target MON08 Header. The clock division is fixed Div 4.

6.37 68HC908SR

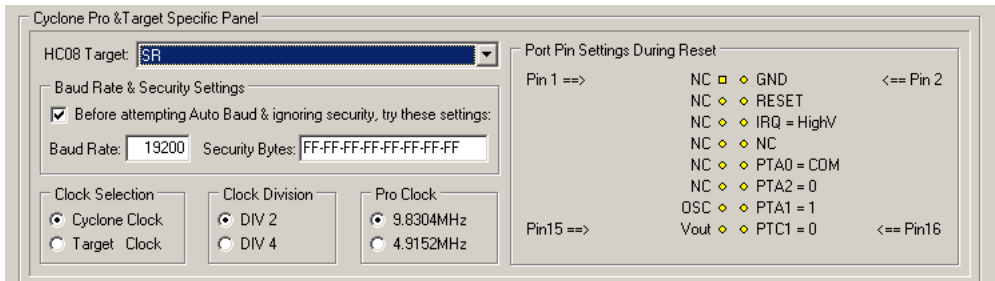


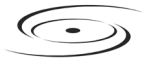
Figure 6-37: 68HC908SR Family MON08 Pinout

The target GND is connected to the Pin 2 of the target MON08 Header.

The target RESET line is directly connected to the Pin 4 of the target MON08 Header.

The target IRQ line is directly connected to the Pin 6 of the target MON08 Header.

PORTA0 from the target processor is connected to the target MON08 Header Pin 10, acting as the communications line. The Cyclone PRO pulls up this signal with a 10K Ohm resistor to the target VDD.



PORTA1, PORTA2 and PORTC1 are used for entering monitor mode. By default the user may directly bring these signals out to the target MON08 Header.

Alternatively, the user may pull up PORTA1 and pull down PORTA2, and pull up/down PORTC1 for clock division. In which case the user does not need to connect these signals to the target MON08 Header.

7 PC-HOSTED DEBUG/PROGRAMMING SOFTWARE

Free or low-cost software options for interactively programming and debugging HC08/HCS08/HC12/HC(S)12(X)/ColdFire V1 MCUs from the PC are available from P&E Microcomputer Systems (www.pemicro.com) and Freescale (www.freescale.com).

Note: The user should make sure they have the most recent version of these software kits. The latest updates can be downloaded from the web pages listed in **Section 7.1.1 - Latest Updates - P&E Software** and **Section 7.2.1 - Freescale CodeWarrior**.

7.1 P&E Microcomputer Systems Software

P&E's software packages contain the WinIDE integrated development environment, which pulls together an assembler, in-circuit simulator, flash memory programmer, and in-circuit debugger.

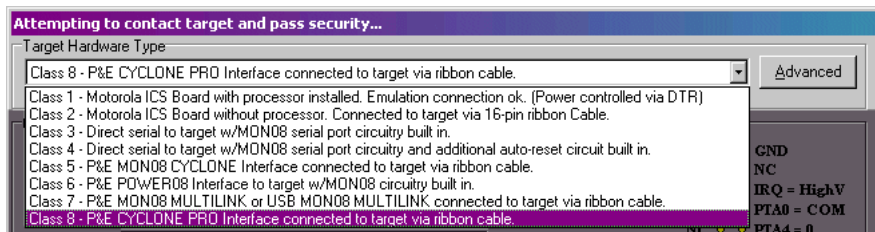


Figure 7-1: Hardware Selection in P&E PROG08SZ

7.1.1 Latest Updates - P&E Software

The most recent updates of P&E's 68HC08 software products are available to download, after a brief registration, at:

<http://www.pemicro.com/ics08>.

7.1.2 In-Circuit Debugger

The ICD In-Circuit Debugger uses the Cyclone PRO to control the target HC08/HCS08/HC(S)12(X)/ColdFire V1 device. With the ICD In-Circuit Debugger you can load code into the on-chip RAM, run code out of RAM or FLASH (already programmed by the In-Circuit Programmer), and set many software breakpoints and a single hardware (meaning in FLASH) breakpoint. The main advantage of using the ICD is that your application runs in real-time at the full bus speed of the processor.

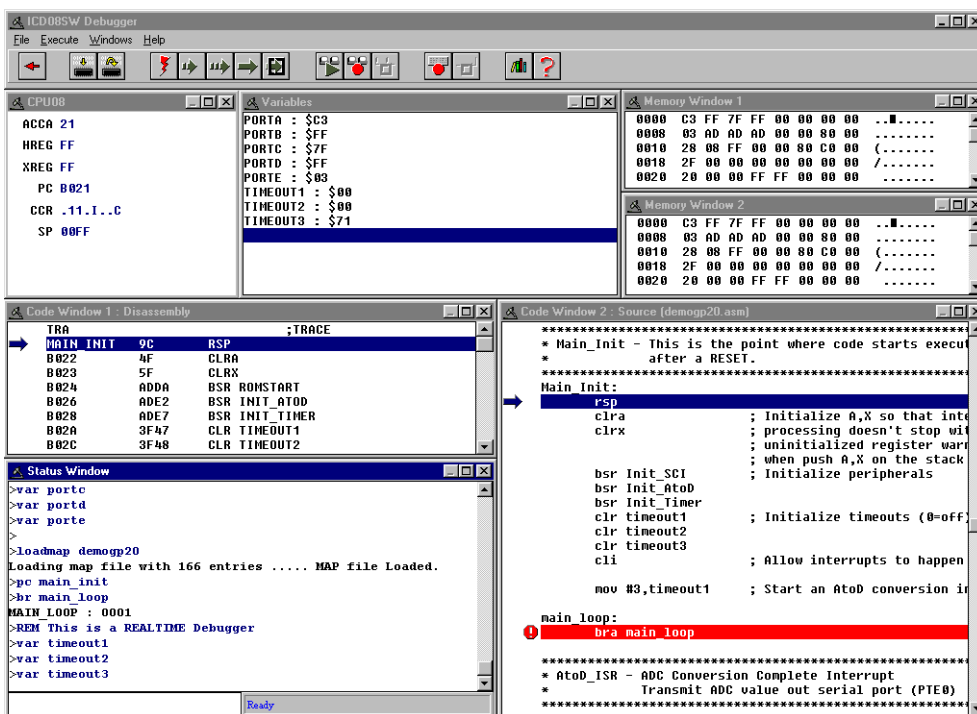


Figure 7-2: ICD08SZ Debugger Screen Snapshot

Debugger features include:

- Full-speed in-circuit emulation
- Breakpoints with counters on the Nth execution
- Variables window showing multiple data types

- Real-time execution as well as multiple tracing modes
- Startup and Macro files for automating the debug process
- Context-sensitive help for all commands
- Support for symbolic register files
- Full source-level debugging

7.1.3 In-Circuit Programmer

The PROG In-Circuit Programmer is a general-purpose programmer which allows the user to program any HC08/HCS08/HC(S)12(X)/ColdFire V1 device with on-chip EEPROM/FLASH, either from an object file (Motorola .S19 format), or byte by byte.

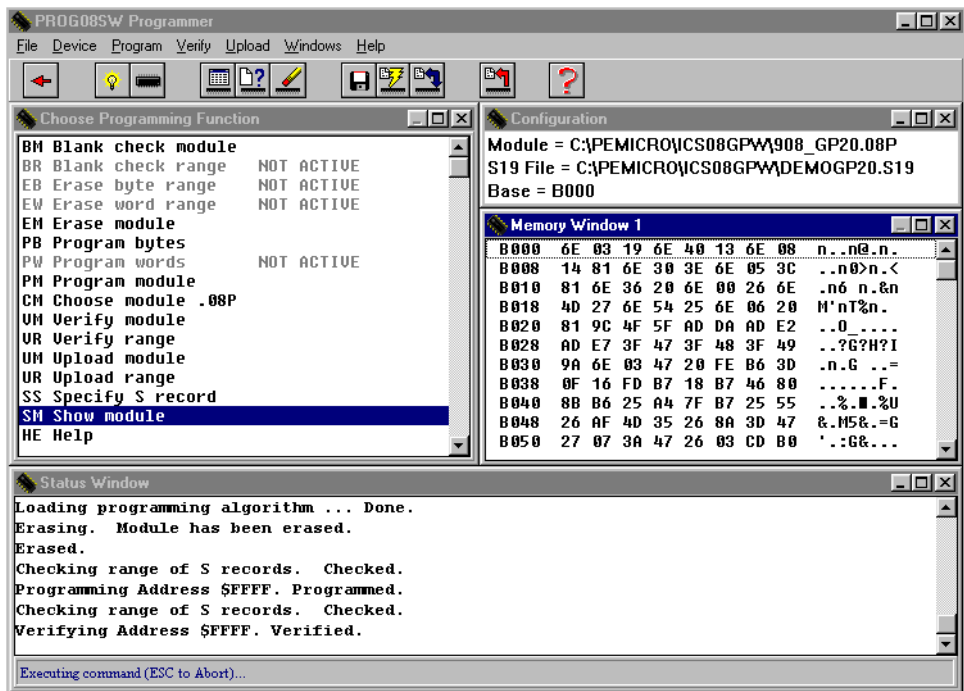
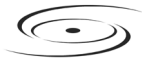


Figure 7-3: PROG08SZ Programmer Screen Snapshot



The PROG08SZ is simple to operate: after clicking the “Contact target with these settings” button, if the programmer successfully contacts the target it will ask you for the algorithm you wish to use during programming. Select the proper algorithm for the device you are attempting to program. Then simply select the s-record object you wish to program using the “SS” command. Now the setup of the PROG08SZ is complete and you are ready for operations on the target EEPROM/FLASH. You may choose “EM – Erase Module” to erase the target EEPROM/FLASH. Then use “BM – Blank Check Module” to see if the target EEPROM/FLASH is indeed erased. After that, you may choose “PM – Program Module” to program the S-record object into the target. Finally, you may use “VC – Verify CRC Checksum” to verify that the contents are properly programmed in the target memory.

7.1.4 Command Line Programmer

CPROG is a command line programmer that allows quick turn-around time for programming target MCUs. The user may create a script file to instruct the software to execute specific commands in sequence. Please refer to CPROG documentation for more information.

7.2 Freescale Software

The special edition of Freescale’s CodeWarrior studio offers absolute assembly and provides debugging capabilities based on P&E’s programming and debug technologies.

7.2.1 Freescale CodeWarrior

A programming or debug session with the project-based CodeWarrior IDE may be launched by double-clicking on the project name (format is `projectname.mcp`) from your file storage. Starting a new project is a little more challenging, but the tutorials, FAQs, and Quick Start Guides are easy to follow and have you building a new project, using pre-built templates, in a short time.

The following example illustrates how to program and debug an M68HC908 MCU from within the CodeWarrior IDE.

Here are the main steps in programming the FLASH with CodeWarrior and starting a debug session.

1. a. Launch the CodeWarrior CW08 software and create a new project,
or
b. Double-click on your project file (`projectname.mcp`)

The Project Manager window appears. See **Figure 7-4**.

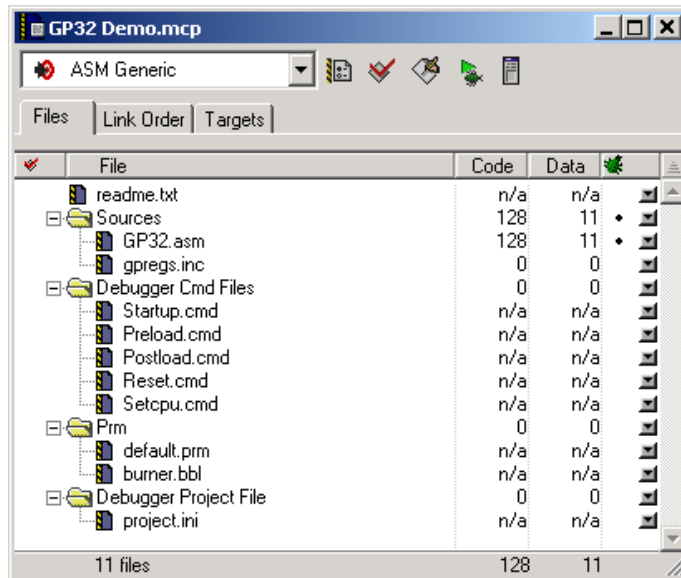


Figure 7-4: CodeWarrior Project Window

2. Click the + sign to expand the Sources folder.
3. Modify the source file if necessary.
4. Click the Debug icon (green arrow). The True-Time Simulator and Real-Time Debugger launches. See **Figure 7-5**.

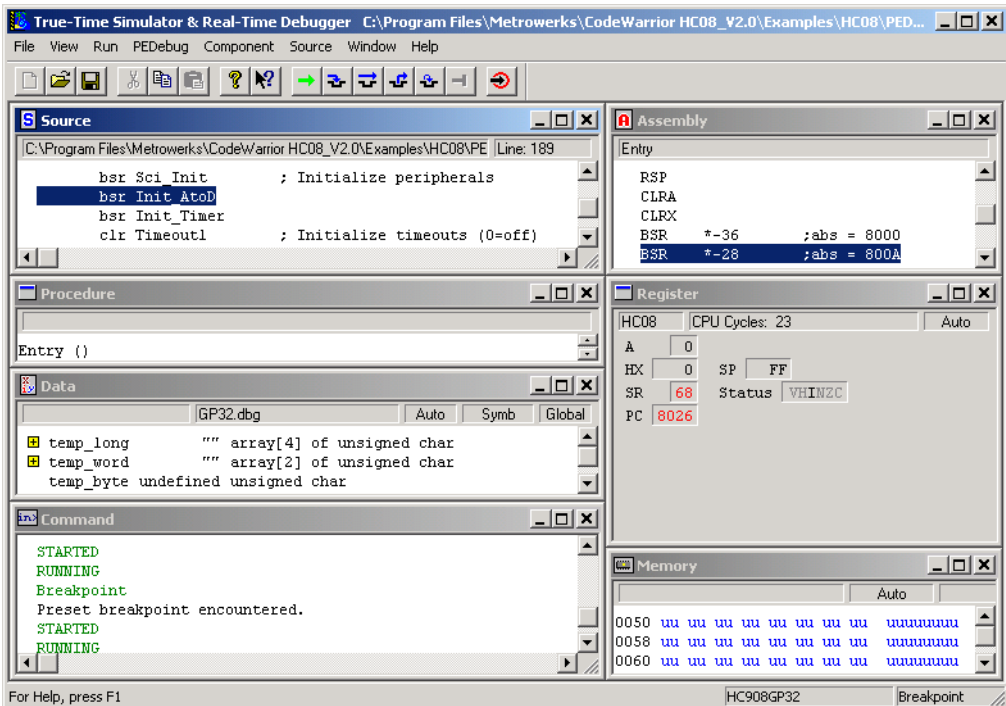


Figure 7-5: True-Time Simulator & Real-Time Debugger Window

5. If you created a new product using the Codewarrior project wizard, and selected a P&E Debug Interface for your project target, then at this point you should see the P&E Communication Assistant.

Alternately, you can select your connection type (target) manually through the HiWave software. In the Component... menu, select the “set connection” menu option. This will allow you to choose from the various connection types available in your version of Codewarrior.

In-Depth detail about selecting and configuring any of the P&E connections can be found later in the manual. For more information using a Cyclone Pro in Codewarrior with an HC08 device, please see **Section 7.3 - HC08 P&E Multilink/Cyclone PRO Connections**. For more information using a Cyclone Pro in Codewarrior with an HCS08

device, please see **Section 7.4 - HCS08 P&E Multilink/Cyclone PRO Connections** section 7.4. For more information using a Cyclone Pro in Codewarrior with an HC(S)12 device, please see the Codewarrior User's manual.

6. After you have configured the Cyclone Pro properly, click the "Contact Target with These Setting" button.

The "Attempting to contact target and pass security" window appears.

7. Select the appropriate class in Target Hardware Type (Class VII for MON08 MULTILINK and Class VIII for Cyclone PRO).
8. Click Contact target with these settings...
9. Follow the Power Cycle dialog instructions.
10. Click Yes in Confirm window. (**Figure 7-6**)
11. Click Yes in Erase and Program FLASH window. (**Figure 7-7**)
12. Follow the subsequent Power Cycle dialog instructions as the scripted procedure automatically establishes communications, erases the FLASH if necessary, and programs the FLASH.

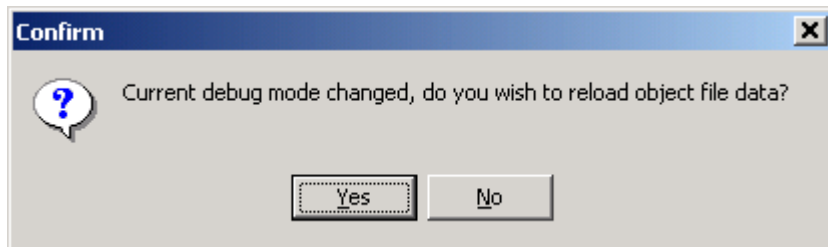


Figure 7-6: Confirm Window

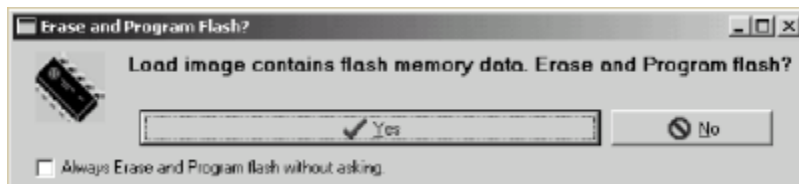


Figure 7-7: Erase And Program Flash Window

At this point, the FLASH memory is programmed and ready for debug. The True-Time Simulator & Real-Time Debugger integrates the debugger tools from P&E Microcomputer Systems in this example. The windows look slightly different between the ICD and True-Time tools but the same basic debugger (ICD) drives both.

7.3 HC08 P&E Multilink/Cyclone PRO Connections

7.3.1 Introduction

The HC08 P&E Multilink/Cyclone Pro Connection setting permits a connection to Class 5, 7 or 8 devices. Please see the descriptions below for a definition of each interface class. HC08 P&E Multilink/Cyclone Pro Connection Interface mode allows the user to debug code, as the firmware is fully resident in the FLASH of the microprocessor. The operation of all modules fully reflects the actual operation of the on-board resources.

7.3.2 Connection Procedure

1. Choose the P&E Multilink/Cyclone Pro option from the Set Connection dialog. See **Figure 7-8**.

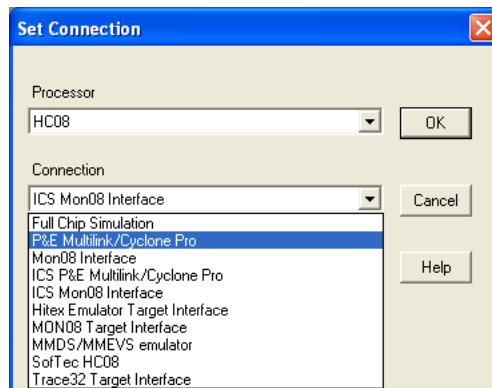


Figure 7-8: Set P&E Multilink/Cyclone PRO

2. The device power selection allows the user to specify whether the target is 2, 3, or 5 Volts, and whether this power is switched/generated

by the P&E interface or if it is separately supplied to the target and under user control. If it is under user control, the software will use dialog boxes to ask the user to power the target up and down when necessary (similar to Class II-IV). See **Figure 7-9**.

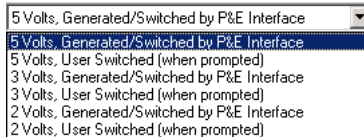


Figure 7-9: Device Power Dialog

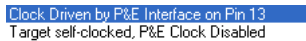


Figure 7-10: HC08 Device Clock Selection Box

3. The device clock menu allows two options. See **Figure 7-10**.
 - P&E provides clock to target
 - The device has its own clock
4. Click on “Refresh” which should display all P&E devices that are automatically detected, such as Cyclone Pro USB, Cyclone Pro Ethernet

and USB-ML-MON08 Multilink. See **Figure 7-11**.

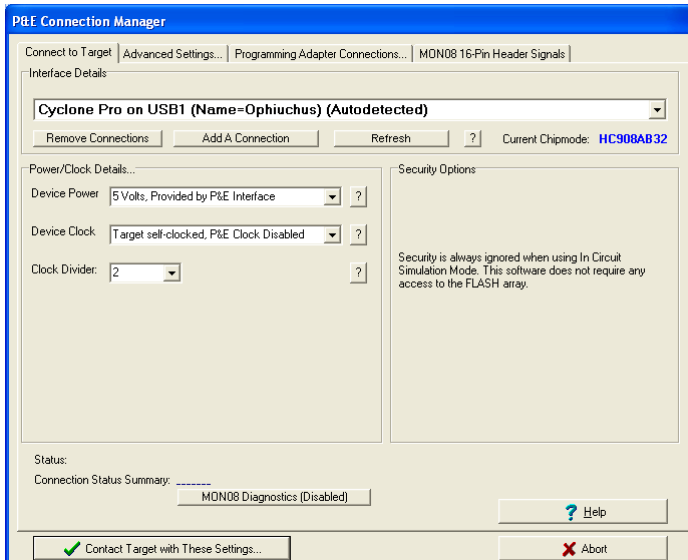


Figure 7-11: P&E Multilink/Cyclone Pro Connection Manager

5. To add a serial/parallel port P&E device such as Cyclone Pro Serial, MON08 Multilink and Cyclone Pro Ethernet (IP outside of subnet mask), proceed to the Interface Selection Manager by pressing the “Add A Connection” button. See **Figure 7-12**. For more information about configuring Cyclone Pro for ethernet operation, please see the Cyclone Pro User's Manual.

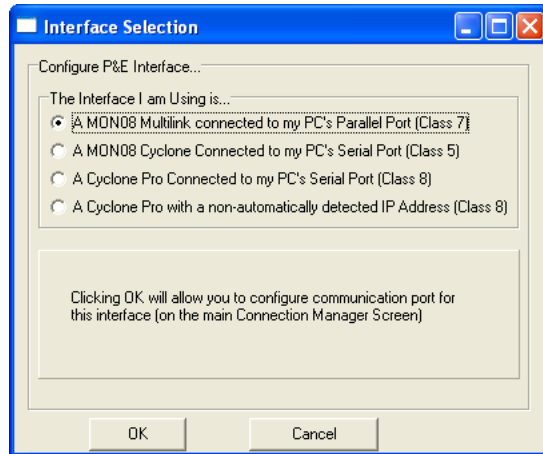


Figure 7-12: P&E Multilink/Cyclone Pro Manual Interface Selection

6. To remove a manually configured interface, please click on the “Remove Connection” button and choose the interface to be deleted. See **Figure 7-13**.

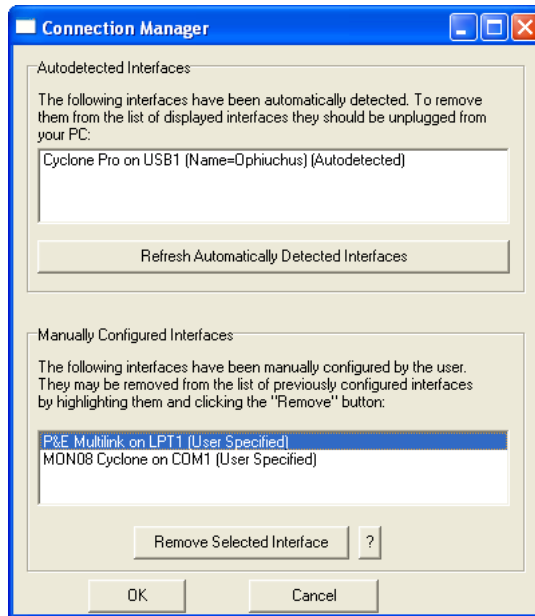


Figure 7-13: Remove A Manually Configured Multilink/Cyclone Pro Interface

7.3.3 Advanced Settings Tab

The Advanced Settings tab allows the user to set specific protocol settings. The following is an explanation of each part of the advanced settings dialog.

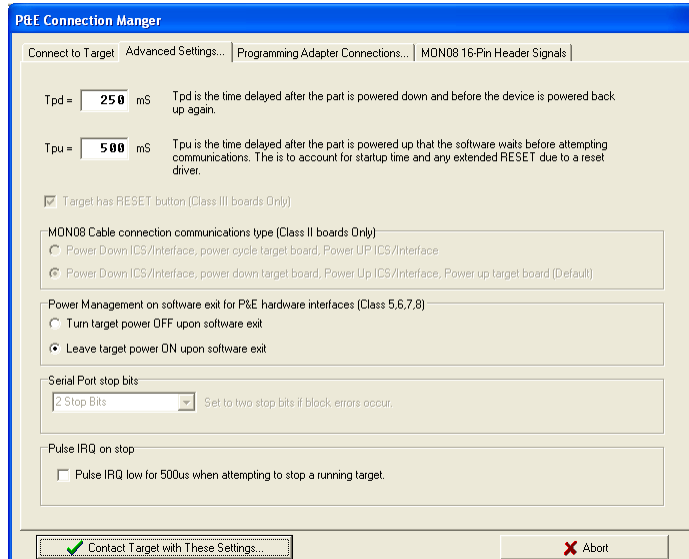


Figure 7-14: Advanced Settings Tab

7.3.3.1 Tpd And Tpu Timing

Tpu and Tpd will set the power-up and power-down delay (respectively) that will be observed when power-cycling a target for entry into Monitor Mode. These settings are only valid for devices with automatically controlled power.

Whenever power is automatically switched off, or is manually requested to be switched off, the software waits for an amount of time equal to the Tpd delay time before proceeding to the connection protocol. This is because a board or power supply may have capacitance which holds the power up for a short time after the supply has been switched off, but the supply voltage must reach less than 0.1v before it is turned back on if a Power-On reset is to occur.

Whenever power is automatically switched on, or is manually requested to be switched on, the software waits for an amount of time equal to the Tpu delay time before attempting to contact the 68HC08 processor. This is to allow time not only for power to be fully available, but to wait until any reset driver has finally released the RESET line.

7.3.3.2 Target Has RESET Button (Class III Boards Only)

The software occasionally needs to get control of the target. On systems which are Class III boards with the monitor mode circuitry built-in (including RS-232 driver), there is no means to reset the target to gain control. If the board has a reset button, the software can use this to gain control of the target system. If this option is checked, the software will prompt the user to push the target reset button when a reset of the target system is desired. If the option is unchecked, the software will ask the user to power cycle the target system to achieve a reset.

7.3.3.3 MON08 Cable Connection Communications Type (Class II Boards Only)

This selection box is valid only for Class II hardware configurations using the MON08 cable. It allows the user to specify the sequence that the software uses to power up the ICS system. When the software tries to create a power-on reset condition, two events must occur:

1. Power of the target MCU must go below 0.1v. This means that the processor can not be receiving power from its power pins, nor can it have a significant voltage being driven on port pins or the IRQ line, as these will drive the MCU power back through these pins. It is crucial, therefore, to have the ICS and the Target both powered down at some point in time.
2. The processor MON08 configuration pins, including IRQ, must be properly driven when the target processor resets to drive it into monitor mode. If these pins are not set up properly before the processor powers up, the processor may start up in user mode.

Power Down ICS, Ask the user to power down their board, Power Up ICS, Ask the user to power up their board

This is the default option and should work for most, if not all, ICS08/Target Board solutions. Refer to the manual addendum under startup for the settings for a specific ICS board. It requires the user go through two dialog stages, and requires more time than simply cycling the power.

1. Software automatically powers down the ICS.

2. Software asks the user to power down the board as follows:

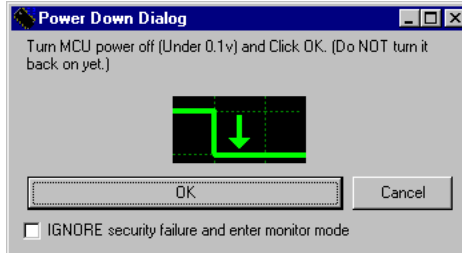


Figure 7-15: Power Down Dialog

3. Software automatically powers up the ICS, which configures the processor's MON08 configuration pins.
4. Software asks the user to power up the board as follows:

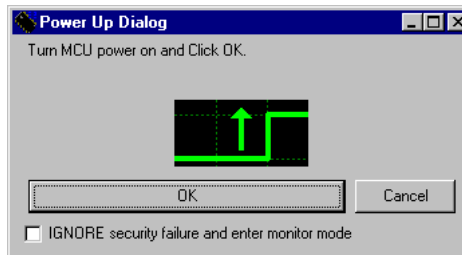


Figure 7-16: Power Up Dialog

Power Down ICS, Ask the user to power cycle their board, Power UP ICS

This option will work for many ICS boards as well, but relies on the fact that while the ICS is powered off, it will hold the target in reset until it is powered up itself and has configured the MON08 configuration pins. The sequence of events in this mode is:

1. Software automatically powers down the ICS.

2. Software asks the user to power cycle their board as follows:

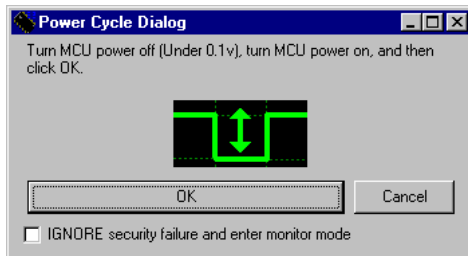


Figure 7-17: Power Cycle Dialog

3. Software automatically powers up the ICS, which configures the processors MON08 configuration pins.

7.3.3.4 Serial Port Stop Bits

Serial Port Stop Bits allow users of Class 1-4 devices that are experiencing unreliable communication to increase the number of stop bits to 2.

7.3.3.5 Pulse IRQ on Stop

The IRQ on Stop allows the users to execute a mechanism to stop a running HC08 microprocessor. For more information, please refer to the Stop a Running HC908 Target.pdf application note that is available from P&E Microcomputer Systems website: www.pemicro.com.

7.3.4 Target MCU Security Bytes

One of the steps that is necessary to properly bypass security is to provide the proper security code for the information that is programmed into the part. This holds true even when the part is blank. The security code consists of the 8 values which are currently stored in flash locations \$FFF6 - \$FFFD of the processor.

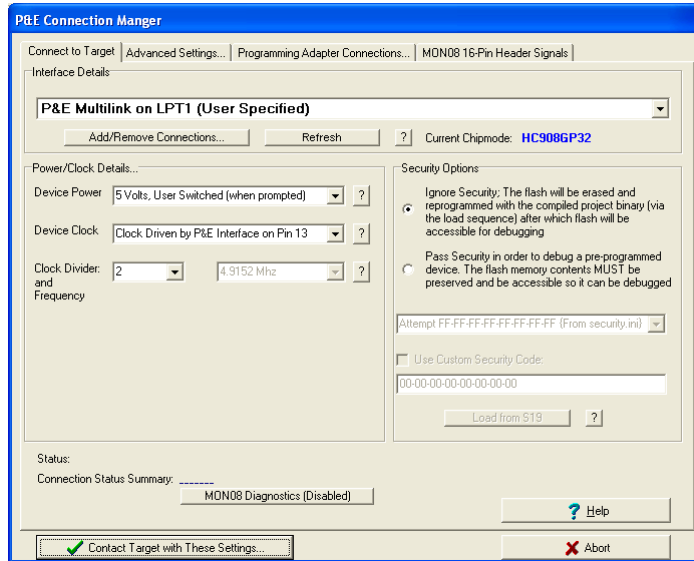


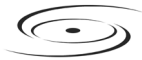
Figure 7-18: Connection Manager Dialog - P&E Multilink/Cyclone PRO

This dialog can be used by the user to manually enter the proper security bytes via the USER setting, or to load the security bytes from the same .S19 file which was programmed. The bytes are loaded from an .S19 file by clicking the “Load from S19” button.

7.3.4.1 IGNORE Security Failure And Enter Monitor Mode

This checkbox can be used to cause the software to ignore a failure to properly pass the 68HC08 security check. If the checkbox is set, the software will attempt to establish monitor mode communications regardless of the security status. As long as the Baud and Port are correct, and the device has been properly powered, this will allow monitor mode entry. By ignoring the security check failure you may use monitor mode, but the ROM/Flash will not be accessible.

Note: If a connection is not established for a reason other than security failure, the connection dialog will always appear.



7.3.5 STATUS

The status area consists of one status string following the “Status:” label, and seven items which list the state of the last attempt to connect to a target and pass security. The description for these items is as follows:

0 – ICS Hardware loopback detected:

Every ICS or board which supports MON08 has a serial loopback in hardware which, by connecting the transmit and receive lines, automatically echoes characters from the PC. A valid character transmitted from the PC should be echoed once by the loopback circuitry on the board and once by the monitor of the target processor itself. This status indicates whether or not the first echoed character from the hardware loopback was received when one of the security bytes was transmitted. If the status is ‘N’, which indicates that the character was not received, it is most likely due to one of the following reasons:

1. Wrong Com Port specified.
2. The baud rate specified was incorrect (probably too low).
3. The ICS/Target is not connected.
4. No Power to the ICS.

If this status bit responded with an ‘N’, you must correct this before analyzing the reset of the status bits.

1 – Device echoed some security bytes:

The monitor resident in a 68HC08 device automatically echoes every incoming character when it is in monitor mode. A valid character transmitted from the PC should be echoed once by the loopback circuitry on the board and once by the monitor of the target processor itself. This status indicates whether or not the second echoed character from the monitor response was received when one of the security bytes was transmitted. If the status is ‘N’, which indicates that the character was not received, or not received properly, it is most likely due to one of the following reasons:

1. The baud rate specified was incorrect.
2. The part did not start the monitor mode security check on reset. Signals to force monitor mode may be incorrect.
3. No Power to the ICS.

If this status bit responded with an ‘N’, you must correct this before analyzing

the reset of the status bits.

2 – Device echoed all security bytes:

In order to pass security, the software must send 8 security bytes to the processor. The processor should echo each of these eight bytes twice. If all 8 bytes did not get the proper two-byte echo, this flag will be 'N'. Reasons for this include:

1. The part did not start the monitor mode security check on reset. Signals to force monitor mode may be incorrect.
2. The baud rate specified was incorrect.
3. The processor was not reset properly. Check the “Target Hardware Type” and if you are connecting to a class II board, check the “MON08 cable communication connections type” in the “advanced settings” dialog.

3 – Device signaled monitor mode with a break:

Once the processor has properly received the 8 bytes from the PC software to complete its security check, it should transmit a break character to the PC signaling entry into monitor mode. This break should be sent regardless of whether the security check was successfully passed. If a break was not received from the processor, this flag will be 'N'. Reasons for this include:

1. The baud rate specified was incorrect.
2. The processor was not reset properly. Check the “Target Hardware Type”. If you are connecting to a class II board, check the “MON08 cable communication connections type” in the “advanced settings” dialog.

4 – Device entered monitor mode:

Once the software has received, or failed to receive, a break from the processor, it attempts to communicate with the monitor running on the 68HC08 processor. It tries to read the monitor version number by issuing a monitor mode read. If the processor fails to respond properly to this command, this flag will be 'N'.

5 – Reset was Power-On Reset:

If the device properly entered monitor mode (4), the software will read the reset status register (RSR). This read does not affect the security sequence, and occurs purely for diagnostic reasons. The reset status register indicates the conditions under which the processor underwent the last reset. For the

software to pass the security check properly, it MUST first cause the processor to undergo a Power-On Reset. The software reads the reset status register to determine if the last reset was indeed caused by power-on. The result of the reset status register is indicated in parentheses after the flag value. If the highest bit is not set then the reset was not a power on reset, and the flag will indicate 'N'. Reasons for this include:

1. The processor did not power all the way down because power was being supplied to the processor through either the port pins, IRQ line, RESET line, or power pins.
2. The voltage driven on the power pin of the processor did not go below 0.1 volts.
3. The processor was not reset properly. Check the "Target Hardware Type". If you are connecting to a class II board, check the "MON08 cable communication connections type" in the "advanced settings" dialog.

6 – ROM is accessible (un-secured):

If the device properly entered monitor mode (4), the software reads locations \$FFF6-\$FFFF to determine if the processor passes the security check. Memory locations which are invalid or protected read back from the device as \$AD. If all bytes from \$FFF6-\$FFFF read a value of \$AD, it is assumed the device is secure, and the flag value is an 'N'. If all flags 0-5 register a value of 'Y' and flag 6 register a value of 'N,' then the reset process has gone correctly except that the security code used to pass security was incorrect. Specify the correct security code and try again, or IGNORE the security failure and erase the device. Once you erase a secured device, you must exit the software and restart it in order to pass security.

7.3.6 MultilinkCyclonePro Menu Description

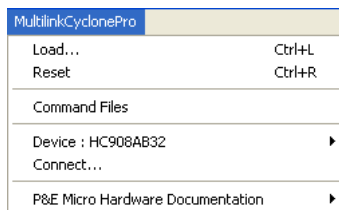


Figure 7-19: MultilinkCyclonePro Menu

7.3.6.1 Device

The “Device” option in the MultilinkCyclonePro menu allows the user to select the particular Freescale processor that they wish to use. When choosing Device from the MultilinkCyclonePro menu, additional boxes will open which allow you to select the family (e.g. KX Family), and device type (e.g. HC908KX2) of the MCU that you are using.

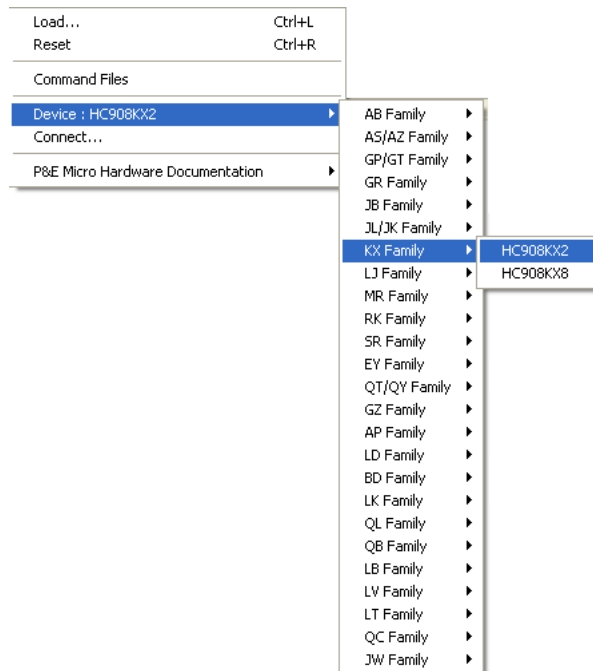


Figure 7-20: HC08 Device Menu

7.3.6.2 Connect

The “Connect” option initiates an attempt to communicate with the device chosen under the device section of the menu.

7.3.7 Active Mode Menu Options

When the microprocessor is connected, more menu entries become available to the user.

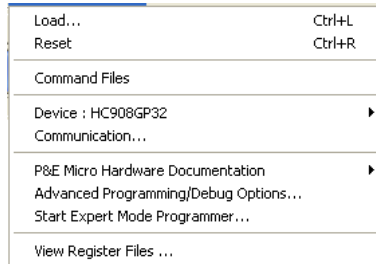


Figure 7-21: Additional Menu Options

7.3.7.1 Start Advanced Programming/Debug Options

The Start Advanced Programming/Debug Options menu entry take user to the Advanced Options dialog, where he can configure the software settings for the FLASH programming procedure.

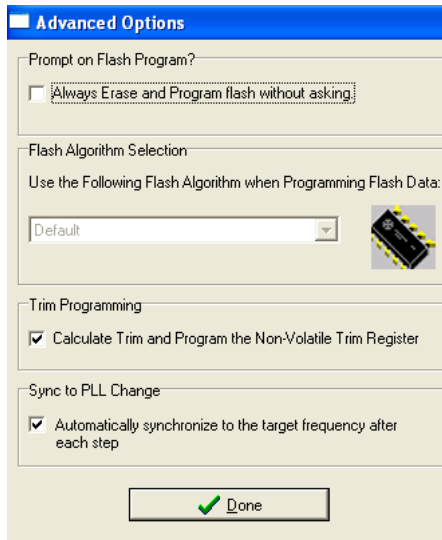


Figure 7-22: Advanced Programming Procedure

Prompt on Flash Program

Checking “Always Erase and Program flash without asking” lets the software

transparently program the microprocessor.

Trim Programming

The “Trim Programming” checkbox enables automatic calculation and programming of the trim value in a designated Non-Volatile memory location.

Sync to PLL Change

“Sync to PLL Change” is required for the software/hardware connection to synchronize with the microprocessor during the flash erasing/programming procedure.

Start Expert Mode Programmer

“Start Expert Mode Programmer” grants the user access to P&E’s graphical flash programming utility, PROG08SZ. PROG08SZ lets an advanced user control the step-by-step execution of the flash erase/programming procedure. See **Figure 7-23**. More information on how to use PROG08SZ can be found on P&E Microcomputer Systems website at: www.pemicro.com.

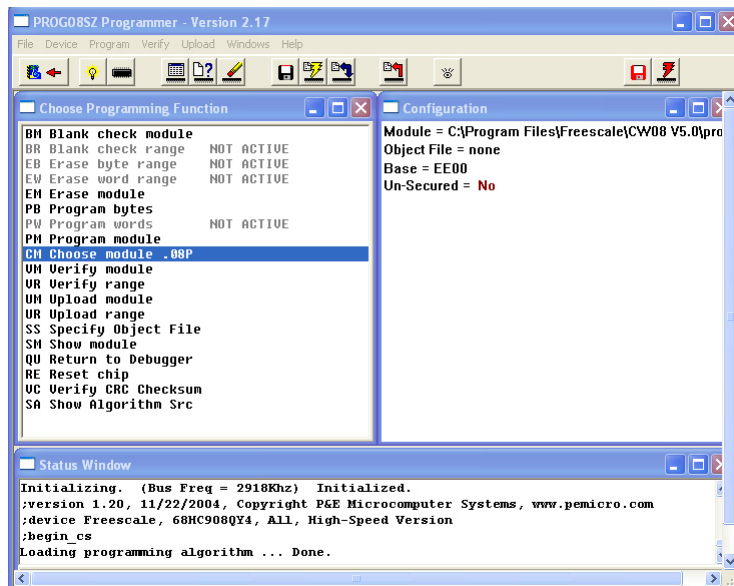


Figure 7-23: PROG08SZ Software

7.3.7.2 View Register Files

The “View Register Files...” menu selection also gives the user the option of running the register file viewer/editor. If register files are available for the device that you have chosen, the “Choose a Register Block” window (see **Figure 7-24**) will be opened. You may also open it by entering the R command in the Command Window command line.

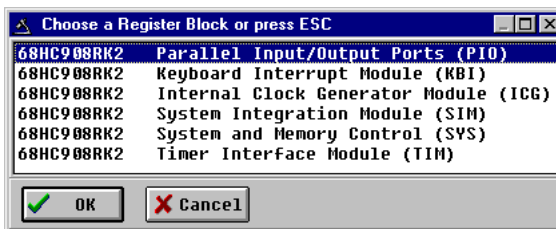


Figure 7-24: Choose A Register Block Window

If register files have been installed on the host computer, selecting a block brings up the Register Block register listing (see **Figure 7-25**), which shows a list of the associated registers, their addresses, and their descriptions. This begins interactive setup of system registers such as I/O, timer, and COP watchdog.

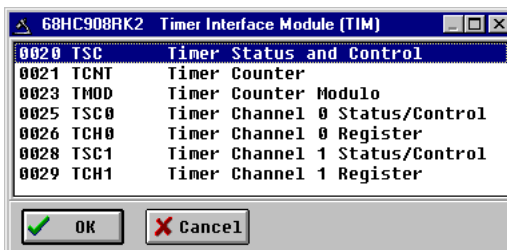


Figure 7-25: Register Block Register Listing

Selecting a file brings up the Register Window (see **Figure 7-26**), which displays the values and significance for each bit in the register. The registers can be viewed and their values modified, and the values can be stored back

into debugger memory.

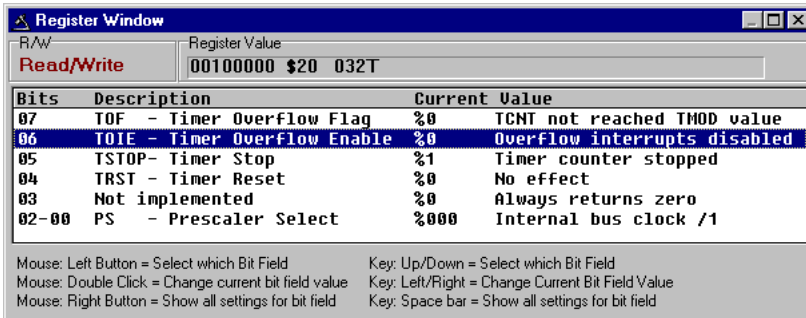


Figure 7-26: Register Window

7.3.8 Debugging Limitations

The following limitations are inherent in MON08 debugging and should be observed carefully.

- Do not step a command that branches to itself.
- Do not step an SWI (software interrupt) command.
- The hardware breakpoint registers are reserved for use by the debugger. Attempting to use these registers for other purposes may not work.
- Be careful about showing peripheral status and data registers in the memory or variables window. A refresh of the window will read these registers and may cause the clearing of flags.
- The debug monitor built into CPU08 processors uses up to 13 bytes of the stack. Do not write to these addresses from (SP-13) to SP. To load a program into RAM, move the stack to the end of RAM.
- If interrupts are turned on during stepping, the debugger will not step into the interrupt. Instead, it will execute the whole interrupt and stop on the command returned to after the interrupt.
- Do not set hardware breakpoints within the monitor ROM area itself, or they will not function properly.

7.3.9 Debugging Tips

The following tips may prove useful:

- Single stepping is allowed in both RAM or ROM.
- The first breakpoint set is always a hardware breakpoint, and any additional breakpoints set are software breakpoints. To make sure that a hardware breakpoint is being set, make sure only one breakpoint is being used.
- Hardware breakpoints will stop execution in ROM or RAM. Software breakpoints will stop execution only in RAM.
- Experiment with the register interpreter. Use the “View Register Files...” option in the MultilinkCyclonePRO menu.
- Executing an SWI instruction while running is functionally equivalent to hitting a breakpoint, except that execution stops at the instruction following the SWI.
- A hardware breakpoint may be used to trap a data read/write to anywhere in the memory map. The debugger stops at the instruction after the one that accesses the data location.
- When the target board is reset by the debugger, power to the microcontroller may be turned off for a short duration. Although much of RAM may look the same, some values may have changed.

7.3.10 Class 5-8 Device Description

7.3.10.1 Class 5

P&E’s MON08 Cyclone Device connects to a PC via the serial port. The Cyclone-to-MCU connection occurs via a standard 16-pin MON08 ribbon cable. The MON08 Cyclone Device allows the user to auto-detect the baud rate as well as auto-cycle the power through the MCU system.

7.3.10.2 Class 7

P&E MON08 Multilink and USB MON08 Multilink cables connect to the user’s MCU system via a standard 16-pin MON08 ribbon cable. MON08 Multilink and USB MON08 Multilink allow one to auto-detect the baud rate as well as auto-cycle the power through the microprocessor system.

7.3.10.3 Class 8

P&E’s Cyclone Pro communicates with the PC through a Serial, Ethernet or USB port. The Cyclone Pro can be used to debug and program the firmware

inside of Freescale HC08 microprocessors via a standard 16 pin MON08 ribbon cable. The Cyclone Pro can provide its own power and clock signals to the microprocessor, as long as proper signals are connected to the corresponding pins of the 16 pin MON08 header. In addition the Cyclone PRO can be used for programming and debugging the ColdFire V1/HC(S)12(X)/HCS08/RS08 Freescale microprocessors via a standard 6 pin ribbon cable. To take advantage of this functionality, one should connect proper signals to the standard 6 pin Background Debug Module header.

7.4 HCS08 P&E Multilink/Cyclone PRO Connections

7.4.1 Introduction

The HCS08 P&E Multilink/Cyclone Pro Connection setting permits a connection to Multilink/Cyclone PRO devices. HCS08 P&E Multilink/Cyclone Pro Connection Interface mode allows the user to debug code, as the firmware is fully resident in the FLASH of the microprocessor. The operation of all modules fully reflects the actual operation of the on-board resources.

7.4.2 Connection Procedure

1. Choose the P&E Multilink/Cyclone Pro option from the set connection dialog: See **Figure 7-27**.

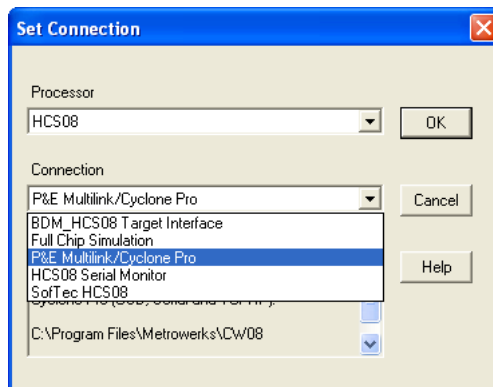


Figure 7-27: HCS08 P&E Multilink/Cyclone Pro Connection Setting

2. Choose the P&E device that you are using from the Interface drop-

down menu and click on Refresh. See **Figure 7-28** and **Figure 7-29**.

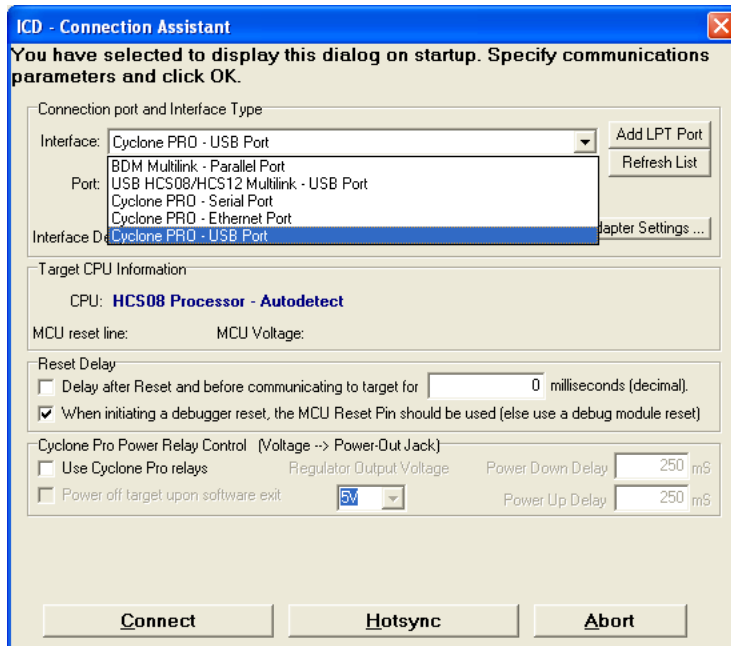


Figure 7-28: HCS08 Connection Assistant Interface Selection

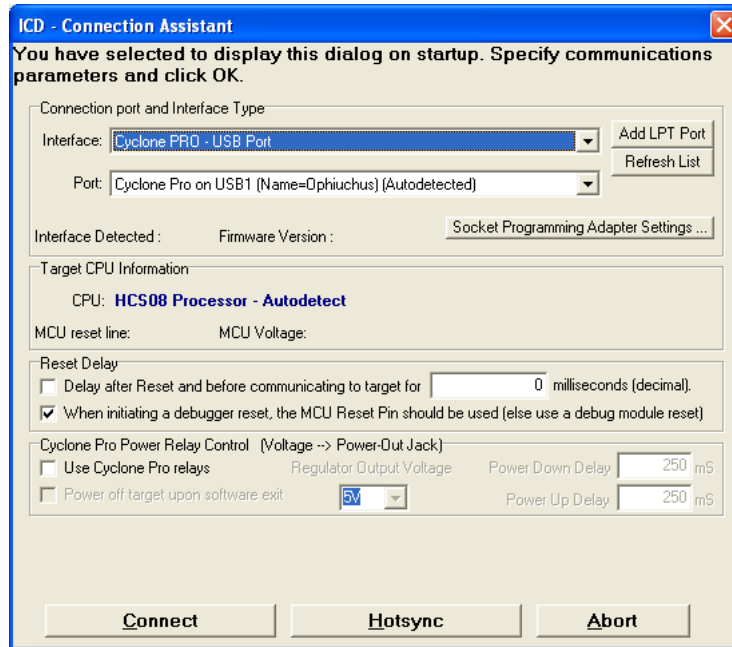


Figure 7-29: HCS08 Connection Assistant Interface Selected

The “Hotsync” button in the Connection Assistant (see **Figure 7-29**) allows the user to connect to an already running target.

7.4.3 MultilinkCyclonePro Menu Description

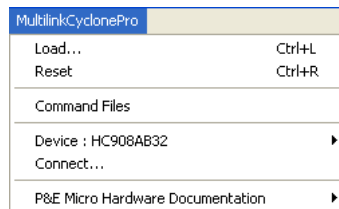


Figure 7-30: MultilinkCyclonePro Menu

7.4.3.1 Device Selection

The “Device” option in the MultilinkCyclonePro menu allows the user to select

the particular Freescale processor that they wish to use. When choosing Device from the MultilinkCyclonePro menu, additional boxes will open which allow you to select the family (e.g. GB Family), and device type (e.g. 9S0GB60) of the MCU that you are using.

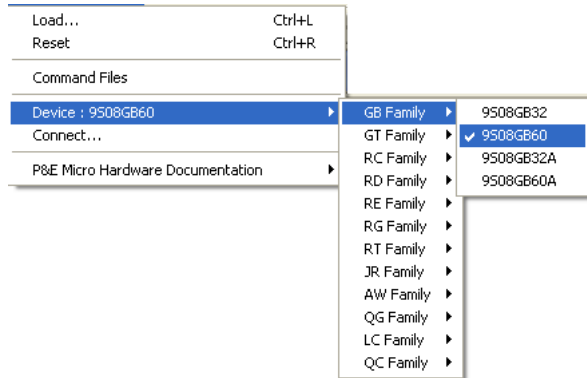


Figure 7-31: HCS08 Device Menu

7.4.3.2 Connect

The “Connect” option initiates an attempt to communicate with the device chosen under the device section of the menu.

7.4.4 Active Mode Menu Options

When the microprocessor is connected, more menu entries become available to the user.

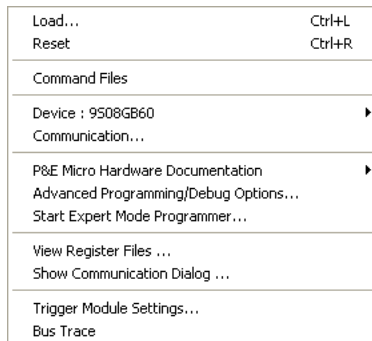


Figure 7-32: Active Mode Menu

7.4.4.1 Advanced Programming/Debug Options

The Start Advanced Programming/Debug Options menu entry take user to the Advanced Options dialog, where he can configure the software settings for the FLASH programming procedure.

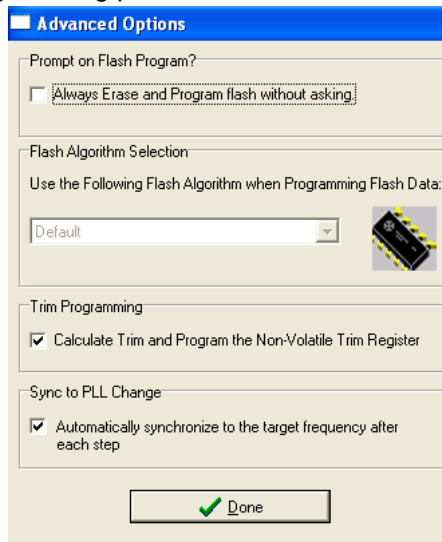


Figure 7-33: Advanced Programming Procedure

Prompt on Flash Program

Checking “Always Erase and Program flash without asking” lets the software transparently program the microprocessor.

Trim Programming

The “Trim Programming” checkbox enables automatic calculation and programming of the trim value in a designated Non-Volatile memory location.

Sync to PLL Change

“Sync to PLL Change” is required for the software/hardware connection to synchronize with the microprocessor during the flash erasing/programming procedure.

Start Expert Mode Programmer

Start Expert Mode Programmer grants to the user access to P&E’s graphical flash programming utility, PROGHCS08. PROGHCS08 lets an advanced user control the step-by-step execution of the flash erase/programming procedure. See **Figure 7-34**. More information on how to use the PROGHCS08 can be found on the P&E Microcomputer Systems website at www.pemicro.com.

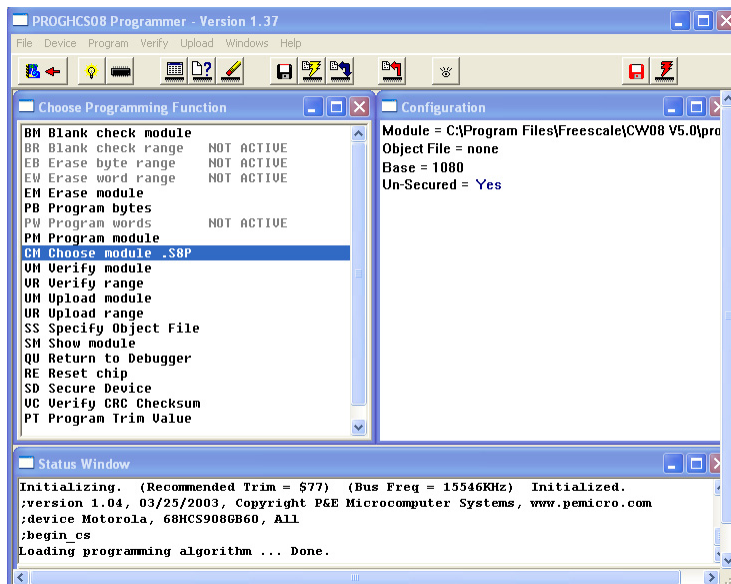


Figure 7-34: PROGHCS08 Software

7.4.4.2 View Register Files

The “View Register Files...” menu selection also gives the user the option of running the register file viewer/editor. If register files are available for the device that you have chosen, the “Choose a Register Block” window (see **Figure 7-35**) will be opened. You may also open it by entering the R command in the Command Window command line.

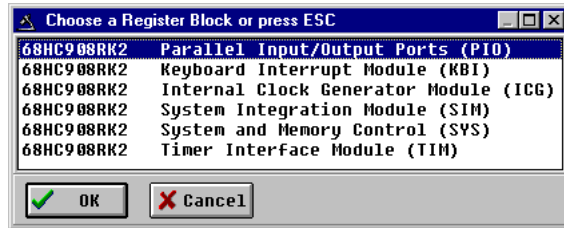


Figure 7-35: Choose A Register Block Window

If register files have been installed on the host computer, selecting a block brings up the Register Block register listing (see **Figure 7-36**), which shows a list of the associated registers, their addresses, and their descriptions. This begins interactive setup of system registers such as I/O, timer, and COP watchdog.

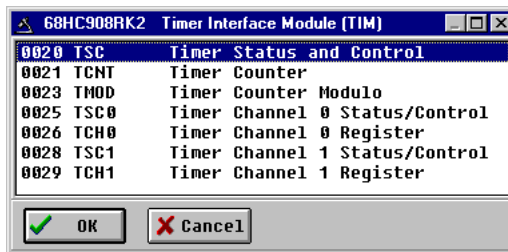


Figure 7-36: Register Block Register Listing

Selecting a file brings up the Register Window (see **Figure 7-37**), which displays the values and significance for each bit in the register. The registers can be viewed and their values modified, and the values can be stored back

into debugger memory.

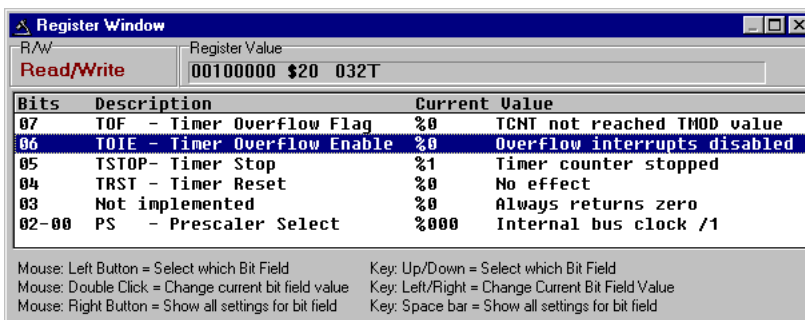


Figure 7-37: Register Window

7.4.4.3 Trigger Module Settings

For more information about the “Trigger Module Settings...” menu option, please see the CodeWarrior User Manual.

7.4.4.4 Bus Trace

For more information about the “Bus Trace” menu option, please see the CodeWarrior User Manual.

8 ETHERNET CONFIGURATION

This section describes the mechanism used by the Cyclone PRO device to transact data over an Ethernet network. It primarily focuses on the User Datagram Protocol (UDP), which is a popular method for sending data over a network when the speed of a data transaction is of more concern than the guarantee of its delivery. The Cyclone takes advantage of the UDP protocol's penchant for speed, and adds an extra layer of logic to guarantee the delivery of UDP packets in order to offer a best-of-both-worlds solution.

8.1 Network Architectures

Before delving into the innards of Ethernet message passing, it is prudent to briefly describe the different network architectures in use today, and how they pertain to the operation of the Cyclone. Computers are, of course, connected to one another through intermediary devices in order to form networks. There are several classes of these intermediary devices, but they generally fall into one of the following three groups:

Hubs

At the most basic level, computers are connected to one another through a Hub. A Hub is a device with several ports that are used to connect multiple computers together. It is a repeater device – a Hub simply copies the data incoming on one port as data outgoing on the other ports. In this manner, if there are four computers connected through a Hub, and if the first computer is sending data to the second computer, then the third and the fourth computers will also receive an identical copy of that data. Hubs are usually used to set up a small Local Area Network (LAN), which may have on the order of 10 to 20 computers.

Switches

The aforementioned type of process, where the data is simply replicated onto every available port, quickly becomes inefficient for larger sized networks. For this reason, a larger sized LAN employs the usage of Switches instead of Hubs. A Switch is essentially a smart Hub, in that it limits the input and output of data to the two transacting computers.

Routers

Larger networks, such as Wide Area Networks (WANs), or the Internet for that matter, use progressively more sophisticated devices to transact data. At the core of these devices is the Router, which functions as a switch between networks.

The Cyclone performs irrespective of the connection mechanism, with one very important caveat: it needs to be set up with the appropriate network parameters for the underlying network architecture.

8.2 Network Parameters

A typical network becomes operational not after the physical connections have been established, but after network parameters in the form of IP (Internet Protocol) numbers have been assigned to the individual computers. An IP number is a unique string that consists of four numbers ranging between 0 and 255, separated by dots, e.g., 192.168.1.2. Every computer that is on a network needs to have a unique IP number. The computer uses this IP number to identify itself on the network, and also to address the recipient of its data.

Assignment of this IP number is sufficient information to transact data on a simple network connected by a hub. On a more complex network, however, routing information becomes important. The routing information consists of two more IP numbers. The first of these is called the Subnet Mask, and is used to determine whether or not the destination address resides on the same subnet (i.e., doesn't need to be forwarded to another network). The other IP number is the Gateway Address, which is the address of the computer that handles forwarding and receiving of packets to and from other networks.

Before first use, the Cyclone needs to be programmed with a unique IP number, the Subnet Mask IP number, and also the default Gateway's IP number. This can be done via the USB or the Serial port, and is described in

greater detail in the “Configuring the Cyclone” section of this manual.

8.3 Internet Protocol

Once the network has been established, and the IP numbers have been assigned, data can be transacted over a network with one of several protocols. By far the most prevalent protocol is the Transmission Control Protocol (TCP), which runs on top of the Internet Protocol in what is collectively known as the TCP/IP protocol. The TCP/IP protocol was developed by the Department of Defense to connect different computers from different vendors by a “network of networks,” which has become what is known as the Internet today.

The primary purpose of the TCP/IP protocol was to prevent a complete network outage in the case of a nuclear attack, by automatically rerouting data traffic through the functioning part of the network. As such, the TCP/IP mechanism guaranteed delivery of data packets by introducing a system of acknowledgements and sequence numbers for the data packets. This mechanism, while good for transacting large amounts of data (such as email or file transfers), is unsuitable in the real-time type environment in which the Cyclone operates. Because the Cyclone needs to transact data as quickly as possible to the target, it takes advantage of TCP/IP’s alternative, the UDP/IP protocol.

Unlike TCP/IP, the UDP/IP protocol is a connectionless, single-packet protocol that sends short data packets at the expense of not guaranteeing their delivery. This makes the UDP/IP protocol efficient in real-time applications such as broadcasting video over the Internet, where the occasional loss of a frame of data is not going to hamper the overall viewing experience. Left unmodified, the UDP/IP, with its lack of guarantees for packet delivery, would be unusable in an environment where the delivery of a single byte of data needs to be guaranteed. The Cyclone firmware adds mechanisms to the UDP/IP protocol, without affecting its underlying efficiency, to guarantee delivery of data packets.

8.4 Connecting The Cyclone Device

There are two methods for establishing a connection between a Cyclone and a PC with an Ethernet cable. The most basic method is to connect the Cyclone directly to a PC, via a cross-over Ethernet cable. However, the more common method is to place the Cyclone and the PC on the same network

through a Hub.

8.4.1 Connecting the Cyclone to the PC over a network:

The Cyclone was intended for use on a network of multiple computers (and other Cyclones). There are many possible network configurations, and to describe them all is beyond the scope of this document. However, most configurations are a modification of a basic theme, which is that of connecting one or more PCs through a Hub to one or more Cyclones.

In order to connect these devices to the Hub, you will need to use the provided straight-through Ethernet cable. The straight-through cable, which is the “standard” Ethernet cable, is used to connect devices of different types together, such as a PC to a Hub, or a Hub to a Cyclone.

At this point it once again becomes necessary to program the Cyclone with valid IP numbers, the process for which is described in greater detail in the following section. However, it is important for the Cyclone and the PCs to have matching Subnet and Gateway IP numbers, and for each to have a unique IP number on the network. An example of a setting for above is as follows:

	<u>IP Number</u>	<u>Gateway IP</u>	<u>Subnet Mask</u>
PC1	192.168.100.1	192.168.100.3	255.255.255.0
PC2	192.168.100.2	192.168.100.3	255.255.255.0
CYCLONE	192.168.100.4	192.168.100.3	255.255.255.0
Gateway	192.168.100.3	192.168.100.3	255.255.255.0

It is important to briefly touch upon the underlying network architecture, which can be a 10Mb (Megabit), 100Mb, 10/100Mb, half-duplex, or a full-duplex connection. The details of the underlying network architecture are beyond the scope of this document, but it is sufficient to note that most modern network cards, as well as the Cyclone device, have the capability to configure themselves for the underlying network through the Auto-negotiation mechanism. Auto-negotiation is performed as soon as a network cable is connected to the device, and it sets the operating parameters of the device to match those of the network.

8.4.2 Connecting Cyclone-to-PC via an Ethernet cable

In order to connect the Cyclone to a PC directly via an Ethernet cable, you need to use what is known as a cross-over cable. A cross-over cable, which is

not provided by P&E, is normally used to connect two similar devices such as a PC to a PC, or a Hub to a Hub. It is a cable that has its receive and transmit wires crossed over so that the similar devices can effectively communicate with one another.

With this configuration, it is still important to assign IP numbers to both the PC and the Cyclone device. Although at first glance it may not seem necessary to assign a Gateway address in this configuration, the Cyclone was designed to operate on a network of more than two computers, and therefore it needs to be programmed with a Gateway address.

Assuming the desktop's IP number to be 192.168.100.1, this is an example of the three IP numbers that would need to be programmed into the Cyclone:

	<u>IP Number</u>	<u>Gateway IP</u>	<u>Subnet Mask</u>
PC	192.168.100.1	none	255.255.255.0
CYCLONE	192.168.100.2	192.168.100.1	255.255.255.0

For more information on programming these IP numbers into the Cyclone device, please see the following section.

8.5 Cyclone IP Setup Via LCD Menu

For instructions on how to configure the Cyclone PRO using the LCD Menu, please see **Section 4.2 - Operation Via LCD Menu (Rev. C Only)**.

8.6 Cyclone IP Setup Utility User Interface (ConfigureIP)

Before the Cyclone device transacts data on an Ethernet network, it will need to be configured with the relevant network parameters. The application that provides this capability is the Cyclone IP Setup Utility (ConfigureIP), which can be found as part of the distribution software.

This utility is used to configure the Cyclone PRO with network parameters,

and also to update the firmware of the Cyclone PRO.

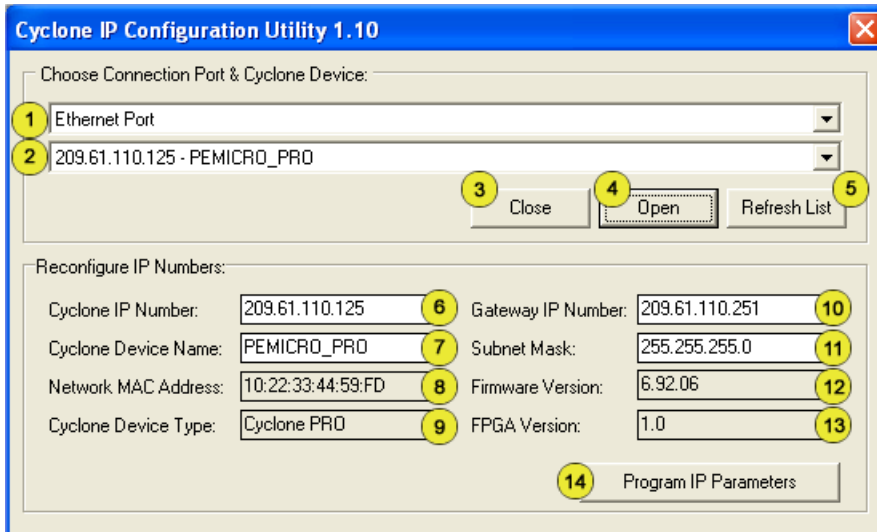


Figure 8-1: IPSetup.exe Default Screen

(1) Drop-down Box 1

There are three options available in this drop-down box, of which “Ethernet Port” is displayed. The other options are “Serial Port” and “USB Port”. Changing to any one of the three Ports will list the devices which are found over that specific Port.

(2) Drop-down Box 2

Once one of the three (Serial, USB, or Ethernet) communication interfaces has been selected in the first drop-down box, a list of all available Cyclone devices over that interface will be displayed for selection.

(3) Close Button

The “Close” button is active only when a device has been opened for access. Once a device has been opened for access, it needs to be closed before another device can be opened for access.

(4) Open Button

The “Open” button opens a device for access. This is a required step before changing the parameters on the selected device. Once a device

has been selected through the second drop-down box and is opened for access, its information will be displayed at the bottom of the dialog box.

(5) Refresh List

Will refresh the dialog boxes by searching for devices which are currently connected via the Serial or USB interfaces, or are found on the network.

(6) Cyclone IP Number

This is the IP number which will be associated with the Cyclone PRO. It needs to be a unique IP number which can be accessible on the network.

(7) Cyclone Device Name

This is a label which can be used to identify the Cyclone PRO by name, e.g., "John's Cyclone" or "Manufacturing Floor."

(8) MAC Address

This is the Media Access Control address, the unique number of an Ethernet device on the network. This is programmed by P&E and cannot be modified.

(9) Cyclone Device Type

This displays the type of Cyclone hardware.

(10) Gateway IP Number

The IP number of a gateway on the network.

(11) Subnet Mask

The subnet mask of the network.

(12) Firmware Version

A read-only field which returns information pertaining to the build date and firmware version of the Cyclone PRO device.

(13) FPGA Version

A read-only field which returns the hardware version of the FPGA.

(14) Program Cyclone Parameters

This button saves the information as it appears in the "Reconfigure IP Numbers" area onto the Cyclone device.

8.7 Using ConfigureIP.exe To Configure The Cyclone PRO

Before the Cyclone PRO is ready to communicate over an Ethernet network,

it will need to be configured with the relevant network parameters. The application that provides this capability is the Cyclone Configuration Utility (IPSetup.exe), and is provided as part of the standard Cyclone PRO software distribution.

In order to update the network parameters, perform the following steps:

1. Connect a Cyclone PRO to the PC via a serial or a USB cable, and make sure that it is powered before launching the Cyclone Configuration Utility. The Cyclone PRO Configuration Utility starts up with the following screen:

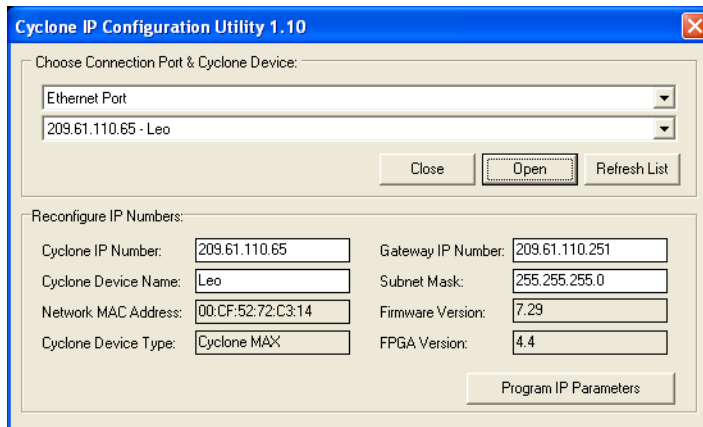


Figure 8-2: Cyclone IP Setup Utility - Initial Screen

2. Assuming that the Cyclone PRO is connected to the COM1 serial port of the PC, switch from “Ethernet Port” to “Serial Port”, at which point the second drop-down box will display COM1. Click “Open” to get a dialog box similar to the following:

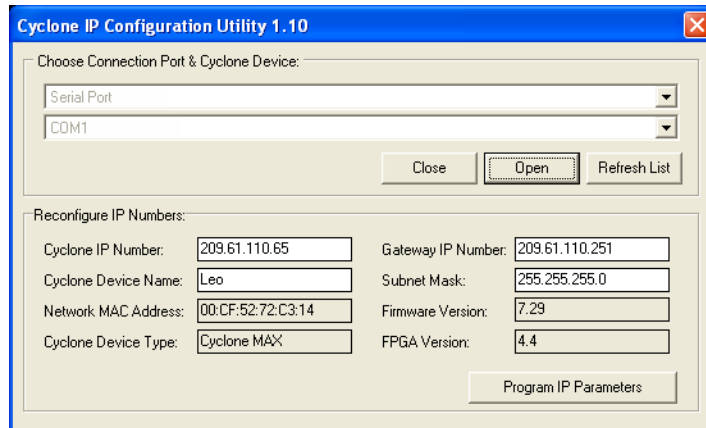


Figure 8-3: Cyclone IP Setup Utility - Continue Setup

3. The Cyclone PRO now needs to be programmed with IP numbers for the network on which it will operate. The Cyclone IP Number field must contain a unique IP number.

8.8 Managing Multiple Images

Once the programming images have been created and saved to the disk using the Create Image utility, they may then be loaded collectively onto the Cyclone. The Manage Images utility, shown in **Figure 8-4**, allows the Cyclone to store and manage multiple images in the Cyclone’s internal memory. If you have purchased a CompactFlash activation license it will also allow you to store and manage multiple images on any compatible CompactFlash cards that are loaded into the Cyclone’s CompactFlash port.

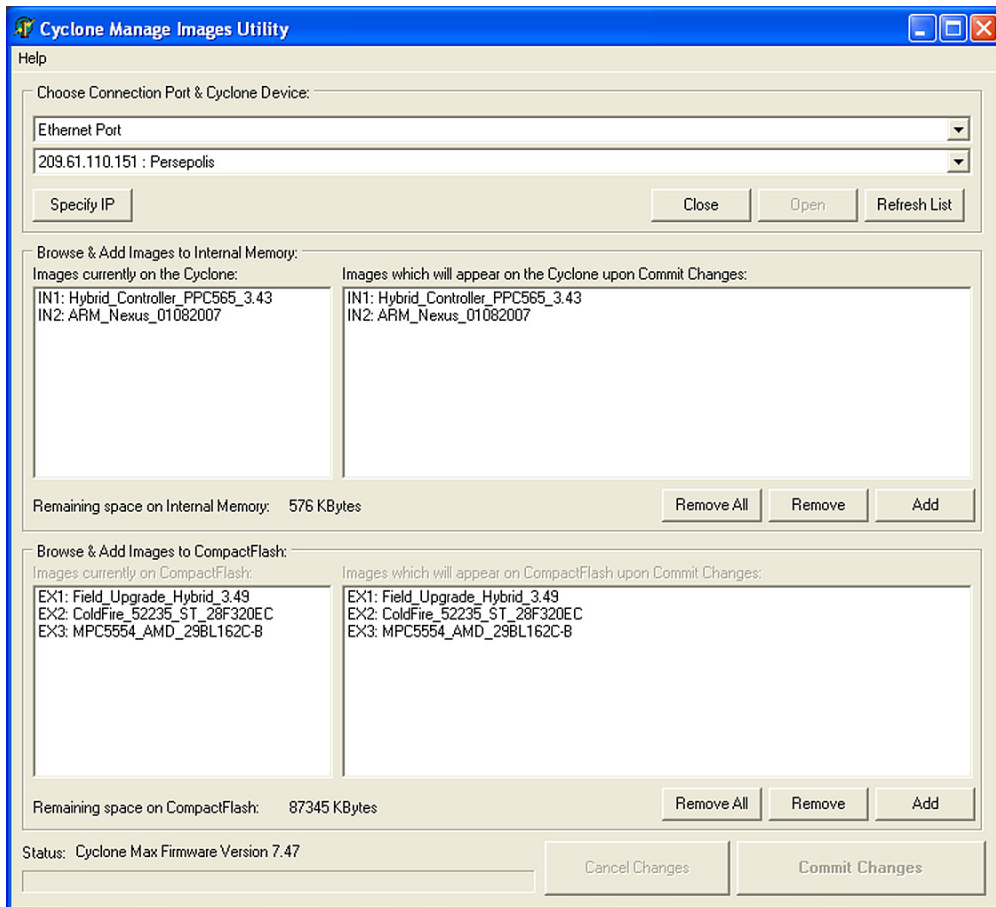


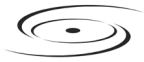
Figure 8-4: Manage Images Utility

Upon opening a selected Cyclone PRO, the user is provided in the top left panel with a list of the images currently on the unit's internal memory. If the CompactFlash license has been activated, a list of images on any connected CompactFlash card will also be displayed in the bottom left panel. The panels to the right can be used to add or delete additional images by using the Add and Remove buttons beneath each panel.

Once the images that you wish to load appear in the panels to the right, you

must press “Commit Changes” to load the Cyclone accordingly. No changes made to the Cyclone PRO until the “Commit Changes” button is pressed.

Note: Any images that are already stored on the Cyclone PRO or CompactFlash can only be removed by using the “Remove All” button.

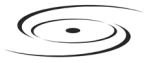


9 SERIAL PORT CONFIGURATION

Standard serial cables may be used for serial port Cyclone PRO configuration.

10 USB PORT CONFIGURATION

Standard USB cables may be used for USB port Cyclone PRO configuration. The user may use network hubs as necessary.



CYCLONE PRO





— *Freescale Controller Continuum* —

...68HC08/S08/RS08/(S)12(X) ... ColdFire® V1 ... ColdFire® V2/V3/V4 ... PowerPC® ... Nexus® ... ARM® ...



Cyclone PRO



Cyclone MAX

©2008, P&E Microcomputer Systems, Inc. All rights reserved.

P&E Microcomputer Systems, Inc.
P.O. Box 2044
Woburn, MA 01888

www.pemicro.com

Freescale™ and the Freescale logo are trademarks of Freescale Semiconductor, Inc.
All other product or service names are the property of their respective owners.
© Freescale Semiconductor, Inc. 2006.