

# **M68ICS08MR**

## **In-Circuit Simulator**

**User's Manual**



## Important Notice to Users

While every effort has been made to ensure the accuracy of all information in this document, Motorola assumes no liability to any party for any loss or damage caused by errors or omissions or by statements of any kind in this document, its updates, supplements, or special editions, whether such errors are omissions or statements resulting from negligence, accident, or any other cause. Motorola further assumes no liability arising out of the application or use of any information, product, or system described herein: nor any liability for incidental or consequential damages arising from the use of this document. Motorola disclaims all warranties regarding the information contained herein, whether expressed, implied, or statutory, *including implied warranties of merchantability or fitness for a particular purpose*. Motorola makes no representation that the interconnection of products in the manner described herein will not infringe on existing or future patent rights, nor do the descriptions contained herein imply the granting or license to make, use or sell equipment constructed in accordance with this description.

## Trademarks

This document includes these trademarks:

Motorola and the Motorola logo are registered trademarks of Motorola, Inc.

Windows and Windows 95 are registered trademarks of Microsoft Corporation in the U.S. and other countries.

Intel is a registered trademark of Intel Corporation.

Motorola, Inc., is an Equal Opportunity / Affirmative Action Employer.

## **Table of Contents**

### **Section 1. General Information**

1.1	Introduction.....	9
1.2	MRICS Components.....	12
1.2.1	M68ICS08MR Connector Components.....	13
1.2.2	MRICS Interface Software.....	14
1.3	Hardware and Software Requirements.....	14
1.4	Specifications.....	15
1.5	About This Manual.....	15
1.6	Customer Support.....	15

### **Section 2. Preparation and Installation**

2.1	Introduction.....	17
2.2	Hardware Preparation.....	17
2.2.1	MRICS Limitations.....	17
2.2.2	Configuring MRICS Jumper Headers.....	18
2.2.3	Target Interface Connection Options.....	25
2.2.4	Host Computer (PC) — MRICS Serial Interconnection (J12) ...	34
2.2.5	J4 Pin Assignments; +5Vdc Power Connector.....	34
2.3	Connecting the MRICS.....	35
2.4	Connecting the MRICS to a Target System.....	36
2.5	Installing the Software.....	36

### **Section 3. Support Information**

3.1	Introduction.....	37
3.2	MRICS Connector Signal Definitions.....	37

3.3	Target-Cable Pin Assignments . . . . .	52
3.4	Parts List . . . . .	57
3.5	MRICS Printed Circuit Board Layout and Schematic Diagrams . . . .	60

## **Section 4. Using the MON08 Interface**

4.1	Introduction. . . . .	71
4.2	Target System Header Placement . . . . .	71
4.3	Target Requirements for Using MON08 . . . . .	74
4.4	Connecting to the In-Circuit Simulator . . . . .	74

## **User's Manual — Glossary**

## **List of Figures**

2-1	. MR16/32 Clock Source Selection Jumper (W1) . . . . .	21
2-2	. Reset Source Selection Jumper (W2) . . . . .	21
2-3	. MR16/32 I/O Baud Rate Selection Jumper (W3) . . . . .	22
2-4	. MCU Emulation Selection Jumper (W4) . . . . .	22
2-5	. MRICS XTAL Clock Enable Jumper (W5) . . . . .	23
2-6	. MR4/8 Clock Source Selection Jumper (W6) . . . . .	23
2-7	. Board Reset Selection Jumper (W7) . . . . .	24
2-8	. J2 Pin Assignments; MR16/32 Target Interface . . . . .	26
2-9	. J3 Pin Assignments; MR16/32 Target Interface . . . . .	27
2-10	. J5 Pin Assignments; MR4/8 Target Interface . . . . .	28
2-11	. J6 Pin Assignments; MR4/8 Target Interface . . . . .	29
2-12	. J7 Pin Assignments; MR16/32 MON08 Cable Connectors . . . . .	30
2-13	. J8 Pin Assignments; MR4/8 MON08 Cable Connectors . . . . .	31
2-14	. J1 Pin Assignments; MR16/32 Target Connector . . . . .	32
2-15	. J11 Pin Assignments; MR4/8 DIP Target Connector . . . . .	33
2-16	. J12 Pin Assignments; RS232 Connector . . . . .	34
2-17	. J4 Pin Assignment; +5Vdc Power Connector . . . . .	34



## **List of Tables**

<b>Table</b>	<b>Title</b>	<b>Page</b>
Table 1-1.	M68ICS08MR Product Components . . . . .	12
Table 1-2.	Hardware Components . . . . .	13
Table 1-3.	Software Components . . . . .	14
Table 1-4.	M68ICS08MR Board Specifications. . . . .	15
Table 2-1.	MRICS Jumper Header Description . . . . .	20
Table 2-2.	Cable/Connector Options for MCUs. . . . .	25
Table 3-1.	MR16/32 Target Connector J2 . . . . .	38
Table 3-2.	MR16/32 Target Connector J3 . . . . .	40
Table 3-3.	MR4/8 Target Connector J5 . . . . .	42
Table 3-4.	MR4/8 Target Connector J6 . . . . .	44
Table 3-5.	MR16/32 MON08 Connector J7 Pin Assignments . . . . .	46
Table 3-6.	MR4/8 MON08 Connector J8 Pin Assignments . . . . .	47
Table 3-7.	MR16/32 Target Connector J1 . . . . .	48
Table 3-8.	MR4/8 Target DIP Connector J11. . . . .	51
Table 3-9.	Power Connector J4 Pin Assignments. . . . .	52
Table 3-10.	RS-232C Communication Connector J12 Pin Assignments . . . . .	52
Table 3-11.	Flex Target Cable (M68CBL05C) for QFP Target Head Adapter B53	
Table 3-12.	Flex Target Cable (M68CBL05C) for DIP Target Head Adapter A54	
Table 3-13.	Flex Target Cable (M68CBL05C) for QFP Target Head Adapter B55	
Table 3-14.	Flex Target Cable (M68CBL05C) for DIP/QFP Target Head Adapter B56	
Table 3-15.	Bill of Materials. . . . .	57
Table 4-1.	MR16/32 MON08 Target System Connector J7. . . . .	72
Table 4-2.	MR4/8 MON08 Target System Connector J8. . . . .	73





# Section 1. General Information

## 1.1 Introduction

This manual provides information about the Motorola M68ICS08MR in-circuit simulator (MRICS).

The MRICS is a stand-alone development and debugging tool . It contains all of the hardware and software needed to develop and simulate source code, and to program this series of MCU devices:

- MC68HC908MR4
- MC68HC908MR8
- MC68HC908MR32

The MRICS and it's software comprise an editor, assembler, programmer, simulator, and limited real-time input/output emulator for the MCUs. When connected to a host PC (personal computer) and target hardware (your prototype product), actual inputs and outputs of the target system may be used during code simulation.

The MRICS can interface with any IBM<sup>®</sup> Windows 95<sup>®</sup>-based computer (or later version) through connection of a single RS-232 serial port using a DB-9 serial cable.

Connection to the target system is accomplished by a ribbon cable, a Motorola M6CLB05C flex cable, or a MONO8 cable. The ribbon cable or flex cable is used when an MCU is resident on the MRICS for emulation or simulation, and the MONO8 cable is used to debug or program a target system's MCU, directly, when the MCU resides on the target hardware.

The MRICS is a low-cost development system that supports editing, assembling, in-circuit simulation, in-circuit emulation, and FLASH memory programming. Its features include:

- Editing with WinIDE
- Assembling with CASM08W
- FLASH memory programming with PROG08SW
- In-circuit and stand-alone simulation of MC68HC908MR MCUs with ICS08MRW software, providing:
  - Simulation of all instructions, memory, and peripherals
  - Simulation of pin inputs from the target system
  - Installation of conditional breakpoints, script files, and logfiles
- Limited real-time emulation and debugging with ICD08SW, including:
  - Loading code into RAM
  - Executing real-time in RAM or FLASH
  - Placing one hardware breakpoint in FLASH
  - Placing multiple breakpoints in RAM
- On-line help documentation for all software
- Software integrated into the WinIDE environment, allowing function key access to all applications
- MON08 emulation connection to the target system allowing:
  - In-circuit emulation (limited)
  - In-circuit programming
- Four modes of operation:
  - Standalone — using the MRICS as a standalone system without a target board
  - Simulation — using the MRICS as an in-circuit simulator/emulator with a target cable
  - Evaluation - using the MRICS for real-time evaluation of the MCU and to debug user developed hardware and software
  - Programming — using the MRICS as a programmer

Functions of the MRICS vary depending upon which software you choose:

- With the WinIDE, CASM08W, editor, simulator, and assembler software - the function is as a limited real-time emulator.
- With the PROG08SW software - the function is to program MCU FLASH memory. (This function is not available on the MC68HC08MR4 MCU, which does not have FLASH). Only one part may be programmed at a time. The MRICS also supports in-circuit programming of either version of the part through either MON08 cable.
- With the ICS08MRZ simulation software, the MCU provides the required input/output information that lets the host computer simulate code, performing all functions except for maintaining port values. (The internal FLASH memory on the device is downloaded with a program that generates the appropriate port values.) The ICS08MRZ software on the host computer lets the host computer become a simulator.
- With using the ICD08SZ debugging software, code can be run directly out of the MCU's internal FLASH at real-time speeds.
- The MRICS board also provides +5 Vdc power, +8.6 Vdc power for the  $V_{TST}$  voltage required to enter monitor mode, either a 4.0000-MHz or 4.9152-MHz clock signal, and host PC RS-232 level translation, when it is supplied an active DTR signal.

**NOTE:** *The simulation speed will be slower than this rate because the host computer is the simulator.*

## 1.2 MRICS Components

The MRICS system includes the product components listed in **Table 1-1**.

**Table 1-1. M68ICS08MR Product Components**

<b>Part Number</b>	<b>Description</b>
ICS08MR	MRICS software development package
ICS08MRZ	MRICS simulator
ICD08SZ	MRICS debugger
MC68HC908MR32	MCU
MC68HC908MR8	MCU
M68CLB05C	Flex target cable
KRISTA 22-122	Serial cable
FRIWO 11.8999-P5	Power supply
01-RE91008WI	MON08 cable
M68DIP28S01C	28-pin SOIC adapter

## 1.2.1 M68ICS08MR Connector Components

**Table 1-2. Hardware Components**

Components	Description
XU1 XU2 XU4 XU3	Target sockets for the Motorola MC68HC908MR16/32 MCUs: 56-pin SDIP (dual-in-line package) 64-pin QFP (quad flat pack) Target sockets for the Motorola MC68HC(9)08MR4/8 MCUs: 28-pin DIP (dual-in-line package) 32-pin QFP (quad flat pack)
J2, J3	Two 2-row × 20-pin, 0.1-inch spacing connectors connect the MRICS to a target system using the M68CLB05C flex cable. Connectors J2 and J3 are used when emulating MC68HC908MR16/32 MCUs.
J4	+5 Vdc input voltage ( $V_{DD}$ )
J5, J6	Two 2-row × 20-pin, 0.1-inch spacing connectors connect the MRICS to a target system using the M68CLB05C flex cable. Connectors J5 and J6 are used when emulating MC68HC(9)08MR4/8 MCUs.
J1	One 60-pin, 0.3-inch spacing connector connect the MRICS to the target system using a ribbon cable. Connector J1 is used when emulating a MC68HC908MR16/32 MCU.
J7, J8	Two 2-row × 8-pin, 0.1-inch spacing connectors connect MON08 debug circuit to your remote target. Use connector J7 when emulating MC68HC908MR32/16 MCUs. Use connector J8 when emulating MC68HC(9)08MR4/8 MCUs.
J9, J10	Two 3-pin, 0.1-inch spacing connectors connect the MRICS clock circuit to your remote target. Connector J9 routes the 4.9152 MHz clock for the MC68HC908MR16/32 MCUs. Connector J10 routes the 4.000 MHz clock for the MC68HC(9)08MR4/8 MCUs.
J11	One 2-row × 14-pin, 0.3-inch spacing dual in-line package (28-pin DIP) socket connects the MRICS to the target system, using a ribbon cable, when emulating an MC68HC(9)08MR4/8 MCU.
J12	RS-232 connector interface to the host serial connector.

### 1.2.2 MRICS Interface Software

Windows-optimized software components, **Table 1-3**, are referred to, collectively, as "MRICS software" (part number ICS08MRZ). It is a product of *P&E Microcomputer Systems, Inc.*, and is included in the MRICS kit.

**Table 1-3. Software Components**

Components	Description
WINIDE.EXE	Integrated development environment (IDE) software interface for editing and performing software or in-circuit simulation
CASM08Z.EXE	CASM08Z command-line cross-assembler
ICS08MRZ.EXE	In-circuit/stand-alone simulator software for MC68HC908MR MCUs
PROG08SZ.EXE	FLASH memory programming software
ICD08SZ.EXE	In-circuit debugging software for limited, real-time emulation

### 1.3 Hardware and Software Requirements

The MRICS software requires an IBM-compatible host computer with this minimum hardware and software configuration:

- Windows 95 or later version operating system
- Approximately 2 Mbytes of available random-access memory (RAM) and 5 Mbytes of free disk space
- A serial port for communications between the MRICS and the host computer

## 1.4 Specifications

**Table 1-4. M68ICS08MR Board Specifications**

Characteristic	Specification
Temperature: Operating Storage	0° to 40°C -40° to +85°C
Relative humidity	0 to 95%, non-condensing
Power requirement	+5 Vdc, from included ac/dc adapter
Size	5" 8 "

## 1.5 About This Manual

The procedural instructions in this manual assume that the user is familiar with the Windows interface and selection procedures.

## 1.6 Customer Support

To obtain information about technical support or ordering parts, call the Motorola help desk at 800-521-6274.





## Section 2. Preparation and Installation

### 2.1 Introduction

This section provides information and instruction for configuring, installing, and readying the MRICS for use.

### 2.2 Hardware Preparation

These paragraphs explain:

- Limitations of the MRICS
- Configuration of the MRICS
- Installation of the MRICS
- Connection of the MRICS to a target system

**ESD CAUTION:** *Ordinary amounts of static electricity from clothing or the work environment can damage or degrade electronic devices and equipment. For example, the electronic components installed on the printed circuit board are extremely sensitive to electrostatic discharge (ESD). Wear a grounding wrist strap whenever handling any printed circuit board. This strap provides a conductive path for safely discharging static electricity to ground.*

#### 2.2.1 MRICS Limitations

This section describes system limitations of the MRICS.

##### 2.2.1.1 Bus Frequency

The MRICS communicates using the MON08 features. On the MR16/32 only, this forces the communication rate to  $f_{\text{bus}}/256$ . The bus frequencies are limited by standard baud rates allowed by the host software, or 9600 (and possibly 4800) baud for the MR16/32, and 9600 baud for the MR4/8.

### 2.2.1.2 MR16/32 Port A0

On the MR16/32, port A0 is used for communications, so it is unavailable for emulation.

### 2.2.1.3 MR4/8 Port B0 and Port B1

On the MR4/8, ports B0 and B1 are used for communications, so they are unavailable for emulation.

### 2.2.1.4 Low Voltage Interrupt (LVI)

The LVI is disabled by default in monitor mode. It is enabled by a dummy write to LVISR.

### 2.2.1.5 Internal Clock Generator (ICG)

The ICG is bypassed in monitor mode, so it is not available for use.

## 2.2.2 Configuring MRICS Jumper Headers

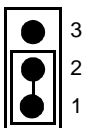
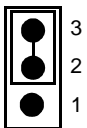
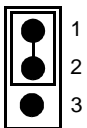
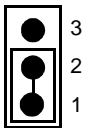
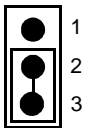
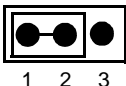
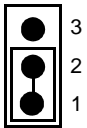
The MRICS supports four configuration options: standalone, simulation, evaluation, and programming.

- Standalone — ICS08MRZ.exe running on the host computer (the MRICS is not connected.) Emulation of the M68HC(9)08MR MCU CPU, registers, and I/O ports are done within the host computer environment.
- Simulation — Host computer connected to the MRICS via the RS-232 cable and ICS08MRZ.exe running on the host computer. This provides access to the M68HC(9)08MR MCU CPU, internal registers, and I/O ports.
- Evaluation — Host computer connected to the MRICS and the MRICS connected to the target system via the flex cable. This method provides limited real-time evaluation of the MCU and debugging user developed hardware and software.

- Programming — Host computer connected to the MRICS, and the MRICS connected to the target system via the MON08 cable. Use the PROG08SZ.exe to program the MCU FLASH module. In the programming mode there is limited evaluation (port A0 on the MR32 and port B0 and port B1 on the MR4/8 are used for communications, so they are unavailable for emulation).

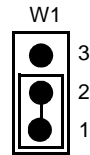
Seven jumper headers on the MRICS are used to configure the hardware options. **Table 2-1** is a quick reference to these optional settings, and subparagraphs **2.2.2.1** through **2.2.2.6** describe jumper header configuration in greater detail.

**Table 2-1. MRICS Jumper Header Description**

Jumper Header	Type	Description
W1 MR16/32 Target clock selection		Jumper on pins 1 and 2 (factory default) — Connects the MC68HC908MR16/32 clock to the target system via connectors J1 and J2. This position is labeled I. Jumper on pins 2 and 3 — Disconnects the MC68HC908MR16/32 clock from the target system.
W2 Target reset selection		Jumper on pins 1 and 2 — The MC68HC908MR MCU $\overline{\text{RESET}}$ signal initiates resets to the target system. This position is labeled I. Jumper on pins 2 and 3 (factory default) — The target-system $\overline{\text{RESET}}$ signal initiates resets to the MRICS on-board MC68HC908MR MCU. This position is labeled O.
W3 I/O Baud rate selection: MR16/32 only		Jumper on pins 1 and 2 (factory default) — MR16/32 I/O baud rate is set to 4800. This position is labeled 4800. Jumper on pins 2 and 3 — MR16/32 I/O baud rate is set to 9600. This position is labeled 9600. This jumper header does not affect the MR4/8, which operates at 9600 baud only
W4 Serial communication selection		Jumper on pins 1 and 2 (factory default) — MC68HC908MR16/32 MCU is installed. This position is labeled MR32/16. Jumper on pins 2 and 3 — MC68HC(9)08MR4/8 MCU is installed. This position is labeled MR8/4.
W5 XTAL clock enable		Jumper on pins 1 and 2 — MRICS XTAL clocks are disabled. Jumper header W5 has a cut-trace short between pins 1 and 2 so no jumper is required unless the cut-trace short is removed. This position is labeled O. Jumper on pins 2 and 3 (factory default) — MRICS XTAL clocks are enabled: 4.9152 MHz for the MC68HC908MR16/32 MCUs and 4.000 MHz for the MC68HC(9)08MR4/8 MCUs. This position is labeled I.
W6 MR4/8 target clock		Jumper on pins 1 and 2 (factory default) — Connects the MC68HC(9)08MR4/8 clock to the target system via connectors J6 and J11. This position is labeled I. Jumper on pins 2 and 3 — Disconnects the MC68HC(9)08MR4/8 clock from the target system. This position is labeled O.
W7 MRICS reset clock		Jumper on pins 1 and 2 (factory default) — Reset clock set at 4.9152 MHz. Use this setting when using the MC68HC908MR16/32 MCUs. This position is labeled MR32/16. Jumper on pins 2 and 3 — Reset clock set at 4.0000 MHz. Use this setting when using the MC68HC(9)08MR4/8 MCUs. This position is labeled MR8/4.

### 2.2.2.1 MR16/32 Clock Source Selection Jumper (W1)

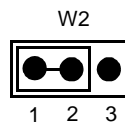
Use jumper header W1 (**Figure 2-1**) to connect the MC68HC908MR16/32 clock to the target system clock. Install a jumper on pins 1 and 2 (factory default) to drive the target system clock with the MC68HC908MR16/32 clock. Install a jumper on pins 2 and 3 to isolate the MC68HC908MR16/32 clock from the target system.



**Figure 2-1. MR16/32 Clock Source Selection Jumper (W1)**

### 2.2.2.2 Reset Source Selection Jumper (W2)

Use jumper header W2 (**Figure 2-2**) to select the target system or the MCU as the source for a system reset. Install a jumper on jumper header pins 1 and 2 (factory default) to drive the MCU  $\overline{\text{RESET}}$  signal to the target system. Install a jumper on jumper header pins 2 and 3 to reset the MCU whenever the target-system initiates a reset.

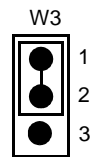


**Figure 2-2. Reset Source Selection Jumper (W2)**

### 2.2.2.3 I/O Baud Rate Selection Jumper (W3) — MR16/32 Only

There are two sockets for clocks on the MRICS board, XY1 and XY2. Socket XY1 is wired to drive the MR16/32 IC sockets and target connectors, and the clock is available for output on J10. Using the 4.9152-MHz clock as Y1 times the MR16/32 to communicate at 4800 or 9600 baud.

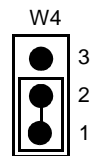
Use jumper W3 (**Figure 2-3**) to define the baud rate of the MR16/32. Install the jumper in jumper position 1-2 (labeled 9600 baud) to communicate at 9600 baud if a 4.9152-MHz clock is installed as Y1. Place the jumper in the 2-3 position (labeled 4800 baud) to communicate at 4800 baud. If Y1 is not 4.9152 MHz, the baud rate will be scaled proportionally.



**Figure 2-3. MR16/32 I/O Baud Rate Selection Jumper (W3)**

### 2.2.2.4 Serial Communication Selection Jumper (W4)

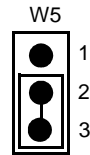
Install the jumper on jumper header W4 (**Figure 2-4**) pins 1 and 2 (factory default) when an MC68HC908MR16/32 MCU is installed. Install the jumper on pins 2 and 3 when an MC68HC(9)08MR4/8 MCU is installed.



**Figure 2-4. MCU Emulation Selection Jumper (W4)**

### 2.2.2.5 MRICS XTAL Clock Enable Jumper (W5)

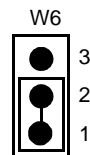
Install a jumper on jumper header W5 (**Figure 2-5**) pins 1 and 2 to disable the MRICS XTAL clock. Install a jumper on pins 2 and 3 (factory default) to enable the MRICS XTAL clock. The clock is 4.9152 MHz for the MC68HC908MR16/32 MCUs and 4.0000 MHz for the MC68HC(9)08MR4/8 MCUs.



**Figure 2-5. MRICS XTAL Clock Enable Jumper (W5)**

### 2.2.2.6 MR4/8 Clock Source Selection Jumper (W6)

Use jumper header W6 (**Figure 2-6**) to connect the MC68HC(9)08MR4/8 clock to the target system clock. Install a jumper on pins 1 and 2 (factory default) to drive the target system clock with the MC68HC(9)08MR4/8 clock. Install a jumper on pins 2 and 3 to isolate the MC68HC(9)08MR4/8 clock from the target system.



**Figure 2-6. MR4/8 Clock Source Selection Jumper (W6)**

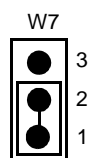
### 2.2.2.7 Board Reset Selection Jumper (W7)

When the power to the MCU is turned on, there is a delay of a few hundred milliseconds during which the  $\overline{\text{ICS\_RST}}$  signal is held low (set by U11).

When  $\overline{\text{ICS\_RST}}$  goes high, the binary ripple counter (U10) begins counting clock cycles. After 1024 clock cycles are counted, the DELAY\_RESET and DELAY\_RESET\_1 signals toggle and the counter stops counting. The delayed reset signals are used to turn on communications to the MCU and to hold several MCU pins in appropriate states to force monitor mode on power up.

Use jumper W7 (**Figure 2-7**) to set the ripple counter reset rate. Install the jumper on jumper header pins 1 and 2 (factory default) to select the MR16/32 clock, which causes Y1 to drive the counter at the factory installed rate of 4.9152 MHz. Use this setting when using the MC68HC908MR16/32 MCUs.

Install the jumper on jumper header pins 2 and 3 to select the MR4/8 clock. This sets Y2 to drive the counter at the factory installed rate of 4.0000 MHz. Use this setting when using the MC68HC(9)08MR4/8 MCUs.



**Figure 2-7. Board Reset Selection Jumper (W7)**



## 2.2.3 Target Interface Connection Options

There are three ways to connect the MRICS simulator board to your target system:

- Flex cable — low-noise target interface connection
- Ribbon cable — low-cost target interface connection
- MON08 cable — target interface connection with MCU FLASH programming and limited emulation

Below is a quick reference for defining the cable/connector setup to use with the appropriate MCU version. Refer to **2.2.3.1 MR16/32 Target Interface Connectors (J2 and J3)** through **2.2.3.5 MR4/8 DIP Target Connector (J11)**.

**Table 2-2. Cable/Connector Options for MCUs**

MCU	Flex Cable M68CLB05C	Ribbon Cable	MON08 Cable
MC68HC08MR4	J5 and J6	J11	J8
MC68HC908MR8	J5 and J6	J11	J8
MC68HC908MR16	J2 and J3	J1	J7
MC68HC908MR32	J2 and J3	J1	J7

### 2.2.3.1 MR16/32 Target Interface Connectors (J2 and J3)

Use connectors J2 (**Figure 2-8**), J3 (**Figure 2-9**), and Motorola's M68CLB05C flex cable, when emulating an MC68HC908MR16/32 MCU, to connect the MRICS to the target system. Connectors J2 and J3 are 40-pin shrouded headers (3M 2540-6002).

		J2			
MR32_PT B2	1	•	•	2	MR32_PT B3
MR32_PT B5	3	•	•	4	MR32_PT B6
PTC0	5	•	•	6	PTC1
Ground	7	•	•	8	No Connect
Ground	9	•	•	10	MR32_PTA7
MR32_PTA3	11	•	•	12	MR32_PTA4
TGT_PTA0	13	•	•	14	Ground
TGT_CLK	15	•	•	16	No Connect
$\overline{\text{TGT\_RST}}$	17	•	•	18	$\overline{\text{TGT\_IRQ}}$
PTF3	19	•	•	20	PTF2
PTF0	21	•	•	22	Ground
PTE6	23	•	•	24	PTE5
PTE3	25	•	•	26	PTE2
Ground	27	•	•	28	MR32_PWM6
MR32_PWM4	29	•	•	30	MR32_PWM3
PTD6	31	•	•	32	Ground
PTD2	33	•	•	34	PTD3
PTD1	35	•	•	36	No Connect
PTC5	37	•	•	38	Ground
TGT_PTC3	39	•	•	40	Ground

**Figure 2-8. J2 Pin Assignments; MR16/32 Target Interface**

**J3**

Ground	1	•	•	2	MR32_PTB4
MR32_PTB7	3	•	•	4	Ground
No Connect	5	•	•	6	No Connect
MR32_PTB0	7	•	•	8	MR32_PTB1
MR32_PTA5	9	•	•	10	MR32_PTA6
MR32_PTA1	11	•	•	12	MR32_PTA2
No Connect	13	•	•	14	No Connect
No Connect	15	•	•	16	No Connect
PTF5	17	•	•	18	PTF4
Ground	19	•	•	20	PTF1
No Connect	21	•	•	22	PTE7
PTE4	23	•	•	24	Ground
PTE1	25	•	•	26	PTE0
MR32_PWM5	27	•	•	28	No Connect
MR32_PWM2	29	•	•	30	MR32_PWM1
PTD4	31	•	•	32	PTD5
No Connect	33	•	•	34	No Connect
PTC6	35	•	•	36	PTD0
TGT_PTC4	37	•	•	38	Ground
TGT_PTC2	39	•	•	40	Ground

**Figure 2-9. J3 Pin Assignments; MR16/32 Target Interface**

## 2.2.3.2 MR4/8 Target Interface Connectors (J5 and J6)

Use connectors J5 (**Figure 2-10**), J6, (**Figure 2-11**), and Motorola's M68CLB05C flex cable to connect the MRICS to the target system when emulating an MC68HC(9)08MR4/8 MCU. Connectors J5 and J6 are 40-pin shrouded headers (3M 2540-6002).

		J5			
No Connect	1	•	•	2	MR4_PTA1
MR4_PTA6	3	•	•	4	No Connect
MR4_PTB1	5	•	•	6	MR4_PTB2
No Connect	7	•	•	8	No Connect
Ground	9	•	•	10	No Connect
MR4_PTA2	11	•	•	12	No Connect
No Connect	13	•	•	14	Ground
MR4_PWM3	15	•	•	16	MR4_PWM4
MR4_PTC0	17	•	•	18	MR4_PTB0
MR4_PTC1	19	•	•	20	No Connect
No Connect	21	•	•	22	No Connect
MR4_PTB5	23	•	•	24	No Connect
MR4_PTA0	25	•	•	26	No Connect
Ground	27	•	•	28	No Connect
No Connect	29	•	•	30	No Connect
No Connect	31	•	•	32	Ground
No Connect	33	•	•	34	$\overline{\text{TGT\_IRQ}}$
No Connect	35	•	•	36	Ground
No Connect	37	•	•	38	Ground
No Connect	39	•	•	40	Ground

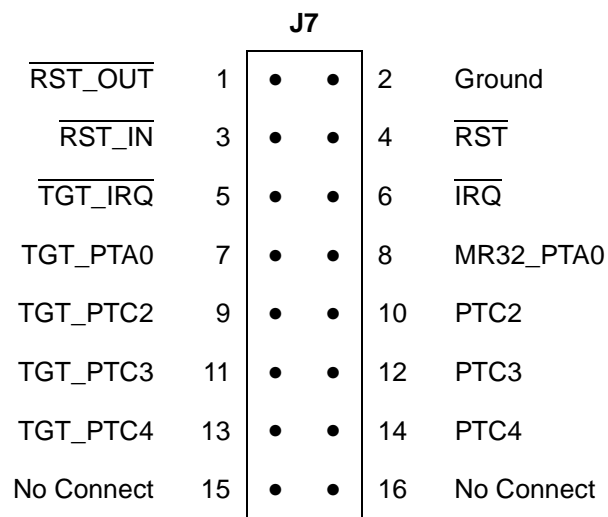
**Figure 2-10. J5 Pin Assignments; MR4/8 Target Interface**

<b>J6</b>					
Ground	1	•	•	2	MR4_PTA4
MR4_PTA5	3	•	•	4	Ground
MR4_PTB3	5	•	•	6	MR4_PTB4
No Connect	7	•	•	8	No Connect
MR4_PTA3	9	•	•	10	No Connect
No Connect	11	•	•	12	No Connect
MR4_PWM1	13	•	•	14	MR4_PWM2
MR4_PWM5	15	•	•	16	MR4_PWM6
No Connect	17	•	•	18	No Connect
Ground	19	•	•	20	No Connect
No Connect	21	•	•	22	No Connect
MR4_PTB6	23	•	•	24	Ground
No Connect	25	•	•	26	No Connect
No Connect	27	•	•	28	No Connect
No Connect	29	•	•	30	No Connect
No Connect	31	•	•	32	No Connect
No Connect	33	•	•	34	MR8/4_TGT_CLK
No Connect	35	•	•	36	$\overline{\text{TGT\_RST}}$
No Connect	37	•	•	38	Ground
No Connect	39	•	•	40	Ground

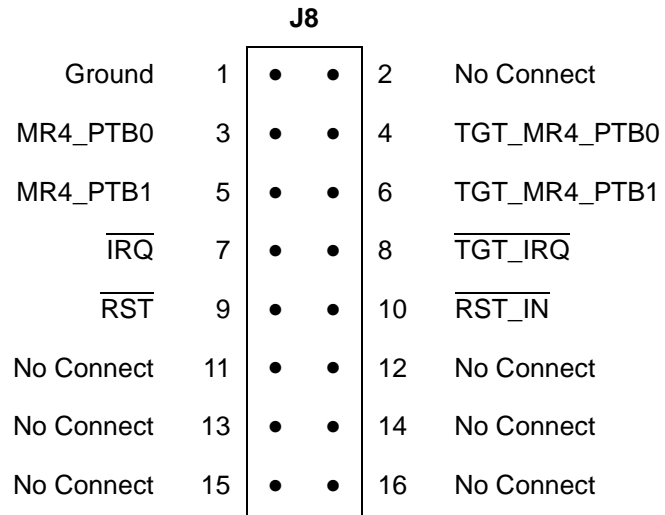
**Figure 2-11. J6 Pin Assignments; MR4/8 Target Interface**

## 2.2.3.3 MON08 Connectors (J7 and J8)

Use connectors J7 (**Figure 2-12**), J8 (**Figure 2-13**), and the provided MON08 cable to connect the MRICS board to the target system. Use connector J7 when emulating an MC68HC908MR16/32 MCU and J8 when emulating an MC68HC(9)08MR4/8 MCU. Refer to **Section 4. Using the MON08 Interface** for detailed information for using the MON08 interface.



**Figure 2-12. J7 Pin Assignments; MR16/32 MON08 Cable Connectors**



**Figure 2-13. J8 Pin Assignments; MR4/8 MON08 Cable Connectors**

#### 2.2.3.4 MR16/32 Target Connector (J1)

Use connector J1 (**Figure 2-14**) and the provided 60-pin ribbon cable to connect the MRICS to the target system when emulating an MC68HC908MR16/32 MCU. shows the interface connector.

		J1			
MR32_PTA2	1	●	●	2	MR32_PTA1
MR32_PTA3	3	●	●	4	MR32_PTA0
MR32_PTA4	5	●	●	6	Ground
MR32_PTA5	7	●	●	8	No Connect
MR32_PTA6	9	●	●	10	TGT_CLK
MR32_PTA7	11	●	●	12	No Connect
MR32_PTB0	13	●	●	14	No Connect
MR32_PTB1	15	●	●	16	$\overline{\text{TGT-RST}}$
MR32_PTB2	17	●	●	18	$\overline{\text{TGT-IRQ}}$
MR32_PTB3	19	●	●	20	PTF5
MR32_PTB4	21	●	●	22	PTF4
MR32_PTB5	23	●	●	24	Ground
MR32_PTB6	25	●	●	26	No Connect
MR32_PTB7	27	●	●	28	PTE7
No Connect	29	●	●	30	PTE6
No Connect	31	●	●	32	PTE5
Ground	33	●	●	34	PTE4
No Connect	35	●	●	36	PTE3
PTC2	37	●	●	38	No Connect
PTC3	39	●	●	40	MR32_PWM6
PTC4	41	●	●	42	MR32_PWM5
PTC5	43	●	●	44	Ground
PTC6	45	●	●	46	MR32_PWM4
PTD0	47	●	●	48	MR32_PWM3
PTD1	49	●	●	50	MR32_PWM2
PTD2	51	●	●	52	MR32_PWM1
PTD3	53	●	●	54	PTD6
PTD4	55	●	●	56	PTD5
Ground	57	●	●	58	Ground
Ground	59	●	●	60	Ground

**Figure 2-14. J1 Pin Assignments; MR16/32 Target Connector**



### 2.2.3.5 MR4/8 DIP Target Connector (J11)

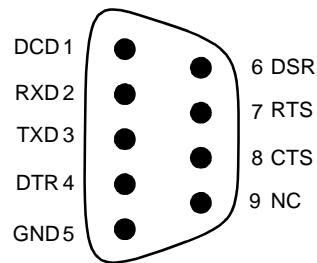
Use connector J11 (**Figure 2-15**) and the provided 28-pin ribbon cable to connect the MRICS to the target system when emulating an MC68HC(9)08MR4/8 MCU.

		<b>J11</b>			
No Connect	1	•	•	28	MR4_PTA3
$\overline{\text{TGT\_RST}}$	2	•	•	27	MR4_PTA2
No Connect	3	•	•	26	MR4_PTA1
Ground	4	•	•	25	MR4_PTA0
No Connect	5	•	•	24	MR4_PTB6
MR8/4_TGT_CLK	6	•	•	23	MR4_PTB5
No Connect	7	•	•	22	Ground
$\overline{\text{TGT\_IRQ}}$	8	•	•	21	No Connect
MR4_PWM1	9	•	•	20	MR4_PTB4
MR4_PWM2	10	•	•	19	MR4_PTB3
MR4_PWM3	11	•	•	18	MR4_PTB2
MR4_PWM4	12	•	•	17	MR4_PTB1
MR4_PWM5	13	•	•	16	MR4_PTB0
MR4_PWM6	14	•	•	15	MR4_PTC0

**Figure 2-15. J11 Pin Assignments; MR4/8 DIP Target Connector**

## 2.2.4 Host Computer (PC) — MRICS Serial Interconnection (J12)

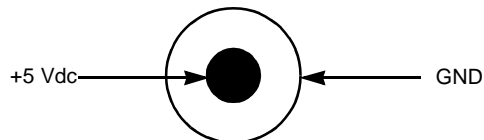
Communication with the MRICS requires an RS-232C compatible host computer connected to the MRICS I/O port J12 (**Figure 2-16**). This cable assembly is supplied with your MRICS kit and is a DE9-male-to-female, 6-ft (2-m) long serial cable. Connect one end of this cable to your host PC and the other end to connector J12 on the MRICS board.



**Figure 2-16. J12 Pin Assignments; RS232 Connector**

## 2.2.5 J4 Pin Assignments; +5Vdc Power Connector

Connect +5-Vdc power directly to the MRICS via connector J4 (**Figure 2-17**) using the provided power supply.



**Figure 2-17. J4 Pin Assignment; +5Vdc Power Connector**

## 2.3 Connecting the MRICS

The following steps provide instructions for connecting the MRICS to the host PC and power connection.

**ESD CAUTION:** *Ordinary amounts of static electricity from clothing or the work environment can damage or degrade electronic devices and equipment. For example, the electronic components installed on the printed circuit board are extremely sensitive to electrostatic discharge (ESD). Wear a grounding wrist strap whenever handling any printed circuit board. This strap provides a conductive path for safely discharging static electricity to ground.*

a. Configure the jumpers W-1 through W-7, on the MRICS, for your application.

b. Install an MCU into the appropriate socket, for your application, onto the MRICS board.

- 56-pin SDIP MC68HC908MR16/32 to XU1
- 28-pin DIP MC68HC(9)08MR4/8 to XU4
- 64-pin QFP MC68HC908MR16/32 to XU2
- 32-pin QFP MC68HC(9)08MR4/8 to XU3

*Note: Observe the pin 1 orientation with the silkscreened dot. The top (label side) of the MCU package must be visible when looking at the component side of the board.*

c. Plug the serial cable into J12 on the MRICS .

d. Plug the serial cable into the COM port on the host PC.

**NOTE:** *Steps e. through g. should not be completed until all connections to the target are completed (**Paragraph 2.4**).*

e. Connect the power cable to J4 on the MRICS board.

f. Plug the power cable into an ac power outlet, using one of the country-specific adapters.

g. The MRICS power LED lights.

### 2.4 Connecting the MRICS to a Target System

Connect the MRICS to the target system using one of these methods:

- Emulating using a flex cable

When emulating an MC68HC908MR16/32 MCU, connect the 80-pin M68CLB05C flex cable (provided with the kit) to the connectors labeled J2 and J3 on the simulator board. (Use the same cable when emulating an MC68HC(9)08MR4/8 MCU, but connect it to J5 and J6 on the MRICS board.) Attach the other end of the cable to the appropriate connector on the target system. Target head adapters are available for the 56-pin SDIP, 28-pin DIP, 32-pin QFP, and 64-pin QFP versions of the MCU.

- Emulating using a ribbon cable

When emulating an MC68HC908MR16/32 MCU connect a 60-pin flat ribbon cable to connector J1 on the simulator board. Attach the other end of the cable to the appropriate connector on the target system. When emulating an MC68HC(9)08MR4/8 MCU connect a 28-pin DIP cable to connector J11 on the simulator board. Attach the other end of the cable to the appropriate connector on the target system.

- Using a MON08 cable to debug the target system.

*Note: An MCU must be installed in the target system. No MCU should on the MRICS.*

Connect the MON08 debug interface cable to the appropriate MON08 debug interface connector (J7 for the MR 16/32 or J8 for the MR 4/8) for communication with the target system's MCU.

*Note: For more detailed information on the MON08, refer to Section 3 of this manual.*

### 2.5 Installing the Software

For instructions for installing the ICS08 software, refer to P&E Microcomputer Systems, Inc., *M68ICS08 68HC08 In-Circuit Simulator Operator's Manual*, Motorola document order number M68ICS08OM/D.

## Section 3. Support Information

### 3.1 Introduction

This section includes data and information that can be useful in the design, installation, and operation of your application.

### 3.2 MRICS Connector Signal Definitions

The tables in this section detail the pin assignments for the connectors on the M68ICS08MR board.

**NOTE:** *The signal descriptions in the following tables are for quick reference only. The MC68HC908MR32 User's Manual, MC68HC908MR32/D, contains a complete description of the MC68HC908MR32 MCU signals.*

**Table 3-1. MR16/32 Target Connector J2**

Pin No.	Schematic NET	Direction	Signal Description
1	MR32_PT B2	Bidirectional	Port B I/O – bit 2
2	MR32_PT B3	Bidirectional	Port B I/O – bit 3
3	MR32_PT B5	Bidirectional	Port B I/O – bit 5
4	MR32_PT B6	Bidirectional	Port B I/O – bit 6
5	PTC0	Bidirectional	Port C I/O – bit 0
6	PTC1	Bidirectional	Port C I/O – bit 1
7	GND	Ground	MCU ground
8	NC		No connect
9	GND	Ground	MCU ground
10	MR32_PT A7	Bidirectional	Port A I/O – bit 7
11	MR32_PT A3	Bidirectional	Port A I/O – bit 3
12	MR32_PT A4	Bidirectional	Port A I/O – bit 4
13	TGT_PT A0	Bidirectional	Port A I/O – bit 0
14	GND	Ground	MCU ground
15	TGT_CLK	In	Target clock
16	NC		No connect
17	$\overline{\text{TGT\_RST}}$	In or out	External reset
18	$\overline{\text{TGT\_IRQ}}$	In	External interrupt
19	PTF3	Bidirectional	Port F I/O – bit 3
20	PTF2	Bidirectional	Port F I/O – bit 2
21	PTF0	Bidirectional	Port F I/O – bit 0
22	GND	Ground	MCU ground
23	PTE6	Bidirectional	Port E I/O – bit 6
24	PTE5	Bidirectional	Port E I/O – bit 5
25	PTE3	Bidirectional	Port E I/O – bit 3
26	PTE2	Bidirectional	Port E I/O – bit 2
27	GND	Ground	MCU ground
28	MR32_PWM6	Out	Pulse width modulation – bit 6

**Table 3-1. MR16/32 Target Connector J2 (Continued)**

<b>Pin No.</b>	<b>Schematic NET</b>	<b>Direction</b>	<b>Signal Description</b>
29	MR32_PWM4	Out	Pulse width modulation – bit 4
30	MR32_PWM3	Out	Pulse width modulation – bit 3
31	PTD6	Bidirectional	Port D I/O – bit 6
32	GND	Ground	Flex cable shield ground
33	PTD2	Bidirectional	Port D I/O – bit 2
34	PTD3	Bidirectional	Port D I/O – bit 3
35	PTD1	Bidirectional	Port D I/O – bit 1
36	NC		No connect
37	PTC5	Bidirectional	Port C I/O – bit 5
38	GND	Ground	MCU ground
39	TGT_PTC3	Bidirectional	Port C I/O – bit 3
40	GND	Ground	MCU ground

**Table 3-2. MR16/32 Target Connector J3**

Pin No.	Schematic NET	Direction	Signal Description
1	GND	Ground	MCU ground
2	MR32_PT B4	Bidirectional	Port B I/O – bit 4
3	MR32_PT B7	Bidirectional	Port B I/O – bit 7
4	GND	Ground	MCU ground
5	NC		No connect
6	NC		No connect
7	MR32_PT B0	Bidirectional	Port B I/O – bit 0
8	MR32_PT B1	Bidirectional	Port B I/O – bit 1
9	MR32_PT A5	Bidirectional	Port A I/O – bit 5
10	MR32_PT A6	Bidirectional	Port A I/O – bit 6
11	MR32_PT A1	Bidirectional	Port A I/O – bit 1
12	MR32_PT A2	Bidirectional	Port A I/O – bit 2
13	NC		No connect
14	NC		No connect
15	NC		No connect
16	NC		No connect
17	PT F5	Bidirectional	Port F I/O – bit 5
18	PT F4	Bidirectional	Port F I/O – bit 4
19	GND	Ground	MCU ground
20	PT F1	Bidirectional	Port F I/O – bit 1
21	NC		No connect
22	PT E7	Bidirectional	Port E I/O – bit 7
23	PT E4	Bidirectional	Port E I/O – bit 4
24	GND	Ground	MCU ground
25	PT E1	Bidirectional	Port E I/O – bit 1
26	PT E0	Bidirectional	Port E I/O – bit 0
27	MR32_P W M5	Out	Pulse width modulation – bit 5
28	NC		No connect



**Table 3-2. MR16/32 Target Connector J3 (Continued)**

Pin No.	Schematic NET	Direction	Signal Description
29	MR32_PWM2	Out	Pulse width modulation – bit 2
30	MR32_PWM1	Out	Pulse width modulation – bit 1
31	PTD4	Bidirectional	Port D I/O – bit 4
32	PTD5	Bidirectional	Port D I/O – bit 5
33	NC		No connect
34	NC		No connect
35	PTC6	Bidirectional	Port C I/O – bit 6
36	PTD0	Bidirectional	Port D I/O – bit 0
37	TGT_PTC4	Bidirectional	Port C I/O – bit 4
38	GND	Ground	MCU ground
39	TGT_PTC2	Bidirectional	Port C I/O – bit 2
40	GND	Ground	MCU ground

**Table 3-3. MR4/8 Target Connector J5**

Pin No.	Schematic NET	Direction	Signal Description
1	NC		No connect
2	MR4_PTA1	Bidirectional	Port A I/O – bit 1
3	MR4_PTA6	Bidirectional	Port A I/O – bit 6
4	NC		No connect
5	MR4_PTB1	Bidirectional	Port B I/O – bit 1
6	MR4_PTB2	Bidirectional	Port B I/O – bit 2
7	NC		No connect
8	NC		No connect
9	GND	Ground	MCU ground
10	NC		No connect
11	MR4_PTA2	Bidirectional	Port A I/O – bit 2
12	NC		No connect
13	NC		No connect
14	GND	Ground	MCU ground
15	MR4_PWM3	Out	Pulse width modulation – bit 3
16	MR4_PWM4	Out	Pulse width modulation – bit 4
17	MR4_PTC0	Bidirectional	Port C I/O – bit 0
18	MR4_PTB0	Bidirectional	Port B I/O – bit 0
19	MR4_PTC1	Bidirectional	Port C I/O – bit 1
20	NC		No connect
21	NC		No connect
22	NC		No connect
23	MR4_PTB5	Bidirectional	Port B I/O – bit 5
24	NC		No connect
25	MR4_PTA0	Bidirectional	Port A I/O – bit 0
26	NC		No connect
27	GND	Ground	MCU ground
28	NC		No connect

**Table 3-3. MR4/8 Target Connector J5 (Continued)**

Pin No.	Schematic NET	Direction	Signal Description
29	NC		No connect
30	NC		No connect
31	NC		No connect
32	GND	Ground	MCU ground
33	NC		No connect
34	$\overline{\text{TGT\_IRQ}}$	In	External interrupt
35	NC		No connect
36	GND	Ground	MCU ground
37	NC		No connect
38	GND	Ground	MCU ground
39	NC		No connect
40	GND	Ground	MCU ground

**Table 3-4. MR4/8 Target Connector J6**

Pin No.	Schematic NET	Direction	Signal Description
1	GND	Ground	MCU ground
2	MR4_PTA4	Bidirectional	Port A I/O – bit 4
3	MR4_PTA5	Bidirectional	Port A I/O – bit 5
4	GND	Ground	MCU ground
5	MR4_PTB3	Bidirectional	Port B I/O – bit 3
6	MR4_PTB4	Bidirectional	Port B I/O – bit 4
7	NC		No connect
8	NC		No connect
9	MR4_PTA3	Bidirectional	Port A I/O – bit 3
10	NC		No connect
11	NC		No connect
12	NC		No connect
13	MR4_PWM1	Out	Pulse width modulation – bit 1
14	MR4_PWM2	Out	Pulse width modulation – bit 2
15	MR4_PWM5	Out	Pulse width modulation – bit 5
16	MR4_PWM6	Out	Pulse width modulation – bit 6
17	NC		No connect
18	NC		No connect
19	GND	Ground	MCU ground
20	NC		No connect
21	NC		No connect
22	NC		No connect
23	MR4_PTB6	Bidirectional	Port B I/O – bit 6
24	GND	Ground	MCU ground
25	NC		No connect
26	NC		No connect
27	NC		No connect
28	NC		No connect

**Table 3-4. MR4/8 Target Connector J6 (Continued)**

Pin No.	Schematic NET	Direction	Signal Description
29	NC		No connect
30	NC		No connect
31	NC		No connect
32	NC		No connect
33	NC		No connect
34	MR8/4_TGT_CLK	In	External clock
35	NC		No connect
36	$\overline{\text{TGT\_RST}}$	In or out	External reset
37	NC		No connect
38	GND	Ground	MCU ground
39	NC		No connect
40	GND	Ground	MCU ground

**Table 3-5. MR16/32 MON08 Connector J7 Pin Assignments**

Pin No.	Mnemonic	Signal
1	$\overline{\text{RST-OUT}}$	TARGET SYSTEM RESET — Active-low, bidirectional signal from the target system to initiate an MRICS MCU reset
2	GND	GROUND
3	$\overline{\text{RST-IN}}$	TARGET SYSTEM RESET — Active-low, bidirectional signal from the target system to initiate an MRICS MCU reset
4	$\overline{\text{RST}}$	RESET — Active-low, bidirectional signal from the MRICS MCU to initiate a target system reset
5	$\overline{\text{TGT-IRQ}}$	TARGET INTERRUPT REQUEST — Active-low input signal from the target that asynchronously provides an interrupt to the MCU
6	$\overline{\text{IRQ}}$	INTERRUPT REQUEST — Active-low input signal from the MRICS MCU that asynchronously provides an interrupt to the target system
7	TGT_PTA0	TARGET PORT A (bit 0) — General-purpose I/O signal from the target system
8	MR32_PTA0	PORT A (bit 0) — General-purpose MCU I/O signal
9	TGT_PTC2	TARGET PORT C (bit 2) — General-purpose I/O signal from the target system
10	PTC2	PORT C (bit 2) — General-purpose MCU I/O signal
11	TGT_PTC3	TARGET PORT C (bit 3) — General-purpose I/O signal from the target system
12	PTC3	PORT C (bit 3) — General-purpose MCU I/O signal
13	TGT_PTC4	TARGET PORT C (bit 4) — General-purpose I/O signal from the target system
14	PTC4	PORT C (bit 4) — General-purpose MCU I/O signal
15	NC	No connection
16	NC	No connection

**Table 3-6. MR4/8 MON08 Connector J8 Pin Assignments**

Pin No.	Mnemonic	Signal
1	GND	GROUND
2	NC	No connection
3	MR4_PTBO	PORT B (bit 0) — General-purpose MCU I/O signal
4	TGT_MR4_PTBO	TARGET PORT B BIT 0 — General-purpose I/O signal
5	MR4_PTB1	PORT B (bit 1) — General-purpose MCU I/O signal
6	TGT_MR4_PTB1	TARGET PORT B BIT 1 — General-purpose I/O signal
7	$\overline{\text{IRQ}}$	INTERRUPT REQUEST — Active-low input signal from the MRICS MCU that asynchronously provides an interrupt to the target system
8	$\overline{\text{TGT-IRQ}}$	TARGET INTERRUPT REQUEST — Active-low input signal from the target that asynchronously provides an interrupt to the MCU
9	$\overline{\text{RST}}$	RESET — Active-low, bidirectional signal from the MRICS MCU to initiate a target system reset
10	$\overline{\text{RST-IN}}$	TARGET SYSTEM RESET — Active-low, bidirectional signal from the target system to initiate an MRICS MCU reset
11	NC	No connection
12	NC	No connection
13	NC	No connection
14	NC	No connection
15	NC	No connection
16	NC	No connection

**Table 3-7. MR16/32 Target Connector J1 (Sheet 1 of 3)**

Pin No.	Schematic NET	Direction	Signal Description
1	MR32_PTA2	Bidirectional	Port A I/O – bit 2
2	MR32_PTA1	Bidirectional	Port A I/O – bit 1
3	MR32_PTA3	Bidirectional	Port A I/O – bit 3
4	MR32_PTA0	Bidirectional	Port A I/O – bit 0
5	MR32_PTA4	Bidirectional	Port A I/O – bit 4
6	GND	Ground	MCU ground
7	MR32_PTA5	Bidirectional	Port A I/O – bit 5
8	None	NC	No connection
9	MR32_PTA6	Bidirectional	Port A I/O – bit 6
10	TGT_CLK	In	External clock
11	MR32_PTA7	Bidirectional	Port A I/O – bit 7
12	None	NC	No connection
13	MR32_PTB0	Bidirectional	Port B I/O – bit 0
14	None	NC	No connection
15	MR32_PTB1	Bidirectional	Port B I/O – bit 1
16	$\overline{\text{TGT\_RST}}$	In or out	External reset
17	MR32_PTB2	Bidirectional	Port B I/O – bit 2
18	$\overline{\text{TGT\_IRQ}}$	In	External interrupt
19	MR32_PTB3	Bidirectional	Port B I/O – bit 3
20	PTF5	Bidirectional	Port F I/O – bit 5
21	MR32_PTB4	Bidirectional	Port B I/O – bit 4
22	PTF4	Bidirectional	Port F I/O – bit 4
23	MR32_PTB5	Bidirectional	Port B I/O – bit 5
24	GND	Ground	MCU ground
25	MR32_PTB6	Bidirectional	Port B I/O – bit 6
26	None	NC	No connection
27	MR32_PTB7	Bidirectional	Port B I/O – bit 7
28	PTE7	Bidirectional	Port E I/O – bit 7



**Table 3-7. MR16/32 Target Connector J1 (Sheet 2 of 3)**

<b>Pin No.</b>	<b>Schematic NET</b>	<b>Direction</b>	<b>Signal Description</b>
29	None	NC	No connection
30	PTE6	Bidirectional	Port E I/O – bit 6
31	None	NC	No connection
32	PTE5	Bidirectional	Port E I/O – bit 5
33	GND	Ground	MCU ground
34	PTE4	Bidirectional	Port E I/O – bit 4
35	None	NC	No connection
36	PTE3	Bidirectional	Port E I/O – bit 3
37	PTC2	Bidirectional	Port C I/O – bit 2
38	None	NC	No connection
39	PTC3	Bidirectional	Port C I/O – bit 3
40	MR32_PWM6	Out	Pulse width modulation – bit 6
41	PTC4	Bidirectional	Port C I/O – bit 4
42	MR32_PWM5	Out	Pulse width modulation – bit 5
43	PTC5	Bidirectional	Port C I/O – bit 5
44	GND	Ground	MCU ground
45	PTC6	Bidirectional	Port C I/O – bit 6
46	MR32_PWM4	Out	Pulse width modulation – bit 4
47	PTD0	Bidirectional	Port D I/O – bit 0
48	MR32_PWM3	Out	Pulse width modulation – bit 3
49	PTD1	Bidirectional	Port D I/O – bit 1
50	MR32_PWM2	Out	Pulse width modulation – bit 2
51	PTD2	Bidirectional	Port D I/O – bit 2
52	MR32_PWM1	Out	Pulse width modulation – bit 1
53	PTD3	Bidirectional	Port D I/O – bit 3
54	PTD6	Bidirectional	Port D I/O – bit 6

**Table 3-7. MR16/32 Target Connector J1 (Sheet 3 of 3)**

<b>Pin No.</b>	<b>Schematic NET</b>	<b>Direction</b>	<b>Signal Description</b>
55	PTD4	Bidirectional	Port D I/O – bit 4
56	PTD5	Bidirectional	Port D I/O – bit 5
57	GND	Ground	MCU ground
58	GND	Ground	MCU ground
59	GND	Ground	MCU ground
60	GND	Ground	MCU ground

**Table 3-8. MR4/8 Target DIP Connector J11**

Pin No.	Schematic NET	Direction	Signal Description
1	None	NC	No connection
2	$\overline{\text{TGT\_RST}}$	In or out	External reset
3	None	NC	No connection
4	GND	Ground	MCU ground
5	None	NC	No connection
6	MR8/4_TGT_CLK	In	External clock
7	None	NC	No connection
8	$\overline{\text{TGT\_IRQ}}$	In	External interrupt
9	MR4_PWM1	Out	Pulse width modulation – bit 1
10	MR4_PWM2	Out	Pulse width modulation – bit 2
11	MR4_PWM3	Out	Pulse width modulation – bit 3
12	MR4_PWM4	Out	Pulse width modulation – bit 4
13	MR4_PWM5	Out	Pulse width modulation – bit 5
14	MR4_PWM6	Out	Pulse width modulation – bit 6
15	MR4_PTC0	Bidirectional	Port C I/O – bit 0
16	MR4_PTB0	Bidirectional	Port B I/O – bit 0
17	MR4_PTB1	Bidirectional	Port B I/O – bit 1
18	MR4_PTB2	Bidirectional	Port B I/O – bit 2
19	MR4_PTB3	Bidirectional	Port B I/O – bit 3
20	MR4_PTB4	Bidirectional	Port B I/O – bit 4
21	None	NC	No connection
22	GND	Ground	MCU ground
23	MR4_PTB5	Bidirectional	Port B I/O – bit 5
24	MR4_PTB6	Bidirectional	Port B I/O – bit 6
25	MR4_PTA0	Bidirectional	Port A I/O – bit 0
26	MR4_PTA1	Bidirectional	Port A I/O – bit 1
27	MR4_PTA2	Bidirectional	Port A I/O – bit 2
28	MR4_PTA3	Bidirectional	Port A I/O – bit 3

**Table 3-9. Power Connector J4 Pin Assignments**

Pin No.	Mnemonic	Signal
1	5VDC	+5 VDC POWER — Input voltage (+5 Vdc @ 1.0 A) from the provided power supply used by the MRICS logic circuits
2	GND	GROUND

**Table 3-10. RS-232C Communication Connector J12 Pin Assignments**

Pin No.	Mnemonic	Signal
1	NC	NO CONNECT
2	RXD	RECEIVE DATA — Output for sending serial data to the DTE device
3	TXD	TRANSMIT DATA — Input for receiving serial data output from the DTE device
4	DTR	DATA TERMINAL READY — Input for receiving on-line/in-service/active status from the DTE device
5	GND	GROUND
6	NC	NO CONNECT
7	NC	NO CONNECT
8	NC	NO CONNECT
9	NC	NO CONNECT

### 3.3 Target-Cable Pin Assignments

The following tables describe the pin assignments for these cables:

- Flex target cable for use with the MR16/32 56-pin SDIP and 64-pin QFP target head adapters
- Flex target cable for use with the MR4/8 28-pin DIP and 32-pin QFP target head adapters
- Target MON08 cable

**Table 3-11. Flex Target Cable (M68CBL05C) for QFP Target Head Adapter B**

QFP Package (UX1) Pin No.	M68ICS08MR Board Label	MR/32 MCU Signal Name	QFP Package (UX1) Pin No.	M68ICS08MR Board Label	MR/32 MCU Signal Name
1	MR32_PTA2	PTA2	29	PTD5	PTD5/IS2#
2	MR32_PTA3	PTA3	30	PTD6	PTD6/IS3#
3	MR32_PTA4	PTA4	31	MR32_PWM1	PWM1
4	MR32_PTA5	PTA5	32	MR32_PWM2	PWM2
5	MR32_PTA6	PTA6	33	MR32_PWM3	PWM3
6	MR32_PTA7	PTA7	34	MR32_PWM4	PWM4
7	MR32_PTB0	PTB0/ATD0	35	GND	PWMGND
8	MR32_PTB1	PTB1/ATD1	36	MR32_PWM5	PWM5
9	MR32_PTB2	PTB2/ATD2	37	MR32_PWM6	PWM6
10	MR32_PTB3	PTB3/ATD3	38	NC	No Connect
11	MR32_PTB4	PTB4/ATD4	39	PTE3	PTE3/TCLKA
12	MR32_PTB5	PTB5/ATD5	40	PTE4	PTE4/TCLK0A
13	MR32_PTB6	PTB6/ATD6	41	PTE5	PTE5/TCH1A
14	MR32_PTB7	PTB7/ATD7	42	PTE6	PTE6/TCH2A
15			43	PTE7	PTE7/TCH3A
16	VDD_SW	VDDAD	44	VDD_SW	VDD
17	GND	VSSAD/VREFL	45	GND	VSS
18	VDD_SW	VREFH	46	PTF4	PTF4/RXD
19	PTC2	PTC2	47	PTF5	PTF5/TXD
20	PTC3	PTC3	48	IRQ#	IRQ#
21	PTC4	PTC4	49	RST#	RST#
22	PTC5	PTC5	50	VDD_SW	VDDA
23	PTC6	PTC6	51	GND	CGMXFC
24	PTD0	PTD0/FAULT1	52	4.9152MHz	OSC1
25	PTD1	PTD1/FAULT2	53	No Connect	OSC2
26	PTD2	PTD2/FAULT3	54	GND	VSSA
27	PTD3	PTD3/FAULT4	55	MR32_PTA0	PTA0
28	PTD4	PTD4/IS1#	56	MR32_PTA1	PTA1

**Table 3-12. Flex Target Cable (M68CBL05C) for DIP Target Head Adapter A**

DIP Package (UX2) Pin No.	M68ICS08MR Board Label	MR4/8 MCU Signal Name	DIP Package (UX2) Pin No.	M68ICS08MR Board Label	MR4/8 MCU Signal Name
1	VDD_SW	VREFH	15	MR4_PTC0	PTC0/FAULT1
2	RST#	RST#	16	MR4_PTB0	PTB0/RXD
3	VDD_SW	VDDA	17	MR4_PTB1	PTB1/TXD
4	GND	VSSA	18	MR4_PTB2	PTB2/TCLK
5	No Connect	OSC2	19	MR4_PTB3	PTB3/TCHD
6	4.0000MHz	OSC1	20	MR4_PTB4	PTB4/TCH1
7	GND	CGMXFC	21	VDD_SW	VDD
8	IRQ#	IRQ#	22	GND	VSS
9	MR4_PWM1	PWM1	23	MR4_PTB5	PTB5/TCH2
10	MR4_PWM2	PWM2	24	MR4_PTB6	PTB6/TCH3
11	MR4_PWM3	PWM3	25	MR4_PTA0	PTA0/ATD0
12	MR4_PWM4	PWM4	26	MR4_PTA1	PTA1/ATD1
13	MR4_PWM5	PWM5	27	MR4_PTA2	PTA2/ATD2
14	MR4_PWM6	PWM6	28	MR4_PTA3	PTA3/ATD3

**Table 3-13. Flex Target Cable (M68CBL05C) for QFP Target Head Adapter B**

QFP Package (UX3) Pin No.	M68ICS08MR Board Label	MR32 MCU Signal Name	QFP Package (UX3) Pin No.	M68ICS08MR Board Label	MR32 MCU Signal Name
1	MR32_PT B2	PTB2/ATD2	33	PTE1	PTE1/TCH0B
2	MR32_PT B3	PTB3/ATD3	34	PTE2	PTE2/TCH1B
3	MR32_PT B4	PTB4/ATD4	35	PTE3	PTE3/TCLKA
4	MR32_PT B5	PTB5/ATD5	36	PTE4	PTE4/TCLK0A
5	MR32_PT B6	PTB6/ATD6	37	PTE5	PTE5/TCH1A
6	MR32_PT B7	PTB6/ATD7	38	PTE6	PTE6/TCH2A
7	PTC0	PTC0/FAULT1	39	PTE7	PTE7/TCH3A
8	PTC1	PTC1/FAULT4	40	VDD_SW	VDD
9	VDD_SW	VDDAD	41	GND	VSS
10	GND	VSSAD	42	PTF0	PTF0/SPSCK
11	GND	VREFL	43	PTF1	PTF1/SS#
12	VDD_SW	VREFH	44	PTF2	PTF2/MOSI
13	PTC2	PTC2	45	PTF3	PTF3/MISO
14	PTC3	PTC3	46	PTF4	PTF4/RXD
15	PTC4	PTC4	47	PTF5	PTF5/TXD
16	PTC5	PTC5	48	IRQ#	IRQ#
17	PTC6	PTC6	49	RST#	RST#
18	PTD0	PTD0/FAULT1	50	VDD_SW	VDDA
19	PTD1	PTD1/FAULT2	51	GND	CGMXFC
20	PTD2	PTD2/FAULT3	52	4.9152MHz	OSC1
21	PTD3	PTD3/FAULT4	53	OSC2	No Connect
22	PTD4	PTD4/IS1#	54	VSSA	GND
23	PTD5	PTD5/IS2#	55	MR32_PTA0	PTA0
24	PTD6	PTD6/IS3#	56	MR32_PTA1	PTA1
25	MR32_PWM1	PWM1	57	MR32_PTA2	PTA2
26	MR32_PWM2	PWM2	58	MR32_PTA3	PTA3
27	MR32_PWM3	PWM3	59	MR32_PTA4	PTA4
28	MR32_PWM4	PWM4	60	MR32_PTA5	PTA5
29	GND	PWMGND	61	MR32_PTA6	PTA6
30	MR32_PWM5	PWM5	62	MR32_PTA7	PTA6
31	MR32_PWM6	PWM6	63	MR32_PT B0	PTB0/ATD0
32	PTE0	PTE0/CLKB	64	MR32_PT B1	PTB1/ATD1

**Table 3-14. Flex Target Cable (M68CBL05C) for DIP/QFP Target Head Adapter B**

QFP Package (UX4) Pin No.	M68ICS08MR Board Label	MR4/8 MCU Signal Name	QFP Package (UX4) Pin No.	M68ICS08MR Board Label	MR4/8 MCU Signal Name
1	VSSA	GND	17	MR4_PT3	PTB3/TCHD
2	No Connect	OSC2	18	MR4_PT4	PTB4/TCH1
3	4.0000MHz	OSC1	19	VDD_SW	VDD
4	GND	CGMXFC	20	GND	VSS
5	IRQ#	IRQ#	21	MR4_PT5	PTB5/TCH2
6	MR4_PWM1	PWM1	22	MR4_PT6	PTB6/TCH3
7	MR4_PWM2	PWM2	23	MR4_PTA0	PTA0/ATD0
8	MR4_PWM3	PWM3	24	MR4_PTA1	PTA1/ATD1
9	MR4_PWM4	PWM4	25	MR4_PTA2	PTA2/ATD2
10	MR4_PWM5	PWM5	26	MR4_PTA3	PTA3/ATD3
11	MR4_PWM6	PWM6	27	MR4_PTA4	PTA4/ATD4
12	MR4_PTC0	PTC0/FAULT1	28	MR4_PTA5	PTA5/ATD5
13	MR4_PTC1	PTC1/FAULT4	29	MR4_PTA6	PTA6/ATD6
14	MR4_PT0	PTB0/RXD	30	VDD_SW	VREFH
15	MR4_PT1	PTB1/TXD	31	RST#	RST#
16	MR4_PT2	PTB2/TCLK	32	VDD_SW	VDDA



### 3.4 Parts List

**Table 3-15. Bill of Materials (Sheet 1 of 3)**

Reference Designator	Description	Manufacturer	Part Number
	Printed circuit board assembly		01-RE10033W01
	Test procedure, ICS08MR		12ASE10033W
	Test fixture, ICS08MR		81ASE10033W
	Printed wiring board, ICS08MR		84-RE10033W01
-	Feet Rubber 0.5-inch tapered squares	Fastex	4009-00-5072
C1-C4, C7, C9-C14, C17, C18, C20-C23, C25-C36, C38, C39, C41, C42	Capacitor, 0.1 $\mu$ F, +80–20, 50 V, Z5U, C0805	AVX	08055E104ZAT2A
C5, C16	Capacitor, tantalum, 47 $\mu$ F, 20%, 16 V, C6032	AVX	TPSD476M016R0150
C6, C19	Capacitor, tantalum, 10 $\mu$ F, 20%, 16 V, C6032	AVX	TAJC106M016
C8, C24, C37, C40	Capacitor, 22000 pF, 20%, 16 V, X7R, C0805	AVX	0805YC223MAT2A
C43-C46	Capacitor, tantalum, 1 $\mu$ F, 20%, 16 V, C3216	AVX	TAJC106M016
C15	Capacitor, 560 pF, 50 V, COG, C0805	AVX	08055A561KAT2A
D1	Diode, zener, 1SMA6.0AT3, SMA	Motorola	1SMA6.0AT3
D2, D4	Diode, Schottky, MBRA130, SMA	Motorola	MBRA130
D3	Diode, Schottky, MBRA0520, SOD-123	Motorola	MBRA0520
DS1	LED, green, 5 mm, T1-3/4	Dialight	521-9173
DS2	LED, yellow, 5 mm, T1-3/4	Dialight	521-9174
F1	Fuse, 0.5 A, 250 V, 5X20 mm, FAST	Schurter	34.1513
J1	Connector, 60-pin, ribbon, with ejector	Amp	1-499922-1
J11	Socket, 28-pin DIP, machine-pin, DIP28-600	Augat	828-AG11D
J12	Connector, DE, R/A, socket	Cinch	DEKL-9SAT-F1
J2, J3, J5, J6	Header, 2 x 20, 100, shrouded	3M	2540-6002-UG

### Table 3-15. Bill of Materials (Sheet 2 of 3)

Reference Designator	Description	Manufacturer	Part Number
J4	Connector, PWR_JACK, 2.5 mm, center	Cui Stack	PJ-202B
J7, J8	Header, 8 x 2, 100, shrouded	3M	2516-6002-UG
L1	Inductor, 180 $\mu$ H, L1812	Murata	LQH4N181K04
L2	Inductor, 10 $\mu$ H, L1206	Murata	LQH1N100K04
L3-L15	Inductor, ferrite, 170 $\Omega$ , L1206	Murata	BLM31A700S
Q1, Q3	Transistor, PFET, MMBF0201, SOT-23	Motorola	MMBF0201
Q2	Transistor, PFET, MMFT5P03, SOT-223	Motorola	MMFT5P03HD
Q4	Transistor, NPN, MMBT3904, SOT-23	Motorola	MMBT3904
R1, R25	Resistor, 100 k, 5%, 1/8W, R0805	Dale	CRCW0805104J
R2, R12, R31	Resistor, 5 k, 5%, R0805	Dale	CRCW0805332J
R20, R22, R23	Resistor, 10 $\Omega$ , 1%, R1206	Dale	CRCW120610R0F
R21	Resistor, 59.0 k, 1%, R0805	Dale	CRCW08055902F
R24	Resistor, 150 $\Omega$ , 1%, R0805	Dale	CRCW08051500F
R3, R4, R6-R11, R13-R17, R19, R26-R30, R32, R34, R36-R38	Resistor, 10 k, 5%, R0805	Dale	CRCW0805103J
R33, R35	Resistor, 33 $\Omega$ , 5%, 1/8W, R0805	Dale	CRCW0805330J
R5, R18	Resistor, 470 $\Omega$ , 5%, R0805	Dale	CRCW0805471J
U10	IC, 12-bit, ripple, AC4040, SO-14	Fairchild	MM74AC4040
U11	IC, 5 V, supervisor, SOT-223	Dallas Semi	DS1233Z-5
U13	IC, MC34063A, SO8	Motorola	MC34063AD
U14	IC, buffer, three-state, SO14	Motorola	MC74ACT125D
U16	IC, LOW_POWER, RS232 driver, SO16W	Linear Tech	LT1181ACSW
U5, U12, U15	IC, hex, inverter, 74AC05, SOIC-14	Motorola	MC74AC05D
U6	IC, NC7SZ38, SOT23-5	Fairchild	NC7SZ38M5
U7	IC, NC7SZ32, SOT23-5	Fairchild	NC7SZ32M5
U8	IC, tripple, 2:1, multiplexor/demultiplexor, SO16	Motorola	MC74LVX8053D

**Table 3-15. Bill of Materials (Sheet 3 of 3)**

Reference Designator	Description	Manufacturer	Part Number
U9	IC, 74AC00, quad NAND, SO-14	Motorola	MC74AC00D
W1-W7, J9, J10	HDR, 3X1, 0.23", GOLD_PLATED	3M	2403-6112TG
XF1	Fuse, holder, 5 x 20, 3AC	Schurter	ODG 0031.8231
XU1	Socket, 56-pin SDIP	Berg	DIP 70-6056-340B
XU2	Socket, 64-pin QFP, clamshell	Yamichi	IC51-0644-824-1
XU3	Socket, 32-pin QFP, YAM_ICS51-0324-1498	Yamichi	ICS51-0324-1498
XU4	Socket, 28-pin DIP, SCREW_MACH	Robinson Nugent	ICE-286-S-TG30
XW1-XW7	Shunt, with handle	Amp	881545-1
XY1, XY2	Socket, 14-pin DIP, machine-pin, DIP14-300	Augat	814-AG11D
Y1	Oscillator, 4.9152 MHz, DIP8	Epson	SG-531P-4.9152MC2
Y2	Oscillator, 4.000 MHz, DIP8	Epson	SG-531P-4.000MC2

### 3.5 MRICS Printed Circuit Board Layout and Schematic Diagrams

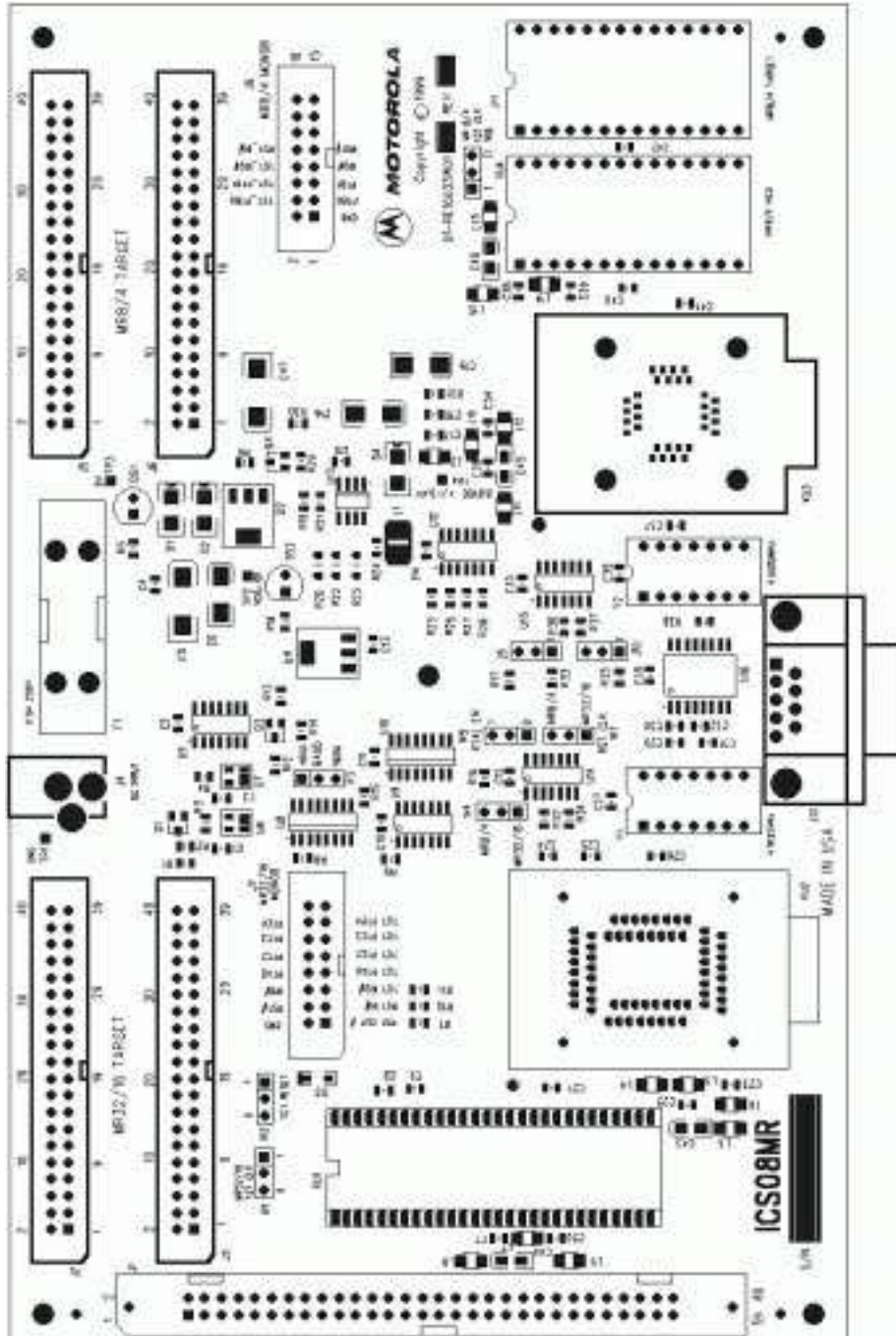
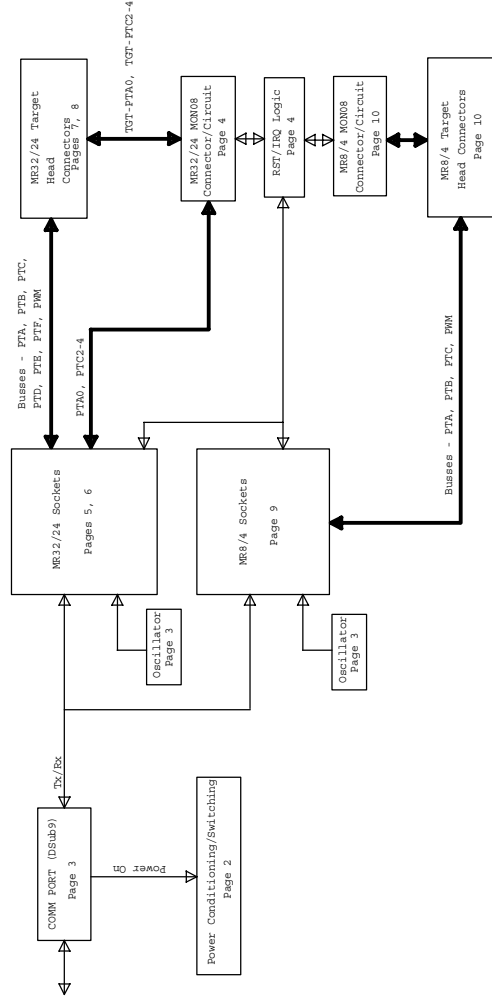


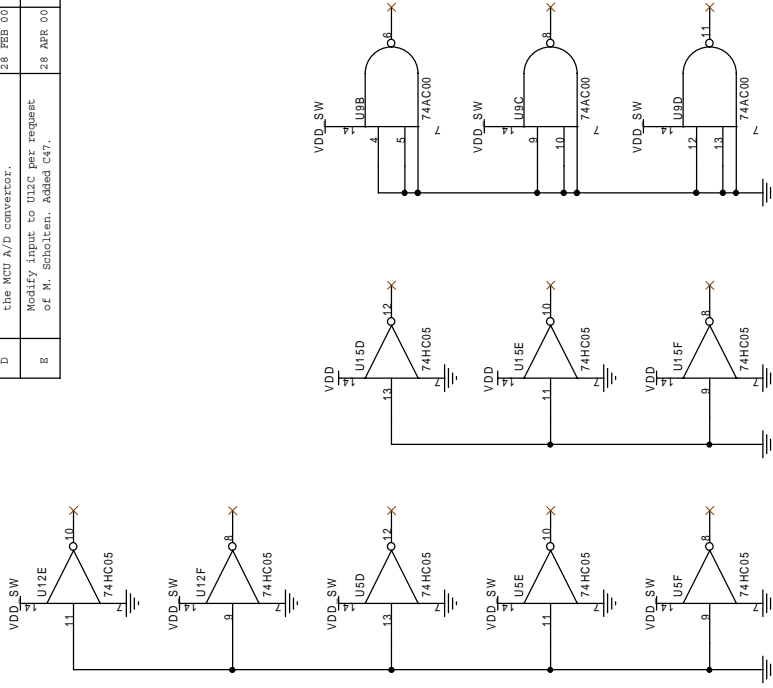
Figure 3-1. MRICS Board Layout

**NOTES:**

- UNLESS OTHERWISE SPECIFIED:  
RESISTORS ARE IN OHMS, 5%, 1/10W  
CAPACITORS ARE IN MICROFARADS, 50V
- ONLY THE NUMBER IS FOR REFERENCE  
PART MANUFACTURER.
- SPECIAL SYMBOL USAGE:  
# DENOTES ACTIVE LOW SIGNAL.  
[ ] DENOTES VECTORED SIGNAL.
- INTERPRET DIAGRAM IN ACCORDANCE WITH  
ANSI SPECIFICATIONS WITH THE  
EXCEPTION OF LOGIC BLOCK SYMBOLLOGY.



**SPARE GATES**



REV	DESCRIPTION	DATE	BY
0	RELEASE TO LAYOUT		
A	BACK ANNOTATE REEDES FROM LAYOUT	11/10/99	
B	Change Power Control and update layout. BACK ANNOTATE REFERENCE DESIGNATORS FROM LAYOUT	7 JAN 00	
C	Change pin-outs for PTB and PTE1 on MR4/8 Target Head Connectors.	11 FEB 00	
D	Modify power connections for the MCI A/D convertor.	28 FEB 00	
E	Modify input to U12C per request of M. Scholten. Added C47.	28 APR 00	

**MOTOROLA**

SEMICONDUCTOR PRODUCTS SECTOR

6501 WILLIAM CANNON DRIVE WEST AUSTIN, TX 78755 USA

THIS DOCUMENT CONTAINS INFORMATION PROPRIETARY TO MOTOROLA AND SHALL NOT BE USED FOR ENGINEERING DESIGN, PROCUREMENT OR MANUFACTURE IN WHOLE OR IN PART WITHOUT CONSENT OF MOTOROLA.

DRAWN:	<b>TEAM DEV./WRR</b>	DATE:	<b>9/99</b>
CHECKED:		DATE:	
APPROVED:		DATE:	

Title: **IN CIRCUIT SIMULATOR - ICS08MR**

Size: **B** Dwg. No.: **63BSE10033W**



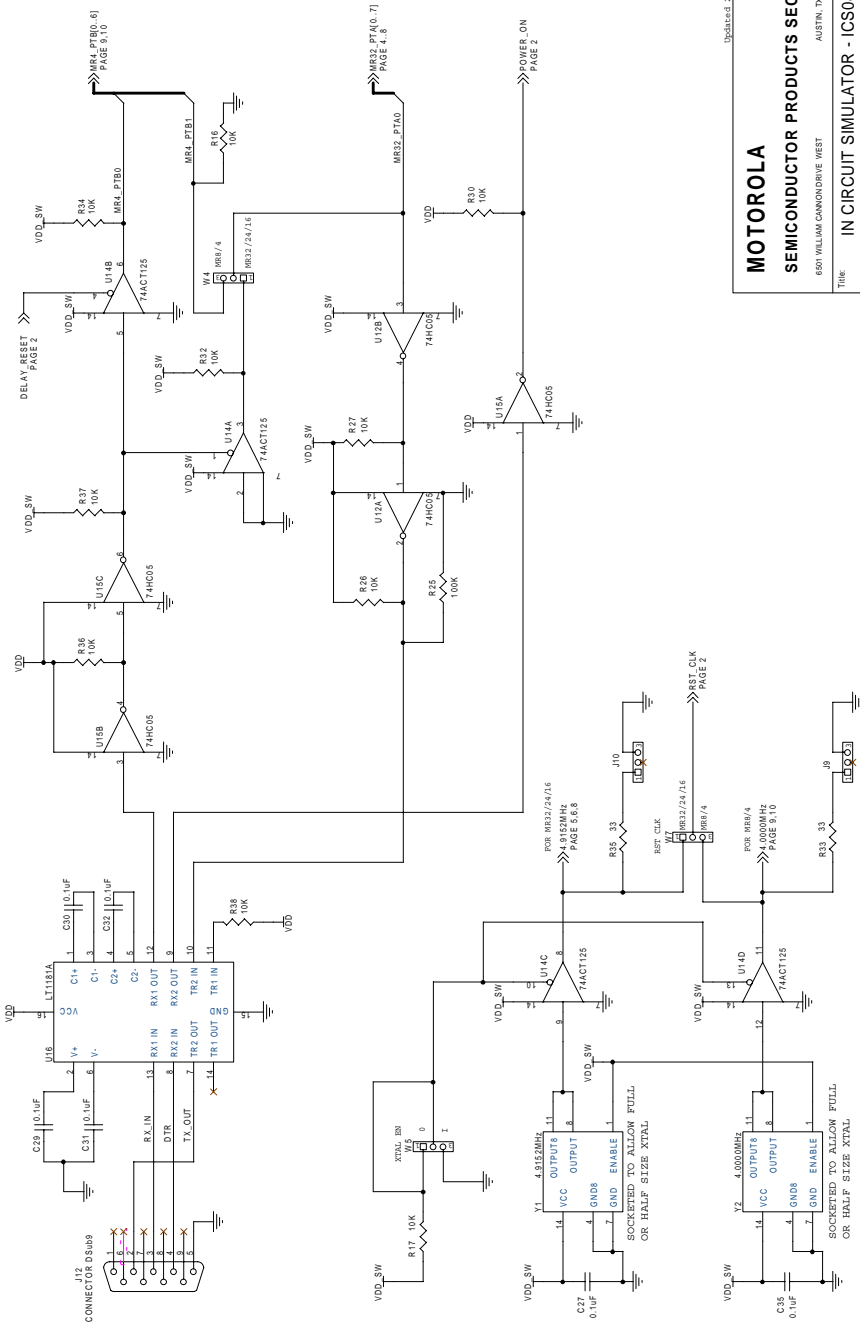
# MOTOROLA

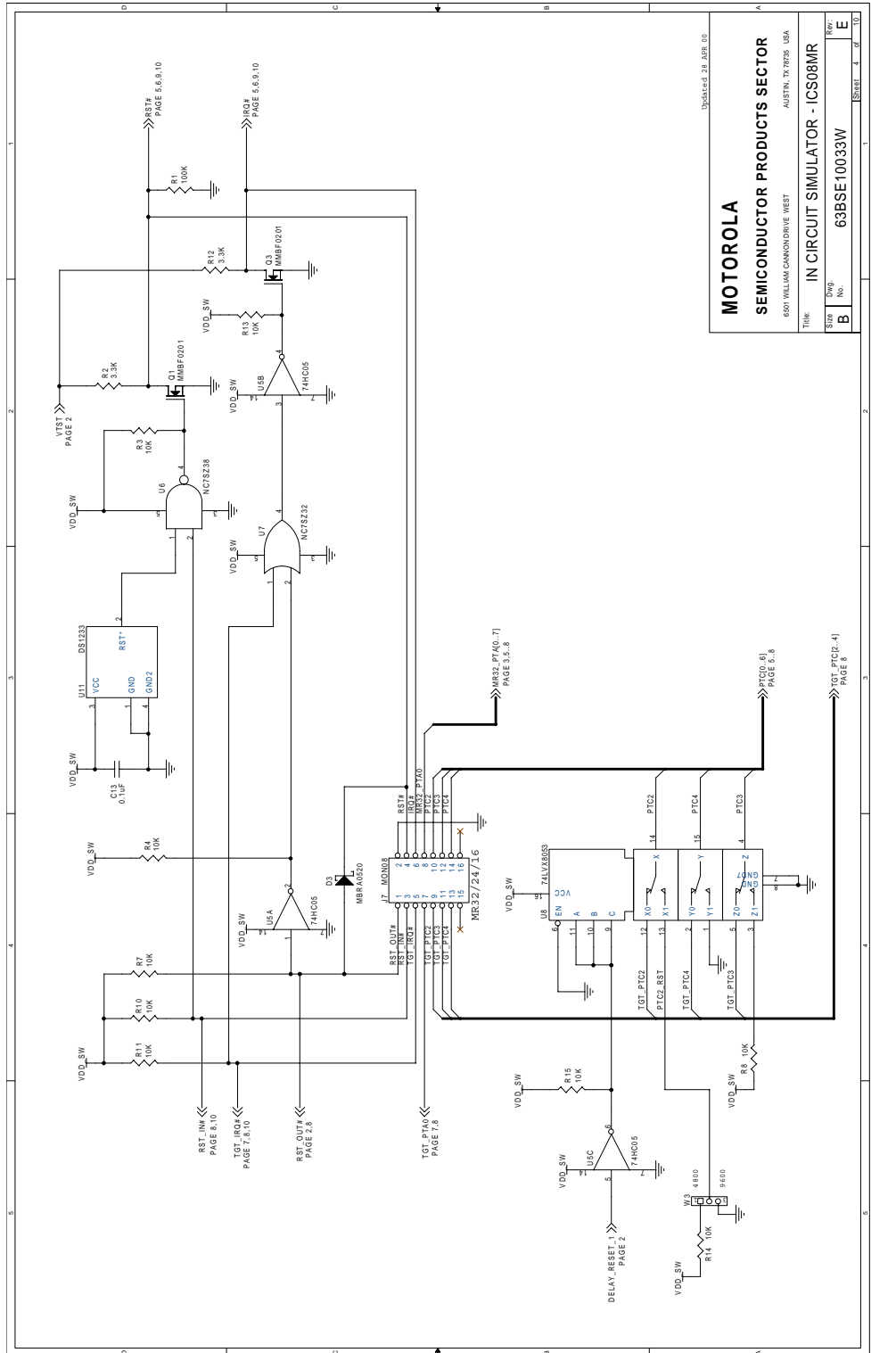
**SEMICONDUCTOR PRODUCTS SECTOR**  
 600 WILLIAM CRAWFORD DRIVE WEST  
 AUSTIN, TX 78768, USA

Title: **IN CIRCUIT SIMULATOR - ICS08MR**

Size: 1 Draw: 1  
 No. 63BSE10033W  
 Rev. E  
 Sheet 3 of 10

Updated: 21 APR 80





Updated: 21\_APR\_00

**MOTOROLA**  
**SEMICONDUCTOR PRODUCTS SECTOR**  
 6501 WILLIAM CANNON DRIVE WEST  
 AUSTIN, TX 78758, USA

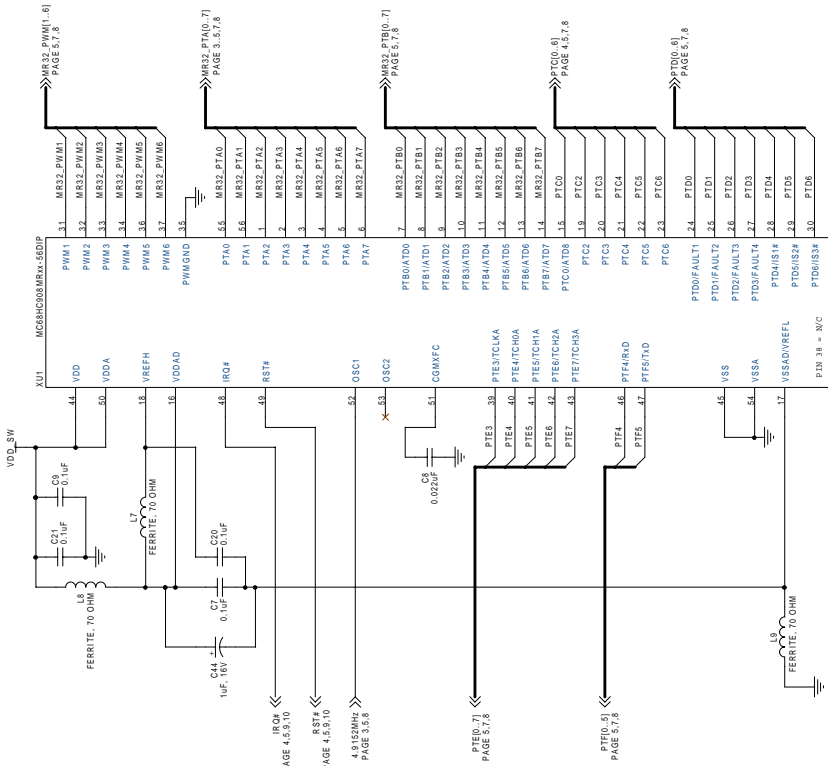
Title: **IN CIRCUIT SIMULATOR - ICS08MR**

Sheet	Dwg. No.	Rev.
<b>B</b>	<b>63BSE10033W</b>	<b>E</b>

Sheet 4 of 10



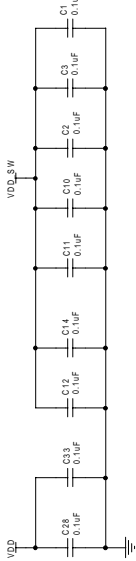




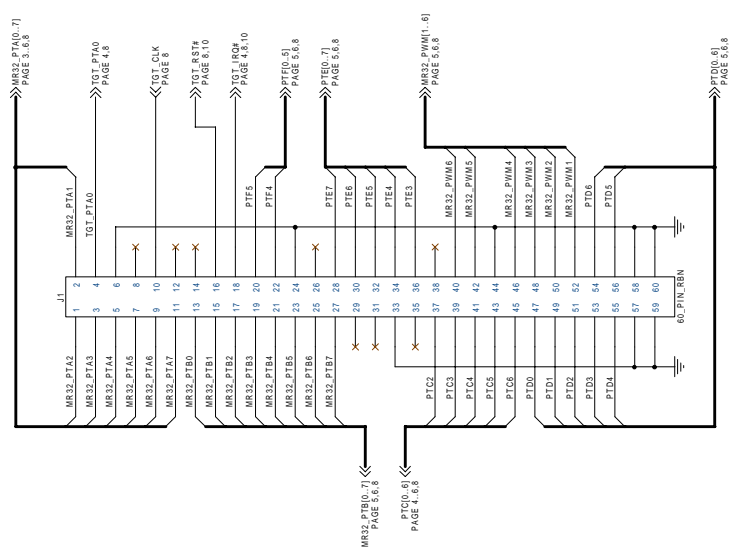
Update 2, 21 APR 00

**MOTOROLA**  
**SEMICONDUCTOR PRODUCTS SECTOR**  
 6901 WILLIAM CANNON DRIVE WEST  
 AUSTIN, TX 78758, USA

Title: **IN CIRCUIT SIMULATOR - ICS08MR**  
 Part No. **63BSE10033W**  
 Rev. **E**  
 Sheet **6** of **10**



Reference



Updated: 28 JAN 01

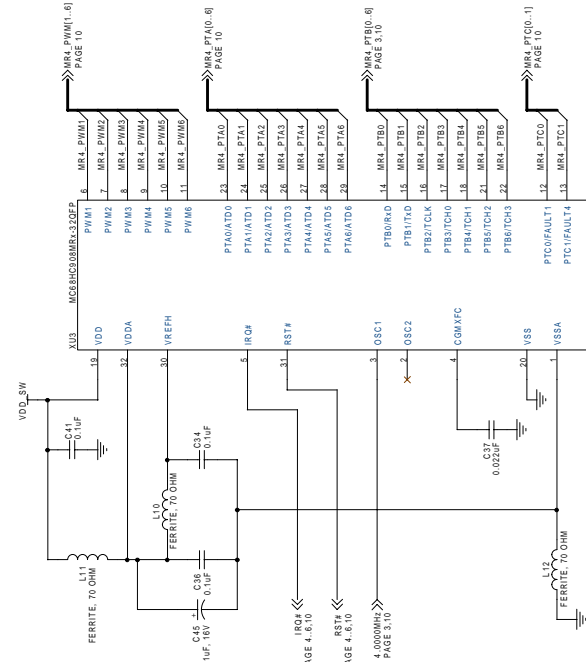
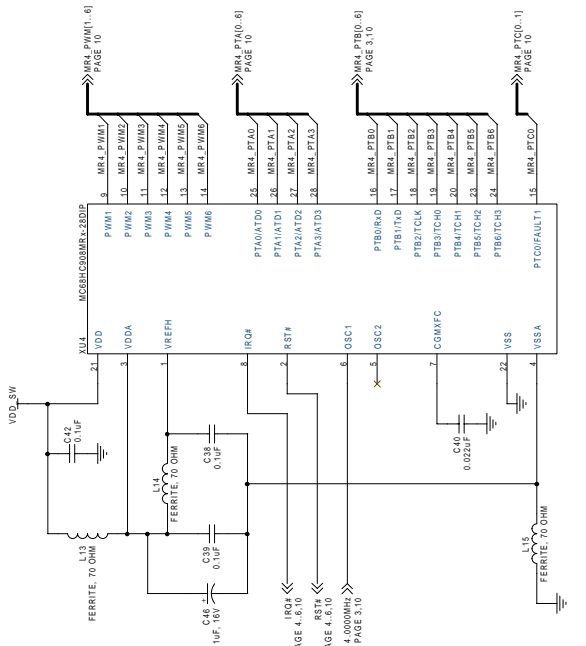
**MOTOROLA**  
**SEMICONDUCTOR PRODUCTS SECTOR**  
 6901 WILLIAM CANNON DRIVE WEST  
 AUSTIN, TX 78758, USA

Title: **IN CIRCUIT SIMULATOR - ICS08MR**

Size: **B**    Orig. No.: **63BSE10033W**    Rev.: **E**

Sheet: **7** of **10**





UPD84C33 3B MR. 10.

**MOTOROLA**

**SEMICONDUCTOR PRODUCTS SECTOR**

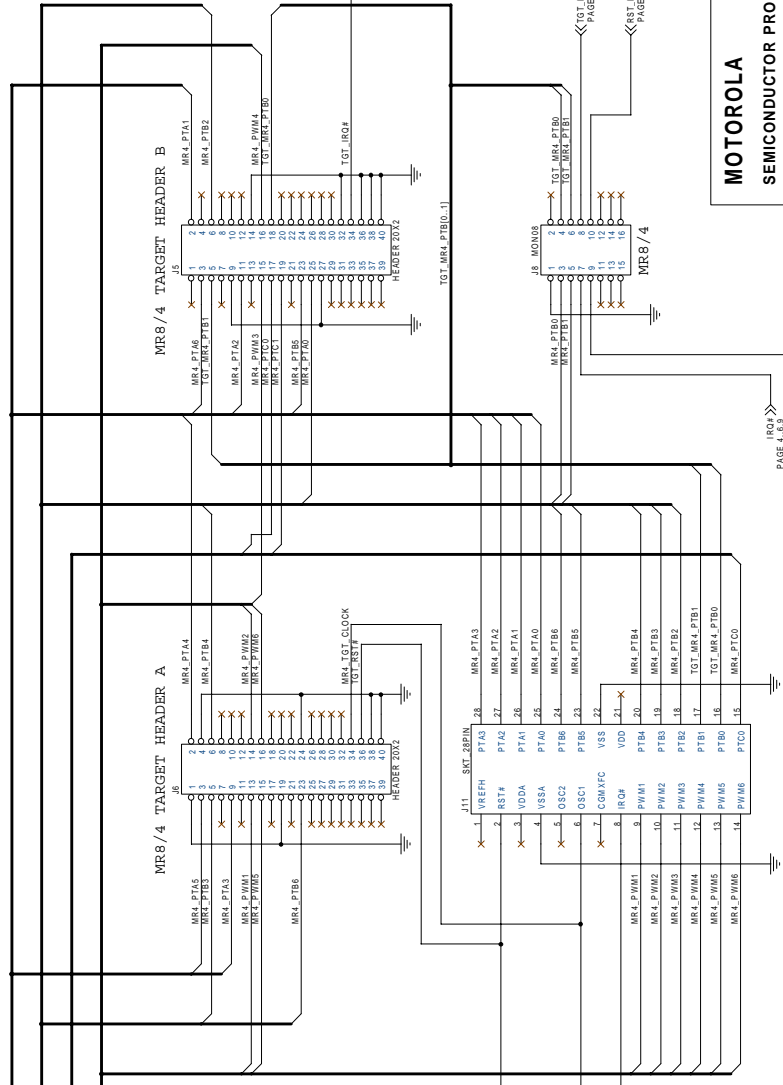
6501 WILLIAM CANNON DRIVE WEST  
AUSTIN, TX 78758, USA

**IN CIRCUIT SIMULATOR - ICS08MR**

Title:	Drawn:	Rev.:
B	B	E
No.	No.	
63BSE10033W		
DATE:	9	02

MR4\_PT40-6] PAGE 9  
 MR4\_PT80-6] PAGE 3,9  
 MR4\_PT00-1] PAGE 9  
 MR4\_PWH1-6] PAGE 9

TGT\_RST# PAGE 4,7,8  
 4.0000MHz PAGE 3,9  
 TGT\_IRQ# PAGE 4,7,8



DISPATCH 38, REV. 03

**MOTOROLA**  
**SEMICONDUCTOR PRODUCTS SECTOR**  
 6501 WILLIAM CANNON DRIVE WEST  
 AUSTIN, TX 78758, USA

Title: **IN CIRCUIT SIMULATOR - ICS08MR**

Sheet: 10 of 10

Rev: **E**

Part No: **63BSE10033W**

## Section 4. Using the MON08 Interface

### 4.1 Introduction

The MON08 debugging interface is used to debug and program an MCU that is installed on your target application. To facilitate this operation, your target board MCU must be connected to the appropriate MRICS's MON08 connector by a MON08 interface cable. This section explains how to accomplish the MON08 interface connection.

### 4.2 Target System Header Placement

Two headers are available for use on the target board, however only one is used at a time for a given application. The header used is dependent upon which MCU is selected for installation.

- MR16/32 — 16-pin header, such as Berg Electronics part number 67997-616, installed in J7 (**Table 4-1**)
- MR4/8 — 16-pin header, such as Berg Electronics part number 67997-616, installed in J8 (**Table 4-2**)

**Table 4-1. MR16/32 MON08 Target System Connector J7**

Pin No.	M68ICS08MR Label	Direction	Target System Connection
1	$\overline{\text{RST-OUT}}$	Out to target	Connect to logic that is to receive the $\overline{\text{RST}}$ signal.
2	GND	Ground	Connect to ground ( $V_{SS}$ ).
3	$\overline{\text{RST-IN}}$	In from target	Connect to all logic that generates resets.
4	$\overline{\text{RST}}$	Bidirectional	Connect to MCU $\overline{\text{RST}}$ pin and P1 pin 1. No other target-system logic should be tied to this signal. It will swing from 0 to +8.6 Vdc.
5	$\overline{\text{TGT-IRQ}}$	In from target	Connect to logic that generates interrupts.
6	$\overline{\text{IRQ}}$	Out to target	Connect to MCU $\overline{\text{IRQ}}$ pin. No other target-system logic should be tied to this signal. It will swing from 0 to +8.6 Vdc.
7	TGT-PTA0	Bidirectional	Connect to user circuit that would normally be connected to PTA0 on the MCU. This circuit will not be connected to the MCU when the in-circuit simulator is being used.
8	PTA0	Bidirectional	Connect to MCU PTA0 pin. No other target-system logic should be tied to this signal. Host I/O present on this pin.
9	TGT-PTB0	Bidirectional	Connect to user circuit that normally would be connected to PTB0 on the MCU.
10	PTB0	Bidirectional	Connect to MCU PTB0 pin. No other target-system logic should be tied to this signal. Grounded during reset and for 256 cycles after reset.
11	TGT-PTB2	Bidirectional	Connect to user circuit that normally would be connected to PTB2 on the MCU.
12	PTB2	Bidirectional	Connect to MCU PTB2 pin. No other target-system logic should be tied to this signal. Held at +5 Vdc during reset.
13	TGT-PTB3	Bidirectional	Connect to user circuit that normally would be connected to PTB3 on the MCU.
14	PTB3	Bidirectional	Connect to MCU PTB3 pin. No other target-system logic should be tied to this signal. Grounded during reset.
15	NC	NC	Not connected
16	NC	NC	Not connected



**Table 4-2. MR4/8 MON08 Target System Connector J8**

Pin No.	M68ICS08MR Label	Direction	Target System Connection
1	$\overline{\text{RST-OUT}}$	Out to target	Connect to logic that is to receive the $\overline{\text{RST}}$ signal.
2	GND	Ground	Connect to ground ( $V_{SS}$ ).
3	MR4_PTBO	Bidirectional	Connect to MCU PTB0 pin. No other target-system logic should be tied to this signal. Grounded during reset and for 256 cycles after reset.
4	TGT_MR4_PTBO	Bidirectional	Connect to user circuit that normally would be connected to PTB0 on the MCU.
5	MR4_PTBB	Bidirectional	Connect to MCU PTB1 pin. No other target-system logic should be tied to this signal. Held at +5 Vdc during reset.
6	TGT_MR4_PTBB	Bidirectional	Connect to user circuit that normally would be connected to PTB1 on the MCU.
7	$\overline{\text{IRQ}}$	Out to target	Connect to MCU $\overline{\text{IRQ}}$ pin. No other target-system logic should be tied to this signal. It will swing from 0 to +8.6 Vdc.
8	$\overline{\text{TGT-IRQ}}$	In from target	Connect to logic that generates interrupts.
9	$\overline{\text{RST}}$	Bidirectional	Connect to MCU $\overline{\text{RST}}$ pin and P1 pin 1. No other target-system logic should be tied to this signal. It will swing from 0 to +8.6 Vdc.
10	$\overline{\text{RST-IN}}$	In from target	Connect to all logic that generates resets.
11	NC	NC	Not connected
12	NC	NC	Not connected
13	NC	NC	Not connected
14	NC	NC	Not connected
15	NC	NC	Not connected
16	NC	NC	Not connected

### 4.3 Target Requirements for Using MON08

The MRICS connects PTB1..3 to TGT-PTB1..3, except during reset. During reset, PTB1..3 have voltages placed on them that configure the processor to enter the proper mode when coming out of reset. (The preceding is true for the MR16/32 but not the MR4/8) Refer to Section 9, Monitor ROM, in the Motorola MC68HC908MR manual for more information. PTB0 will not be connected to TGT-PTB0 because this signal is used for serial communications with the debugging software. (This sentence should refer to PTA0 and TGT\_PTA0 for the MR16/32. For the MR4/8 it should be PTB0 and PTB1 not connected to TGT\_MR4\_PTBO and TGTMR4\_PTBI)

Any pullups on the reset and  $\overline{IRQ}$  signals should be on the target board side of the MON08 connector and not on the MCU side. Any connections to  $\overline{RST}$  or  $\overline{IRQ}$  may cause MON08 debugging to fail and may damage components on the target since these signals will go up to 8.6 Vdc.

### 4.4 Connecting to the In-Circuit Simulator

Using the 16-pin cable provided with the MRICS kit, connect one end of the cable to the MRICS board at J7 (for the MR16/32) or J8 (for the MR4/8). Connect the other end to connector P1 on the target-system board. The pin-1 indicators on each cable end must correspond to the pin-1 indicators on the headers. P2 is not used when connecting to the MRICS board.

## Appendix A. S-Record Information

### A.1 Introduction

The Motorola S-record format was devised to encode programs or data files in a printable format for transport between computer platforms. The format also supports editing S-records and monitoring cross-platform transfer processes.

### A.2 S-Record Contents

Each S record (**Table A-1**) is a character string composed of several fields which identify:

- Record type
- Record length
- Memory address
- Code/data
- Checksum

Each byte of binary data is encoded in the S record as a 2-character hexadecimal number:

- The first character represents the high-order four bits of the byte.
- The second character represents the low-order four bits of the byte.

**Table A-1. S-Record Fields**

Record Type	Record Length	Memory Address	Code/Data	Checksum
-------------	---------------	----------------	-----------	----------

The S-record fields are described in **Table A-2**.

**Table A-2. S-Record Field Contents**

Field	Printable Characters	Contents
Type	2	S-record type — S0, S1, etc.
Record Length	2	Character pair count in the record, excluding the type and record length
Address	4, 6, or 8	2-, 3-, or 4-byte address at which the data field is to be loaded into memory
Code/Data	0 – 2n	From 0 to n bytes of executable code, memory loadable data, or descriptive information. For compatibility with teletypewriter, some programs may limit the number of bytes to as few as 28 (56 printable characters in the S-record).
Checksum	2	Least significant byte of the one's complement of the sum of the values represented by the pairs of characters making up the record length, address, and the code/data fields

Each record may be terminated with a CR/LF/NULL. Additionally, an S-record may have an initial field to accommodate other data such as the line number generated by some time-sharing systems.

Accuracy of transmission is ensured by the record length (byte count) and checksum fields.

### A.3 S-Record Types

Eight types of S-records have been defined to accommodate the several needs of the encoding, transport, and decoding functions. The various Motorola upload, download, and other record transport control programs (as well as cross assemblers, linkers, and other file-creating or debugging programs) utilize only those S-records which serve the purpose of the program.

For specific information on which S-records are supported by a particular program, consult the user manual for the program.

**NOTE:** *The ICS08MR supports only the S0, S1, and S9 record types. All data before the S1 record is ignored. Thereafter, all records must be S1 type until the S9 record, which terminates data transfer.*

An S-record format may contain the record types in **Table A-3**.

**Table A-3. S-Record Types**

Record Type	Description
S0	Header record for each block of S-records. The code/data field may contain any descriptive information identifying the following block of S records. The address field is normally 0s.
S1	Code/data record and the 2-byte address at which the code/data is to reside
S2 – S8	Not applicable to MRICS
S9	Termination record for a block of S1 records. Address field may optionally contain the 2-byte address of the instruction to which control is to be passed. If not specified, the first interplant specification encountered in the input will be used. There is no code/data field.

Only one termination record is used for each block of S-records. Normally, only one header record is used, although it is possible for multiple header records to occur.

## A.4 S Record Creation

S-record format programs may be produced by dump utilities, debuggers, cross assemblers, or cross linkers. Several programs are available for downloading a file in the S-record format from a host system to an 8- or 16-bit microprocessor-based system.

## A.5 S-Record Example

A typical S-record format, as printed or displayed, is shown in this example:

```
S00600004844521B
S1130000285F245F2212226A00042429008237C2A
```

## S-Record Information

```
S11300100002000800082529001853812341001813
S113002041E900084#42234300182342000824A952
S107003000144ED492
S9030000FC
```

In the example, the format consists of:

- An S0 header
- Four S1 code/data records
- An S9 termination record

### A.5.1 S0 Header Record

The S0 header record is described in **Table A-4**.

**Table A-4. S0 Header Record**

Field	S-Record Entry	Description
Type	S0	S-record type S0, indicating a header record
Record Length	06	Hexadecimal 06 (decimal 6), indicating six character pairs (or ASCII bytes) follow
Address	00 00	4-character, 2-byte address field; 0s
Code/Data	48 44 52	Descriptive information identifies these S1 records: ASCII H D R — “HDR”
Checksum	1B	Checksum of S0 record

### A.5.2 First S1 Record

The first S1 record is described in **Table A-5**.

**Table A-5. S1 Header Record**

Field	S-Record Entry			Description	
Type	S1			S-record type S1, indicating a code/data record to be loaded/verified at a 2-byte address	
Record Length	13			Hexadecimal 13 (decimal 19), indicating 19 character pairs, representing 19 bytes of binary data, follow	
Address	0000			4-character, 2-byte address field; hexadecimal address 0000 indicates location where the following data is to be loaded	
Code/Data	Opcode			Instruction	
	28	5F		BHCC	\$0161
	24	5F		BCC	\$0163
	22	12		BHI	\$0118
	22	6A		BHI	\$0172
	00	04	24	BRSET	0, \$04, \$012F
	29	00		BHCS	\$010D
08	23	7C	BRSET	4, \$23, \$018C	
Checksum	2A			Checksum of the first S1 record	

The 16 character pairs shown in the code/data field of **Table A-5** are the ASCII bytes of the actual program.

The second and third S1 code/data records each also contain \$13 (19T) character pairs and are ended with checksum 13 and 52, respectively. The fourth S code/data record contains 07 character pairs and has a checksum of 92.

### A.5.3 S9 Termination Record

The S9 termination record is described in **Table A-6**.

**Table A-6. S9 Header Record**

Field	S-Record Entry			Description	
Type	S9			S-record type S9, indicating a termination record	

**Table A-6. S9 Header Record**

Field	S-Record Entry	Description
Record Length	03	Hexadecimal 04, indicating three character pairs (three bytes) follow
Address	00 00	4-character, 2-byte address field; zeroes
Code/Data		There is no code/data in an S9 record.
Checksum	FC	Checksum of S9 record

## A.5.4 ASCII Characters

Each printable ASCII character in an S record is encoded in binary. **Table A-5** gives an example of encoding for the S1 record. The binary data is transmitted during a download of an S record from a host system to a 9- or 16-bit microprocessor-based system. For example, the first S1 record in **Table A-5** is sent as shown in **Figure A-1**.

TYPE		LENGTH		ADDRESS				CODE/DATA				...	CHECKSUM															
S	1	1	3	0	0	0	0	2	8	5	F	...	2	A														
5	3	3	1	3	1	3	3	3	0	3	0	3	0	3	2	3	8	3	5	4	6	...	3	2	4	1		
0101	0011	0011	0001	0011	0001	0011	0011	0011	0000	0011	0000	0011	0000	0011	0000	0011	0010	0011	1000	0011	0101	0100	0110	...	0011	0010	0100	0001

**Figure A-1. S-1 Record Example**



## Appendix B. Quick Start Hardware Configuration Guide

### B.1 Introduction

This quick start guide explains the:

- Configuration of the M68ICS08MR in-circuit simulator (MRICS) board
- Installation of the hardware
- Connection of the board to a target system

There are four methods for configuring the MRICS: standalone, simulation, evaluation, and programming.

- Standalone — ICS08MRZ.exe is running on the host computer (the MRICS is not connected). Emulation of the M68HC(9)08MR MCU CPU, registers, and I/O ports within the host computer environment.
- Simulation — Host computer is connected to the MRICS via the RS-232 cable and the ICS08MRZ.exe is running on the host computer. This provides access to the M68HC(9)08MR MCU CPU, internal registers, and I/O ports.
- Evaluation — Host computer is connected to the MRICS and the MRICS is connected to the target system via the flex cable. This method provides limited real-time evaluation of the MCU and debugging user developed hardware and software.
- Programming — Host computer is connected to the MRICS, and the MRICS is connected to the target system via the MON08 cable. Use the PROG08SZ.exe to program the MCU FLASH module. In the programming mode there is limited evaluation (port A0 on the MR24/32, and port B0 and port B1 on the MR4/8 are used for communications, so they are unavailable for emulation).

**ESD CAUTION:** *Ordinary amounts of static electricity from clothing or the work environment can damage or degrade electronic devices and equipment. For example, the electronic components installed on the printed circuit board are extremely*

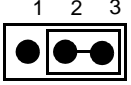
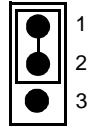
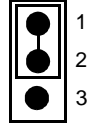
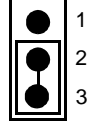
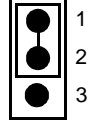
*sensitive to electrostatic discharge (ESD). Wear a grounding wrist strap whenever handling any printed circuit board. This strap provides a conductive path for safely discharging static electricity to ground.*

### **B.1.1 MRICS Configurable Jumper Headers**

Configure the seven jumper headers on the MRICS for your application according to the tables in this section.

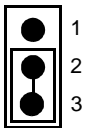
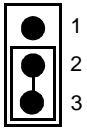
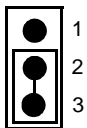
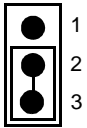
*B.1.1.1 MC68HC908MR16/32 Quick Start Configuration - Standalone Mode*

**Table B-1. MC68HC908MR16/32 Quick Start Jumper Header Configuration**

Jumper Header	Type	Description
W1 MR16/32 clock selection		Jumper on pins 2 and 3 — Disconnects the MC68HC908MR16/32 clock from the target system. This position is labeled O.
W2 Target reset selection	NA	Not applicable when using an MC68HC908MR16/32 MCU in standalone mode.
W3 I/O baud rate selection: MR16/32 only		Jumper on pins 1 and 2 (factory default) — MR16/32 I/O baud rate is set to 9600. This position is labeled 9600.
W4 Serial communication selection		Jumper on pins 1 and 2 (factory default) — MC68HC908MR16/32 MCU is installed. This position is labeled MR32/16.
W5 XTAL clock enable		Jumper on pins 2 and 3 (factory default) — MRICS XTAL clocks are enabled: 4.9152 MHz for the MC68HC908MR16/32 MCUs. This position is labeled I.
W6 MR4/8 target clock	NA	Not applicable when using an MC68HC908MR16/32 MCU in standalone mode.
W7 MRICS reset clock		Jumper on pins 1 and 2 (factory default) — Reset clock set at 4.9152 MHz. Use this setting when using the MC68HC908MR16/32 MCUs. This position is labeled MR32/16.

*B.1.1.2 MC68HC908MR4/8 Quick Start Configuration - Standalone Mode*

**Table B-2. MC68HC(9)08MR4/8 Quick Start Jumper Header Configuration**

Jumper Header	Type	Description
W1 MR16/32 clock selection	NA	Not applicable when using an MC68HC(9)08MR4/8 MCU in standalone mode.
W2 Target reset selection	NA	Not applicable when using an MC68HC(9)08MR4/8 MCU in standalone mode.
W3 I/O baud rate selection: MR16/32 only	NA	Not applicable when using an MC68HC(9)08MR4/8 MCU in standalone mode.
W4 Serial communication selection		Jumper on pins 2 and 3 — MC68HC(9)08MR4/8 is installed. This position is labeled MR8/4.
W5 XTAL clock enable		Jumper on pins 2 and 3 — MRICS XTAL clock is enabled: 4.0000 MHz for the MC68HC(9)08MR4/8 MCUs. This position is labeled I.
W6 MR4/8 target clock		Jumper on pins 2 and 3: Disconnects the MC68HC(9)08MR4/8 clock from the target system. This position is labeled O.
W7 MRICS reset clock		Jumper on pins 2 and 3: Reset clock set at 4.000 MHz. This position is labeled MR8/4.

### B.1.2 Target Interface Cable Connections

**Table B-3. MCU Version to Cable/Connector Configuration**

MCU	Flex Cable M68CLB05C	Ribbon Cable (User Supplied)	MON08 Cable
MC68HC08MR4	J5 and J6	J11	J8
MC68HC908MR8	J5 and J6	J11	J8
MC68HC908MR16	J2 and J3	J1	J7
MC68HC908MR32	J2 and J3	J1	J7

### B.1.3 Host Computer — MRICS Interconnection (J12)

Connect the DE9 serial cable. Connect one end of this cable to your host PC and the other end to connector J12 on the MRICS board.

### B.2 Installing the Hardware

For installing Motorola development tools, the following steps provide installation instructions for the MRICS hardware.

Before beginning, locate these items:

- 9-pin RS-232 serial connector on the board, labeled J12
- 5-volt circular power-input connector on the MRICS (J4)

To prepare the MRICS for use with a host PC:

1. Install the MCU into the M68ICS08MR board.

Locate the appropriate socket on the board:

- For 56-pin SDIP MC68HC908MR16/32, locate XU1
- For 28-pin DIP MC68HC(9)08MR4/8, locate XU4
- For 64-pin QFP MC68HC908MR16/32, locate XU2 (a 64-pin QFP MC68HC908MR32 is provided in the kit)
- For 32-pin QFP MC68HC(9)08MR4/8, locate XU3

Install an MCU (provided with the MRICS package) into the appropriate socket, observing the pin 1 orientation with the silkscreened dot. The top (label side) of the MCU package must be visible when looking at the component side of the board.

2. Connect the board to the host PC.

Locate the 9-pin connector labeled J12 on the board. Using the cable provided, connect it to a serial COM port on the host PC.

3. Apply power to the board.

Connect the 5-volt power supply to the round connector on the board, J4. Plug the power supply into an ac power outlet, using one of the country-specific adapters provided. The ICS power LED on the board should light.

## B.3 Installing the Software

For instructions for installing the ICS08 software, refer to P&E Microcomputer Systems, Inc., *M68ICS08 68HC08 In-Circuit Simulator Operator's Manual*, Motorola document order number M68ICS08OM/D.

## B.4 Connecting to a Target System

The three ways to connect the M68ICS08MR simulator board to a target system are via:

- The flex cable
- The ribbon cable
- The MON08 cable

Connect the simulator board to the target system using one of these methods:

- Using a flex cable

When emulating an MC68HC908MR16/32 MCU, connect the 80-pin M68CLB05C flex cable (provided with the kit) to the connectors labeled J2 and J3 on the simulator board. (Use the same cable when emulating an MC68HC(9)08MR4/8 MCU, but connect it to J5 and J6 on the MRICS board.) Attach the other end of the cable to the appropriate connector on the target system. Target head adapters are available for the 56-pin SDIP, 28-pin DIP, 32-pin QFP, and 64-pin QFP versions of the MCU.

- Using a ribbon cable

When emulating an MC68HC908MR16/32 MCU connect a 60-pin flat ribbon cable to connector J1 on the simulator board. Attach the other end of the cable to the appropriate connector on the target system. When emulating an MC68HC(9)08MR4/8 MCU connect a 28-pin DIP cable to connector J11 on the simulator board. Attach the other end of the cable to the appropriate connector on the target system.

- Using a MON08 cable

Connect the MON08 debug interface cable to the appropriate MON08 debug interface connector (either J7 or J8) for communication with the target system's MCU. The MON08 cable lets you program and debug the target system's MCU FLASH. An MCU must be installed in the target system while the MRICS board's MCU must be removed.



# Glossary

**8-bit MCU** — A microcontroller whose data is communicated over a data bus made up of eight separate data conductors. Members of the MC68HC(9)08 Family of microcontrollers are 8-bit MCUs.

**A** — An abbreviation for the accumulator of the HC08 MCU.

**accumulator** — An 8-bit register of the HC08 CPU. The contents of this register may be used as an operand of an arithmetic or logical instruction.

**assembler** — A software program that translates source code mnemonics into opcodes that can then be loaded into the memory of a microcontroller.

**assembly language** — Instruction mnemonics and assembler directives that are meaningful to programmers and can be translated into an object code program that a microcontroller understands. The CPU uses opcodes and binary numbers to specify the operations that make up a computer program. Humans use assembly language mnemonics to represent instructions. Assembler directives provide additional information such as the starting memory location for a program. Labels are used to indicate an address or binary value.

**ASCII** — American Standard Code for Information Interchange. A widely accepted correlation between alphabetic and numeric characters and specific 7-bit binary numbers.

**breakpoint** — During debugging of a program, it is useful to run instructions until the CPU gets to a specific place in the program, and then enter a debugger program. A breakpoint is established at the desired address by temporarily substituting a software interrupt (SWI) instruction for the instruction at that address. In response to the SWI, control is passed to a debugging program.

**byte** — A set of exactly eight binary bits.

**clock** — A square wave signal that is used to sequence events in a computer.

**command set** — The command set of a CPU is the set of all operations that the CPU knows how to perform. One way to represent an instruction set is with a set of shorthand mnemonics such as LDA meaning load A. Another representation of an instruction set is the opcodes that are recognized by the CPU.

**CPU** — Central processor unit. The part of a computer that controls execution of instructions.

**CPU cycles** — A CPU clock cycle is one period of the internal bus-rate clock. Normally, this clock is derived by dividing a crystal oscillator source by two or more so the high and low times will be equal. The length of time required to execute an instruction is measured in CPU clock cycles.

**CPU registers** — Memory locations that are wired directly into the CPU logic instead of being part of the addressable memory map. The CPU always has direct access to the information in these registers. The CPU registers in an MC68HC908 are A (8-bit accumulator), X (8-bit index register), CCR (condition code register containing the H, I, N, Z, and C bits), SP (stack pointer), and PC (program counter).

**cycles** — See CPU cycles.

**data bus** — A set of conductors that are used to convey binary information from a CPU to a memory location or from a memory location to a CPU; in the HC08, the data bus is 8-bits.

**development tools** — Software or hardware devices used to develop computer programs and application hardware. Examples of software development tools include text editors, assemblers, debug monitors, and simulators. Examples of hardware development tools include simulators, logic analyzers, and PROM programmers. An in-circuit simulator combines a software simulator with various hardware interfaces.

**DIP** — Dual in-line package.

**DTR** — Data transfer request.

**EPROM** — Erasable, programmable read-only memory. A non-volatile type

of memory that can be erased by exposure to an ultra-violet light source. MCUs that have EPROM are easily recognized by their packaging: a quartz window allows exposure to UV light. If an EPROM MCU is packaged in an opaque plastic package, it is termed a one-time-programmable OTP MCU, since there is no way to erase and rewrite the EPROM.

**EEPROM** — Electrically erasable, programmable read-only memory.

**ESD** — Electrostatic discharge.

**IC** — Integrated circuit.

**index register** — An 8-bit CPU register in the HC08 that is used in indexed addressing mode. The index register (X) also can be used as a general-purpose 8-bit register in addition to the 8-bit accumulator.

**input-output (I/O)** — Interfaces between a computer system and the external world. For example, a CPU reads an input to sense the level of an external signal and writes to an output to change the level on an external signal.

**instructions** — Instructions are operations that a CPU can perform. Instructions are expressed by programmers as assembly language mnemonics. A CPU interprets an opcode and its associated operand(s) as an instruction.

**listing** — A program listing shows the binary numbers that the CPU needs alongside the assembly language statements that the programmer wrote. The listing is generated by an assembler in the process of translating assembly language source statements into the binary information that the CPU needs.

**LSB** — Least significant bit.

**MCU – Microcontroller unit** — Microcontroller. A complete computer system including CPU, memory, clock oscillator, and I/O on a single integrated circuit.

**MRICS** — M68ICS08MR in-circuit simulator and programmer board.

**MR4/8** — MCUs MC68HC08MR4 and MC68HC908MR8.

**MR32** — MCU MC68HC908MR32.

**MSB** — Most significant bit.

**N** — Abbreviation for negative, a bit in the condition code register of the HC08. In two's-complement computer notation, positive signed numbers have a 0 in their MSB (most significant bit) and negative numbers have a 1 in their MSB. The N condition code bit reflects the sign of the result of an operation. After a load accumulator instruction, the N bit will be set if the MSB of the loaded value was a 1.

**object code file** — A text file containing numbers that represent the binary opcodes and data of a computer program. An object code file can be used to load binary information into a computer system. Motorola uses the S-record file format for object code files.

**operand** — An input value to a logical or mathematical operation.

**opcode** — A binary code that instructs the CPU to do a specific operation in a specific way. The HC08 CPU recognizes 210 unique 8-bit opcodes that represent addressing mode variations of 62 basic instructions.

**OTPROM** — A non-volatile type of memory that can be programmed but cannot be erased. An OTPROM is an EPROM MCU that is packaged in an opaque plastic package. It is called a one-time-programmable MCU because there is no way to expose the EPROM to a UV light.

**PC** — Abbreviation for program counter CPU register of the HC08.

**PCBA** — Printed circuit board assembly.

**PLL** — Phase-locked loop.

**program counter** — The CPU register that holds the address of the next instruction or operand that the CPU will use.

**QFP** — Quad flat pack.

**RAM** — Random access memory. Any RAM location can be read or written by the CPU. The contents of a RAM memory location remain valid until the CPU writes a different value or until power is turned off.

**registers** — Memory locations that are wired directly into the CPU logic

instead of being part of the addressable memory map. The CPU always has direct access to the information in these registers. The CPU registers in the HC08 are A (8-bit accumulator), X (8-bit index register), CCR (condition code register containing the H, I, N, Z, and C bits), SP (stack pointer), and PC (program counter). Memory locations that hold status and control information for on-chip peripherals are called I/O and control registers.

**reset** — Reset is used to force a computer system to a known starting point and to force on-chip peripherals to known starting conditions.

**S record** — A Motorola standard format used for object code files.

**SDIP** — Skinny dual in-line package.

**simulator** — A computer program that copies the behavior of a real MCU.

**source code** — See source program.

**SP** — Abbreviation for stack pointer CPU register in the HC08 MCU.

**source program** — A text file containing instruction mnemonics, labels, comments, and assembler directives. The source file is processed by an assembler to produce a composite listing and an object file representation of the program.

**stack pointer** — A CPU register that holds the address of the next available storage location on the stack.

**TTL** — Transistor-to-transistor logic.

**TVS** — Transient voltage suppression.

**V<sub>DD</sub>** — The positive power supply to a microcontroller (typically 5 volts dc).

**V<sub>SS</sub>** — The 0-volt dc power supply return for a microcontroller.

**Word** — A group of binary bits. Some larger computers consider a set of 16 bits to be a word but this is not a universal standard.

**X** — Abbreviation for index register, a CPU register in the HC08.

**Z** — Abbreviation for zero, a bit in the condition code register of the HC08. A

compare instruction subtracts the contents of the tested value from a register. If the values were equal, the result of this subtraction would be 0 so the Z bit would be set; after a load accumulator instruction, the Z bit will be set if the loaded value was \$00.

## Index

### A

ASCII  
  characters 80  
assembler  
  description 14

baud rates 17  
bus frequency 17  
bus frequency selection 17

### C

cables  
  flex target 52  
CASM08Z 14  
  description 14  
checksum 75  
communications 18  
connectors  
  Target 38, 40, 42, 44  
  Target DIP 48, 51

### D

debuggers  
  ICD08SW  
    description 10  
  MON08 interface 71

### E

examples  
  S records 77

### F

features 10  
FLASH memory 11, 14  
Flex 12  
flex target cable 12  
function keys 10

## H

- hardware
  - installation 17, 81
  - requirements 14
- host 9
- hotkeys. See function keys
- humidity 15

## I

- ICD08SZ
  - debugger 12
  - description 14
- ICG
  - MON08 limitations to 18
- ICS PWR LED 86
- ICS08MR software 12
- ICS08MRZ
  - description 14
  - simulator 12
- integrated development environment 14

## L

- LED
  - ICS power 86
- limitations
  - ICG MON08 18
  - LVI MON08 18
  - MRICS 17
- LVI
  - MON08 limitations to 18

## M

- M68CLB05C 12
- M68ICS08MR 9
- M68ICS08MR board 17, 35, 81, 86
- M68ICS08MR in-circuit simulator
  - components 12
  - hardware 14
- M68ICS08RK in-circuit simulator
  - features 10
- M68ICS08RK2 board 37
- MC68HC908MR MCUs 12
- MC68HC908MR32 9
- MC68HC908MR4 9
- MC68HC908MR8 9
- MCU 9, 13



MCU subsystem  
  clocks 23  
  target interface connector 26, 28  
memory  
  address 75  
  system requirements 14  
MON08 12, 13, 52  
MONO8 9  
Motorola 9  
MRICS 12  
  limitations 17  
  software 12  
MRICS limitations 17

## O

operating systems 14

## P

port A0 18  
port B0 18  
port B1 18  
Power 12  
power  
  connector 86  
  requirements 15  
power connector 86  
PROG08SZ  
  description 14

## R

RAM 10, 14  
record length 75  
record type 75  
relative humidity 15  
requirements  
  hardware 14  
  software 14  
RS 9  
RS-232 serial connector 86

## S

S records 75–80  
Serial 12  
serial port  
  connector 86  
software  
  ICS08MR (or MRICS software) 14

MRICS (or ICS08MR software) 14  
requirements 14

SOIC 12

S-record

content 75

creating 77

field contents 76

fields 75

overview 75

S0 record 78

S1 record 78

S9 record 79

termination record 79

types 76

## T

target 9

target system

cables 52–56

connecting to 9

connectors 37–52

description 9

MON08 interface 71

temperature

operating 15

storage 15

## W


Windows 95 9

Windows 98 9

WinIDE 10

description 14



Motorola reserves the right to make changes without further notice to any products herein. Motorola makes no warranty, representation or guarantee regarding the suitability of its products for any particular purpose, nor does Motorola assume any liability arising out of the application or use of any product or circuit, and specifically disclaims any and all liability, including without limitation consequential or incidental damages. "Typical" parameters which may be provided in Motorola data sheets and/or specifications can and do vary in different applications and actual performance may vary over time. All operating parameters, including "Typicals" must be validated for each customer application by customer's technical experts. Motorola does not convey any license under its patent rights nor the rights of others. Motorola products are not designed, intended, or authorized for use as components in systems intended for surgical implant into the body, or other applications intended to support or sustain life, or for any other application in which the failure of the Motorola product could create a situation where personal injury or death may occur. Should Buyer purchase or use Motorola products for any such unintended or unauthorized application, Buyer shall indemnify and hold Motorola and its officers, employees, subsidiaries, affiliates, and distributors harmless against all claims, costs, damages, and expenses, and reasonable attorney fees arising out of, directly or indirectly, any claim of personal injury or death associated with such unintended or unauthorized use, even if such claim alleges that Motorola was negligent regarding the design or manufacture of the part. Motorola and  are registered trademarks of Motorola, Inc. Motorola, Inc. is an Equal Opportunity/Affirmative Action Employer.

**How to reach us:**

**USA/EUROPE/Locations Not Listed:** Motorola Literature Distribution, P.O. Box 5405, Denver, Colorado 80217. 1-303-675-2140 or 1-800-441-2447. Customer Focus Center, 1-800-521-6274

**JAPAN:** Motorola Japan Ltd.; SPS, Technical Information Center, 3-20-1, Minami-Azabu, Minato-ku, Tokyo 106-8573 Japan. 81-3-3440-8573

**ASIA/PACIFIC:** Motorola Semiconductors H.K. Ltd.; Silicon Harbour Centre, 2 Dai King Street, Tai Po Industrial Estate, Tai Po, N.T., Hong Kong. 852-26668334

**Mfax™, Motorola Fax Back System:** RMFAX0@email.sps.mot.com; <http://sps.motorola.com/mfax/>; TOUCHTONE, 1-602-244-6609; US and Canada ONLY, 1-800-774-1848

**HOME PAGE:** <http://motorola.com/sps/>

Mfax is a trademark of Motorola, Inc.

© Motorola, Inc., 2000

**M681CS08MRUM/D**