M68ICS08RKUM/D
REV. 1

M68ICS08RK

In-Circuit Simulator

User's Manual

MOTOROLA

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.

For an electronic copy of this book, visit Motorola's web site at http://mcu.motsps.com/documentation

© Motorola, Inc., 2000; All Rights Reserved

List of Sections

Section 1. Introduction 13

Section 2. Hardware Installation 19

Section 3. Support Information 35

Section 4. Using the MON08 Interface 45

Appendix A. S-Record Information 49

Appendix B. Technical Reference and Troubleshooting 55

Glossary. 77

Index. 83

List of Sections

Table of Contents

Section 1. Introduction

1.1 Contents 13

1.2 Overview. 13

1.3 RKICS Components 15
1.3.1 M68ICS08RK Hardware 15
1.3.2 ICS Interface Software. 16

1.4 RKICS Features 16

1.5 Hardware and Software Requirements 17

1.6 Specifications 18

1.7 About This Manual 18

1.8 Customer Support 18

Section 2. Hardware Installation

2.1 Contents 19

2.2 Overview. 20

2.3 Configuring the In-Circuit Simulator Board. 20
2.3.1 RKICS Limitations 23
2.3.1.1 Bus Frequency. 23
2.3.1.2 Port A0 23
2.3.1.3 MC68HC908RFRK2 RF Features 23
2.3.1.4 Low Voltage Interrupt (LVI). 23
2.3.1.5 Internal Clock Generator (ICG) 23
2.3.2 RKICS Configurable Jumper Headers. 23
2.3.2.1 RKICS Power Source Selection Header (W1)
(For Target Cable Connection) 25
2.3.2.2 MCU Bus Frequency Selection Header (W2). 25

2.3.2.3 Board Reset Selection Header (W3) 26
2.3.2.4 Clock Source Selection Jumper Headers (W4 and W5) 26
2.3.3 DIP Target Connector (J3) 27
2.3.4 Operating Voltage Selection Header (J4) and Adjustment Potentiometer (R39) 28
2.3.5 Target Interface Connector (J1) 28
2.3.6 MON08 Connector (J2) 30
2.3.7 Host Computer - RKICS Interconnection (P2) 31
2.3.8 Power Connector (P1) 31

2.4 Installing the Hardware 32

2.5 Installing the Software 32

2.6 Connecting to a Target System 33

Section 3. Support Information

3.1 Contents 35

3.2 MCU Subsystem 35

3.3 Level Translation 36

3.4 RKICS Connector Signal Definitions 36

3.5 Target-Cable Pin Assignments 42

Section 4. Using the MON08 Interface

4.1 Contents 45

4.2 Overview 45

4.3 Target System Header Placement and Layout 45

4.4 Connecting to the In-Circuit Simulator 47

4.5 Disabling the Target-System Interface 47

4.6 Sample Application 48

Appendix A. S-Record Information

A.1 Contents 49
A.2 Overview. 49
A.3 S-Record Contents 49
A.4 S-Record Types. 51
A.5 S Record Creation. 52
A.6 S-Record Example 52
A.6.1 S0 Header Record 52
A.6.2 First S1 Record. 53
A.6.3 S9 Termination Record 54
A.6.4 ASCII Characters 54

Appendix B. Technical Reference and Troubleshooting

B.1 Contents 55
B.2 Overview. 55
B.3 RKICS Board 56
B.4 Troubleshooting the Start Up 58
B.5 Troubleshooting MON08 Mode 64
B.6 Parts List 68
B.7 Board Layout and Schematic Diagrams 71

Glossary

Glossary 77

Index

Index 83

Table of Contents

List of Figures

Figure	Title	Page
2-1	RKICS Board Layout	21
2-2	M68ICS08RK2 In-Circuit Simulator Block Diagram	22
2-3	ICS Functional Diagram	22
2-4	RKICS Power Source Selection Header	25
2-5	MCU Bus Frequency Selection Header	25
2-6	Board Reset Selection Header	26
2-7	Clock Source Selection Jumper Headers	26
2-8	J3 DIP Emulation Cable	27
2-9	J4 Jumper Header Voltage Ouput Options	28
2-10	J1 Target Interface	29
2-11	J2 MON08 Cable Connectors	30
2-12	Host Computer - RKICS Interconnection	31
2-13	P1 Power Connector	31
4-1	MON08 Target System Connector Layout	47
4-2	Target System Stand-Alone Connection	48
4-3	Application Designed with a Circuit for MON08	48
B-1	M68IC508RK2 Board Layout	71

List of Figures

List of Tables

Table	Title	Page
1-1	M68ICS08RK Product Components	15
1-2	Hardware Components	15
1-3	Software Components	16
1-4	M68ICS08RK Board Specifications	18
2-1	RKICS Jumper Header Description	24
2-2	MCU Bus Frequency Formula	25
2-3	Logic Table	27
3-1	Target Connector J1	37
3-2	MON08 Connector J2 Pin Assignments	39
3-3	Target DIP Connector J3	40
3-4	Power Connector P1 Pin Assignments	41
3-5	RS-232C Communication Connector P2 Pin Assignments	41
3-6	Flex Target Cable (M68CBL05A) for SOIC/SSOP/QFP Target Head Adapters	42
4-1	MON08 Target System Connector P1	46
4-2	MON08 Target System Connector P2	47
A-1	S-Record Fields	50
A-2	S-Record Field Contents	50
A-3	Record Types	51
A-4	S0 Header Record	52
A-5	S1 Header Record	53
A-6	S9 Header Record	54
B-1	RKICS Parts List	68

List of Tables

Section 1. Introduction

1.1 Contents

1.2 Overview 13

1.3 RKICS Components 15
1.3.1 M68ICS08RK Hardware 15
1.3.2 ICS Interface Software 16

1.4 RKICS Features 16

1.5 Hardware and Software Requirements 17

1.6 Specifications 18

1.7 About This Manual 18

1.8 Customer Support 18

1.2 Overview

This section provides an overview of the Motorola M68ICS08RK in-circuit simulator (RKICS).

The RKICS board, a 4-inch × 6-inch printed circuit board (PCB), is a stand-alone development and debugging aid for designers using MC68HC908RK2 microcontroller unit (MCU) devices. The RKICS contains both the hardware and software needed to develop and simulate source code for, and to program, Motorola's MC68HC908RK2 microcontrollers.

The RKICS and the RKICS software form a complete editor, assembler, programmer, simulator, and limited real-time I/O (input/output) emulator for the MC68HC908RK2 MCUs. When the RKICS is connected to a host PC (personal computer) and target hardware, the actual inputs and outputs of the target system can be used during simulation of code.

Depending on the software, the uses of the RKICS development package are:

- The WINIDE and CASM08Z software may be used as editor and assembler.
- With ICS08RKZ, the RKICS is used as a simulator.
- With the PROG08SZ software, the RKICS is used to program MCU FLASH memory.
- With the ICD08SZ software, the RKICS is used as a limited real-time emulator.

The RKICS connects to the target machine via the provided 20-pin ribbon cable or an optional Motorola M68CLB05A flex cable (available separately). It connects to the software host via a single RS-232 connection and a standard DB-9 serial cable.

Use the RKICS with any IBM® Windows 95®-based computer (or later version) with a serial port.

1.3 RKICS Components

The complete RKICS system includes hardware, software, and documentation. **Table 1-1** lists the RKICS product components.

Table 1-1. M68ICS08RK Product Components

Part Number	Description
ICS08RK	RKICS software development package
MC68HC908RK2	MCU
M68ICS08RK2	RKICS board
	Ribbon target cable
	Serial cable
	Power supply
	MON08 cable

1.3.1 M68ICS08RK Hardware

Table 1-2 lists the RKICS hardware components.

Table 1-2. Hardware Components

Components	Description
XU1 XU2 XU3	Test sockets for the Motorola MC68HC908RK2 MCU: 20-pin SSOP (small shrink outline package) 20-pin SOIC (small outline integrated circuit) 32-pin QFP (quad flat pack)
J4	User-selectable 2.0-volt to 3.3-volt variable voltage level shifter (V_{DD})
P2	RS-232 to interface the RKICS to the host serial connector
J3	One 2-row × 10-pin, 0.3-inch spacing dual in-line package (DIP) socket to allow the RKICS to be connected to the target using a ribbon cable
J1	One 2-row × 20-pin, 0.1-inch spacing connector to connect the RKICS to a target using an optional M68CLB05A flex cable
J2	One 2-row × 8-pin, 0.1-inch spacing connector to connect to a remote target via the MON08 debug circuit

1.3.2 ICS Interface Software

Windows-optimized software components are referred collectively to as the RKICS software (part number ICS08RK). The ICS08RK software package is a product of P&E Microcomputer Systems, Inc., and is included in the RKICS kit. **Table 1-3** lists these software components.

Table 1-3. Software Components

Components	Description
WINIDE.EXE	Integrated development environment (IDE) software interface for editing and interfacing with the items below
CASM08Z.EXE	CASM08Z command-line cross-assembler
ICS08RKZ.EXE	In-circuit/stand-alone simulator software for the MC68HC908RK2 MCU
PROG08SZ.EXE	FLASH memory programming software
ICD08SZ.EXE	In-circuit debugging software for limited, real-time emulation

1.4 RKICS Features

The RKICS 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 CASM08Z
- FLASH memory programming with PROG08SZ
- In-circuit and stand-alone simulation of MC68HC908RK2 MCUs with ICS08RKZ, including:
  - Simulation of all instructions, memory, and peripherals
  - Optional simulator pin inputs from the target
  - Conditional breakpoints, script files, and logfiles

- Limited real-time emulation and debugging with ICD08SZ, including:
  - Loading code into RAM
  - Executing real-time in RAM or FLASH
  - One hardware breakpoint in FLASH
  - 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, allowing:
  - Limited in-circuit emulation
  - In-circuit simulation
  - In-circuit programming

1.5 Hardware and Software Requirements

The RKICS software requires this minimum hardware and software configuration:

- An IBM-compatible host computer running 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 RKICS and the host computer

1.6 Specifications

Table 1-4 summarizes the M68ICS08RK hardware specifications.

Table 1-4. M68ICS08RK 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

1.7 About This Manual

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

1.8 Customer Support

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

Section 2. Hardware Installation

2.1 Contents

2.2 Overview 20

2.3 Configuring the In-Circuit Simulator Board 20
2.3.1 RKICS Limitations 23
2.3.1.1 Bus Frequency 23
2.3.1.2 Port A0 23
2.3.1.3 MC68HC908RFRK2 RF Features 23
2.3.1.4 Low Voltage Interrupt (LVI) 23
2.3.1.5 Internal Clock Generator (ICG) 23
2.3.2 RKICS Configurable Jumper Headers 23
2.3.2.1 RKICS Power Source Selection Header (W1) 25
2.3.2.2 MCU Bus Frequency Selection Header (W2) 25
2.3.2.3 Board Reset Selection Header (W3) 26
2.3.2.4 Clock Source Selection Jumper Headers (W4 and W5) 26
2.3.3 DIP Target Connector (J3) 27
2.3.4 Operating Voltage Selection Header (J4) and Adjustment Potentiometer (R39) 28
2.3.5 Target Interface Connector (J1) 28
2.3.6 MON08 Connector (J2) 30
2.3.7 Host Computer - RKICS Interconnection (P2) 31
2.3.8 Power Connector (P1) 31

2.4 Installing the Hardware 32

2.5 Installing the Software 32

2.6 Connecting to a Target System 33

2.2 Overview

This section explains:

- Configuration of the M68ICS08RK2 in-circuit simulator board
- Limitations of the RKICS
- Installing the hardware
- Installing the software
- Connecting the board to a target system

In interactive mode, the RKICS is connected to the serial port of a host PC. The actual inputs and outputs of a target system can be used during simulation of source code.

In software stand-alone mode, the RKICS is not connected to the PC. The ICS08RKZ software can be used as a stand-alone simulator running on the PC.

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.*

2.3 Configuring the In-Circuit Simulator Board

The RKICS includes a single 4-inch × 6-inch printed circuit board (PCB) (M68ICS08RK2). **Figure 2-1** shows a diagram of the RKICS board, **Figure 2-2** shows a block diagram of the RKICS board, and **Figure 2-3** is the functional block diagram.

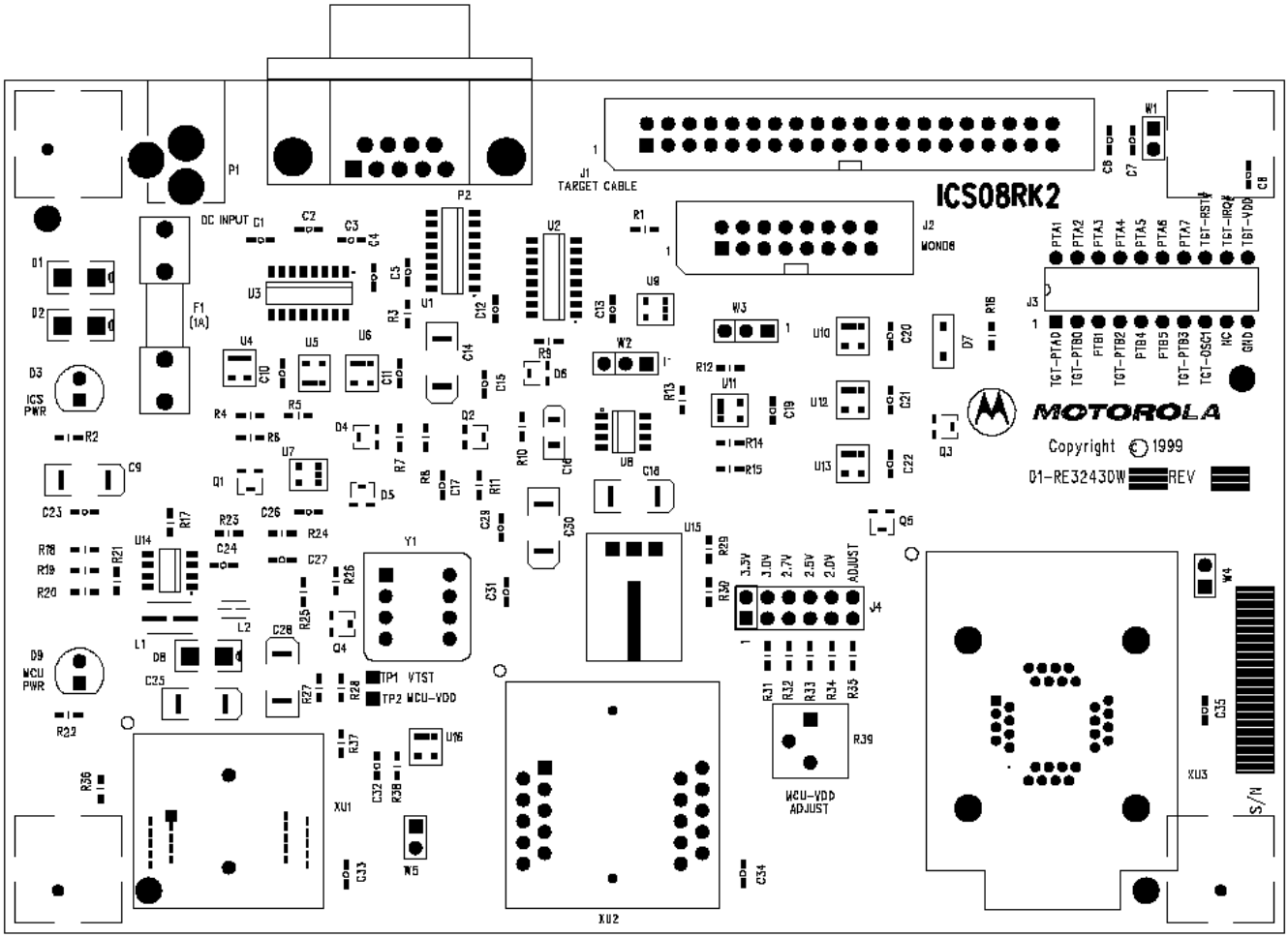


Figure 2-1. RKICS Board Layout

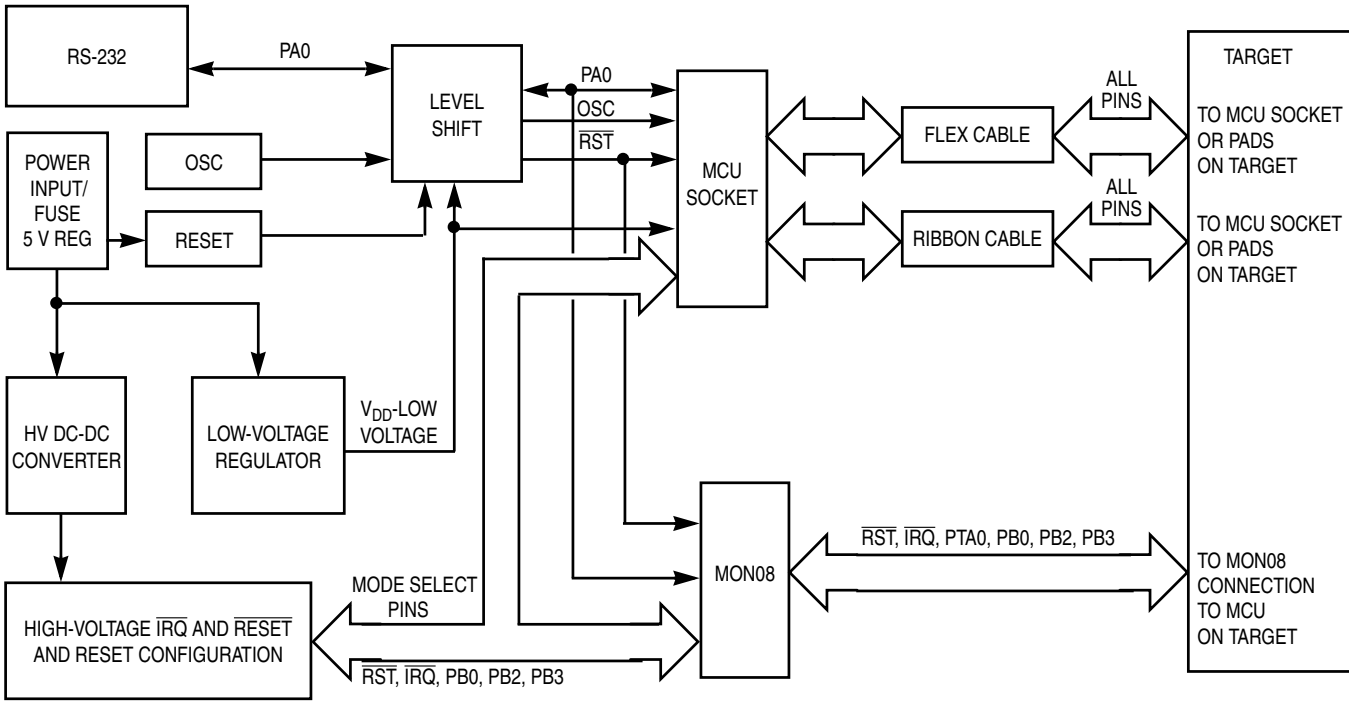


Figure 2-2. M68ICS08RK2 In-Circuit Simulator Block Diagram

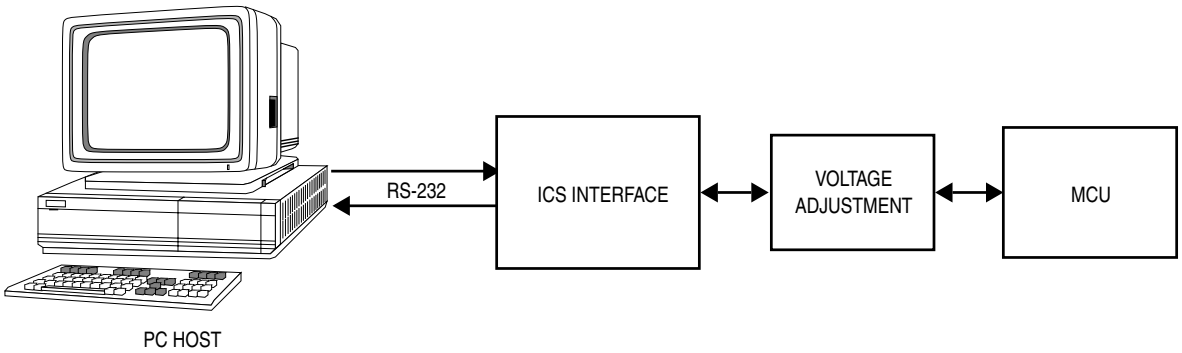


Figure 2-3. ICS Functional Diagram

2.3.1 RKICS Limitations

This section describes system limitations of the RKICS.

2.3.1.1 Bus Frequency

The RKICS communicates using the MON08 features. This forces the communication rate to $f_{bus}/256$. Therefore, the bus frequencies are limited by standard baud rates allowed by the host software. See **2.3.2.2 MCU Bus Frequency Selection Header (W2)** for available options.

2.3.1.2 Port A0

Port A0 is used for communications, so it is unavailable for emulation.

2.3.1.3 MC68HC908RFRK2 RF Features

The RF features of an MC68HC908RFRK2 installed in the QFP socket are not available. For RF emulation, use the optional flex cable (order number M68CLB05A) and QFP target head adapter (order number M68TA08RK2FA32).

2.3.1.4 Low Voltage Interrupt (LVI)

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

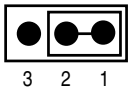
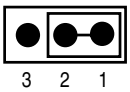
2.3.1.5 Internal Clock Generator (ICG)

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

2.3.2 RKICS Configurable Jumper Headers

Five jumper headers on the RKICS are used to reconfigure the hardware options. **Table 2-1** is a quick reference to these optional settings, while subsections **2.3.2.1** through **2.3.2.4** describe jumper header configuration in greater detail.

Table 2-1. RKICS Jumper Header Description

Jumper Header	Type	Description
W1 Power source		No jumper installed (factory default) — RKICS board power is disconnected from the target system. Target circuitry requires an separate supply. Jumper on pins 1 and 2 — Target power is supplied by the RKICS board. Only limited power is available for the target. (See **2.3.2.1 RKICS Power Source Selection Header (W1) (For Target Cable Connection)** for more information.)
W2 Bus frequency selection		Jumper on pins 1 and 2 (factory default) — MCU bus frequency is set to 2.4576 MHz (OSC ÷ 4) and the I/O baud rate is set to 9600. Jumper on pins 2 and 3 — MCU bus frequency is set to 4.9152 MHz (OSC ÷ 2) and the I/O baud rate is set to 19,200.
W3 Board reset		Jumper on pins 1 and 2 (factory default) — The MC68HC908RK2 MCU RESET signal initiates resets out to the target system. Jumper on pins 2 and 3 — The target-system RESET signal initiates resets in to the RKICS on-board MC68HC908RK2 MCU.
W4 W5 Clock source	 	**W4** / **W5**
		1 / 1: Jumpers installed on W4 and W5 — The RKICS MCU and the target system OSC1 signals are supplied by the on-board oscillator at location Y1.
		1 / 0: Jumper installed on W4 and no jumper on W5 — The RKICS MCU OSC1 signal is supplied by the target system.
		0 / 1: No jumper installed on W4 and a jumper on W5 (factory default) — The RKICS MCU OSC1 signal is supplied by the on-board oscillator at location Y1.
		0 / 0: No jumpers installed on jumper headers W4 and W5 — Connect an external clock source to the RKICS MCU OSC1 input pin via W5 pin 2.

2.3.2.1 RKICS Power Source Selection Header (W1) (For Target Cable Connection)

Use W1 to select the power source for the RKICS and target system's MCU and external circuitry (refer to **Figure 2-4**). To isolate the RKICS and target system power, ensure that no jumper is installed on jumper header W1 (factory default). Install a jumper header on W1 to use the RKICS on-board power to supply power to the the target system's MCU and external circuitry. When a jumper is installed on jumper header W1, power is supplied to the target system via J1 pins 24, 26, and 18 and J3 pin 11 (TGT-VDD).

NOTE: *The target system V_{DD} must match the ICS voltage setting in either case.*

NOTE: *This is not applicable to the MON08 connection. There is no TGT-VDD on the MON08 connector, J2.*

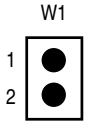


Figure 2-4. RKICS Power Source Selection Header

2.3.2.2 MCU Bus Frequency Selection Header (W2)

Use jumper header W2 to select the MCU bus speed and the I/O baud rate. Install a jumper on jumper header W2 pins 1 and 2 (factory default) to set the MCU bus frequency to 2.4576 MHz and the I/O baud rate to 9600. (Refer to **Figure 2-5**.) Install a jumper on jumper header W2 pins 2 and 3 to set the MCU bus frequency to 4.9152 MHz and the I/O baud rate to 19,200. Refer to **Table 2-2** for the MCU bus frequency formula.

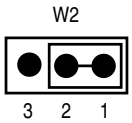


Figure 2-5. MCU Bus Frequency Selection Header

Table 2-2. MCU Bus Frequency Formula

W2 Setting	Formula
1-2 (factory default)	Crystal Oscillator Frequency divided by 4 = Bus Frequency Bus Frequency divided by 256 = Baud Rate
2-3	Crystal Oscillator Frequency divided by 2 = Bus Frequency Bus Frequency divided by 256 = Baud Rate

2.3.2.3 Board Reset Selection Header (W3)

The reset function of the RKICS is both an input and an output. The RKICS drives its $\overline{\text{RST}}$ pin low after encountering several different exception conditions. W3 lets you select whether the target system can reset the MCU on the RKICS or whether the target system receives a $\overline{\text{RST}}$ signal from the RKICS (refer to **Figure 2-6**). Install a jumper on jumper header W3 pins 1 and 2 (factory default) to use the RKICS $\overline{\text{RST}}$ signal to reset the RKICS MCU and the target system. Install a jumper on jumper header W3 pins 2 and 3 to use the target system reset function to reset the RKICS MCU.

$\overline{\text{RST}}$ is not a bidirectional, open-drain signal at the target connectors. Removing the jumper leaves the $\overline{\text{RST-IN}}$ signal pulled up to MCU operating voltage.

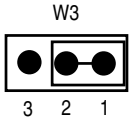


Figure 2-6. Board Reset Selection Header

2.3.2.4 Clock Source Selection Jumper Headers (W4 and W5)

The RKICS contains a 9.8304-MHz crystal oscillator. When the remote target connection is made, the user may opt to feed the output from the RKICS crystal (ICS-OCS) to the external clock input (OSC1) of the RKICS via jumper headers W4 and W5 (refer to **Figure 2-7**). Refer to **Table 2-3** for a description of the clock source options.

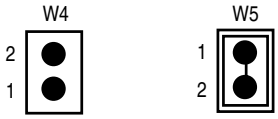


Figure 2-7. Clock Source Selection Jumper Headers

Table 2-3. Logic Table

W4	W5	Description
1	1	Jumpers installed on W4 and W5 — The RKICS MCU and the target system OSC1 signals are supplied by the on-board oscillator at location Y1
1	0	Jumper installed on W4 and no jumper on W5 — The RKICS MCU OSC1 signal is supplied by the target system
0	1	No jumper installed on W4 and a jumper on W5 (factory default) — The RKICS MCU OSC1 signal is supplied by the on-board oscillator at location Y1
0	0	No jumpers installed on jumper headers W4 and W5 — Connect an external clock source to the RKICS MCU OSC1 input pin via W5 pin 2

2.3.3 DIP Target Connector (J3)

The RKICS includes an additional connector, J3, which allows a convenient connection to the target. Use the provided 20-pin ribbon cable to interconnect the RKICS to the target system via connector J3. **Figure 2-8** shows the emulation connector interface connector.

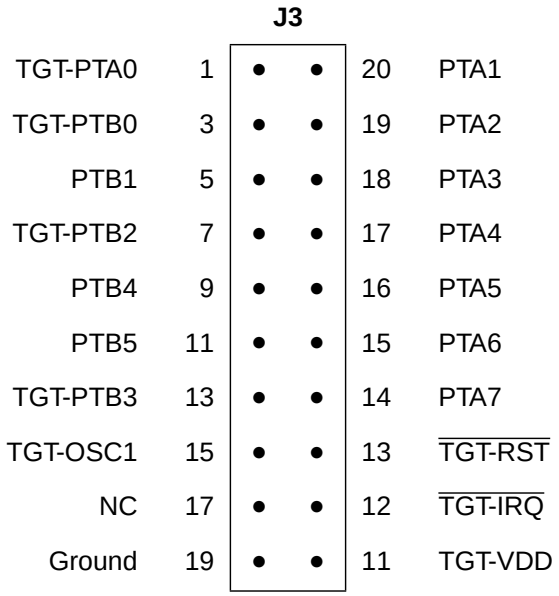


Figure 2-8. J3 DIP Emulation Cable

2.3.4 Operating Voltage Selection Header (J4) and Adjustment Potentiometer (R39)

To provide the RKICS with power input that matches the target environment, the RKICS includes a user-selectable voltage level shift. In the default setting (no jumper installed on jumper header W1) the RKICS is not connected to the power from the user's target system, so the operating voltage of the RKICS must be adjusted independently from the target. The RKICS includes a test point (TP2) for measurement of the processor voltage and a screwdriver adjustable potentiometer (R39).

To set the MCU operating voltage, jumper the set of pins on J4 (refer to **Figure 2-9**) to the appropriate voltage. If the jumper options do not match the desired operating voltage, install the jumper on pins 11 and 12. Use TP2 to measure the MCU voltage and adjust the potentiometer R39 until the correct operating voltage is reached. The default setting is 3.3 Vdc.

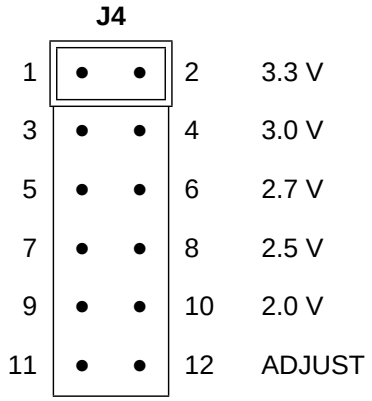


Figure 2-9. J4 Jumper Header Voltage Ouput Options

2.3.5 Target Interface Connector (J1)

Use connector J1 to connect the RKICS to the target system via the provided target cable. Connector J1 is a 40-pin shrouded header. Use the optional Motorola M68CLB05A flex cable (which must be purchased separately) to interconnect the RKICS to the target system, via connector J1.**Figure 2-10** shows J1 pinouts.

J1

TGT-PTB0	1	●	●	2	TGT-PTA0
Ground	3	●	●	4	PTA1
PTB2	5	●	●	6	PTA2
Ground	7	●	●	8	PTA3
TGT-PTB2	9	●	●	10	PTA4
Ground	11	●	●	12	PTA5
PTB4	13	●	●	14	PTA6
Ground	15	●	●	16	PTA7
PTB5	17	●	●	18	Ground
PTB2	19	●	●	20	Ground
Ground	21	●	●	22	Ground
OSC1	23	●	●	24	TGT-VDD
$\overline{\text{TGT-RST}}$	25	●	●	26	TGT-VDD
$\overline{\text{TGT-IRQ}}$	27	●	●	28	TGT-VDD
Ground	29	●	●	30	Ground
Ground	31	●	●	32	Ground
Ground	33	●	●	34	Ground
Ground	35	●	●	36	Ground
Ground	37	●	●	38	Ground
Ground	39	●	●	40	Ground

Figure 2-10. J1 Target Interface

2.3.6 MON08 Connector (J2)

Use connector J2 (refer to **Figure 2-11**) and the provided MON08 cable to connect the RKICS board to the target system. Refer to **Section 4. Using the MON08 Interface** for detailed information for using the MON08 interface.

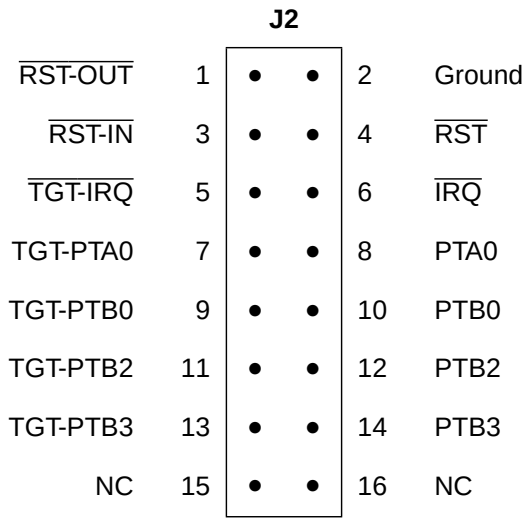


Figure 2-11. J2 MON08 Cable Connectors

2.3.7 Host Computer - RKICS Interconnection (P2)

The host computer/RKICS interface is via the single system connector P2, which is a 9-pin, D-type, through-hole, female, right angle connector (Amp part number AMP-9726-A) mounted on the top side of the PCB.

To communicate with the RKICS, connect an RS-232C compatible host computer to I/O port P2 on the RKICS. This connection requires the cable assembly supplied with your RKICS kit. This assembly cable is a DB9-male-to-female, 6-ft (3 m) long serial cable. **Figure 2-12** shows RS-232C default signal assignments for P2.

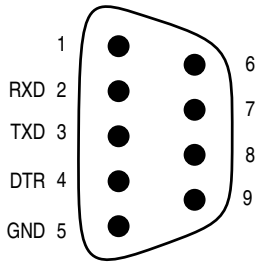


Figure 2-12. Host Computer - RKICS Interconnection

The host system interface is via a 9-pin DB-9 serial connection plug DEKL-9SAT-F.

2.3.8 Power Connector (P1)

Connect +5-Vdc power directly to the RKICS via connector P1 using the provided power supply. Power connector P1 pinouts are shown in **Figure 2-13**.

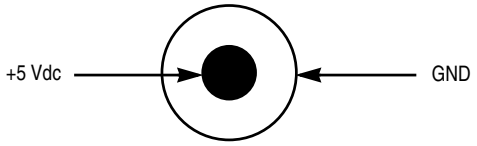


Figure 2-13. P1 Power Connector

2.4 Installing the Hardware

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

Before beginning, locate these items:

- 9-pin RS-232 serial connector on the board, labeled P2
- 5-volt circular power-input connector on the RKICS

To prepare the RKICS for use with a host PC:

1. Install the MCU into the M68ICS08RK2 board.

   Locate the appropriate socket on the board:

   – For SSOP, locate XU1

   – For SOIC, locate XU2

   – For QFP, locate XU3

   Install the MCU (provided with the RKICS package) into this 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 P2 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. 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.

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.

2.6 Connecting to a Target System

The two ways to connect the M68ICS08RK2 simulator board to a target system are:

1. Using the MCU on the board to break its processor signals out to the target system

   This method allows the board's MCU (MC68HC908RK2) to control the target system's hardware. An MCU must be installed on the M68ICS08RK2 board. The target system's MCU must be removed.

   The processor signals can be routed to the target system in two ways:

   – A 20-pin DIP emulation cable is provided with the kit for use with the connector labeled J3 on the board. Attach the cable to an equivalent connector on the target system. The pin assignments for this connector are given in **Appendix B. Technical Reference and Troubleshooting**.

   – Connector J1 on the board may be used with a flex emulation cable and target head adapter, which are available separately. Target head adapters are available for the SOIC (XU1), SSOP (XU2), and QFP (XU3) footprints on the target board.

2. Using the MON08 debug interface for communication with the target system's MCU

   This method allows in-circuit FLASH programming and debugging of the target system's MCU (MC68HC908RK2). An MCU must be installed in the target system. The board's MCU must be removed.

   Connect the board's MON08 connector with a compatible MON08 connector on the target system. Complete instructions for constructing this interface on the target board are found in **Section 4. Using the MON08 Interface**.

Section 3. Support Information

3.1 Contents

3.2 MCU Subsystem 35
3.3 Level Translation 36
3.4 RKICS Connector Signal Definitions 36
3.5 Target-Cable Pin Assignments 42

3.2 MCU Subsystem

The MCU subsystem consists of the MC68HC908RK2 microcontroller, clock generation and selection, monitor mode control logic that places and holds the RKICS in monitor mode, the bus voltage level translation buffers, and processor operating voltage variable regulator.

The MCU is an MC68HC908RK2 and is available in three different packages:

- 20-pin SOIC (small-outline integrated circuit)
- 20-pin SSOP (small shrink outline package)
- 32-pin QFP (quad flat pack)

The QFP package mounts in a clam-shell socket, while the SOIC and SSOP mount in open-top sockets. All three sockets are available on the board, though only one at a time can be used.

The on-board MCU (the test MCU) simulates and debugs the MCU's interface to its peripherals and to other devices on the target board through a variety of connections.

Depending on the connection, the MCU is used in one of three operating modes:

- In the ICS socket for programming and simple simulation
- In the socket and connected to the target for emulation
- On the target for MON08 debug operation

3.3 Level Translation

The RKICS has an operation voltage range of +2.0 to +3.3 volts while the host development system interface is an RS-232 (COM) port. U3 on the ICS converts 5-volt logic signals to RS-232 levels. ICS U5–U6 translate 5-volt logic levels to the MCU operating voltage (2.0–3.3 volts).

3.4 RKICS Connector Signal Definitions

The tables in this section describe the pin assignments for the connectors on the M68ICS08RK2 board.

NOTE: *The signal descriptions in the following tables are for quick reference only. The MC68HC908RK2 User's Manual, MC68HC908RK2/D, contains a complete description of the MC68HC908RK2 MCU signals.*

Table 3-1. Target Connector J1

Pin No.	Mnemonic	Schematic NET	Direction	Signal Description
1	PTB0/MCLK	TGT-PTB0	Bidirectional	Port B I/O – bit 0
2	PTA0	TGT-PTA0	Bidirectional	Port A I/O / keyboard interrupt – bit 0 Unavailable MCU connection
3	V_{SS}	GND	Ground	MCU ground
4	PTA1/KBD1	PTA1	Bidirectional	Port A I/O / keyboard interrupt – bit 1
5	PTB1	PTB1	Bidirectional	Port B I/O – bit 1
6	PTA2/KBD2	PTA2	Bidirectional	Port A I/O / keyboard interrupt – bit 2
7	V_{SS}	GND	Ground	MCU ground
8	PTA3/KBD3	PTA3	Bidirectional	Port A I/O / keyboard interrupt – bit 3
9	PTB2/TCH0	TG-PTB2	Bidirectional	Port B I/O – bit 2
10	PTA4/KBD4	PTA4	Bidirectional	Port A I/O / keyboard interrupt – bit 4
11	V_{SS}	GND	Ground	MCU ground
12	PTA5/KBD5	PTA5	Bidirectional	Port A I/O / keyboard interrupt – bit 5
13	PTB4/TCH1	PTB4	Bidirectional	Port B I/O – bit 4
14	PTA6/KBD6	PTA6	Bidirectional	Port A I/O / keyboard interrupt – bit 6
15	V_{SS}	GND	Ground	MCU ground
16	PTA7	PTA7	Bidirectional	Port A I/O – bit 7
17	PTB5	PTB5	Bidirectional	Port B I/O – bit 5
18	V_{SS}	GND	Ground	MCU ground
19	PTB3/TCLK	TGT-PTB3	Bidirectional	Port B I/O – bit 3
20	V_{SS}	GND	Ground	MCU ground
21	V_{SS}	GND	Ground	MCU ground
22	V_{SS}	GND	Ground	MCU ground
23	OSC1	OSC1	In	Crystal amplifier input
24	V_{DD}	TGT-VDD	Power	MCU power
25	$\overline{RST}$	$\overline{TGT\text{-}RST}$	In or out	External reset
26	V_{DD}	TGT-VDD	Power	MCU power
27	$\overline{IRQ}$	$\overline{TGT\text{-}IRQ}$	In	External Interrupt

Table 3-1. Target Connector J1 (Continued)

Pin No.	Mnemonic	Schematic NET	Direction	Signal Description
28	V_{DD}	TGT-VDD	Power	MCU power
29	None	GND	Ground	Flex cable shield ground
30	None	GND	Ground	Flex cable shield ground
31	None	GND	Ground	Flex cable shield ground
32	None	GND	Ground	Flex cable shield ground
33	None	GND	Ground	Flex cable shield ground
34	None	GND	Ground	Flex cable shield ground
35	None	GND	Ground	Flex cable shield ground
36	None	GND	Ground	Flex cable shield ground
37	None	GND	Ground	Flex cable shield ground
38	None	GND	Ground	Flex cable shield ground
39	None	GND	Ground	Flex cable shield ground
40	None	GND	Ground	Flex cable shield ground

Table 3-2. MON08 Connector J2 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 RKICS MCU reset
2	GND	GROUND
3	$\overline{\text{RST-IN}}$	TARGET SYSTEM RESET — Active-low, bidirectional signal from the target system to initiate an RKICS MCU reset
4	$\overline{\text{RST}}$	RESET — Active-low, bidirectional signal from the RKICS 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 RKICS 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	PTA0	PORT A (bit 0) — General-purpose MCU I/O signal
9	TGT-PTB0	TARGET PORT B (bit 0) — General-purpose I/O signal from the target system
10	PTB0	PORT B (bit 0) — General-purpose MCU I/O signal
11	TGT-PTB2	TARGET PORT B (bit 2) — General-purpose I/O signal from the target system
12	PTB2	PORT B (bit 2) — General-purpose MCU I/O signal
13	TGT-PTB3	TARGET PORT B (bit 3) — General-purpose I/O signal from the target system
14	PTB3	PORT B (bit 3) — General-purpose MCU I/O signal
15	NC	No connection
16	NC	No connection

Table 3-3. Target DIP Connector J3

Pin No.	Mnemonic	Schematic NET	Direction	Signal Description
1	PTA0	TGT-PTA0	Bidirectional	Port A I/O / keyboard interrupt – bit 0 Unavailable MCU connection
2	PTB0/MCLK	TGT-PTB0	Bidirectional	Port B I/O – bit 0
3	PTB1	PTB1	Bidirectional	Port B I/O – bit 1
4	PTB2/TCH0	TGT-PTB2	Bidirectional	Port B I/O – bit 2
5	PTB4/TCH1	PTB4	Bidirectional	Port B I/O – bit 4
6	PTB5	PTB5	Bidirectional	Port B I/O – bit 5
7	PTB3/TCLK	TGT-PTB3	Bidirectional	Port B I/O – bit 3
8	OSC1	OSC1	In	Crystal amplifier input
9	None	None	NC	No connection
10	V_{SS}	GND	Ground	MCU ground
11	V_{DD}	TGT-VDD	Power	MCU power
12	$\overline{IRQ}$	$\overline{TGT\text{-}IRQ}$	In	External interrupt
13	$\overline{RST}$	$\overline{TGT\text{-}RST}$	In or out	External reset
14	PTA7	PTA7	Bidirectional	Port A I/O – bit 7
15	PTA6/KBD6	PTA6	Bidirectional	Port A I/O / keyboard interrupt – bit 6
16	PTA5/KBD5	PTA5	Bidirectional	Port A I/O / keyboard interrupt – bit 5
17	PTA4/KBD4	PTA4	Bidirectional	Port A I/O / keyboard interrupt – bit 4
18	PTA3/KBD3	PTA3	Bidirectional	Port A I/O / keyboard interrupt – bit 3
19	PTA2/KBD2	PTA2	Bidirectional	Port A I/O / keyboard interrupt – bit 2
20	PTA1/KBD1	PTA1	Bidirectional	Port A I/O / keyboard interrupt – bit 1

Table 3-4. Power Connector P1 Pin Assignments

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

Table 3-5. RS-232C Communication Connector P2 Pin Assignments

Pin No.	Mnemonic	Signal
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

3.5 Target-Cable Pin Assignments

The following tables describe the pin assignments for these cables:

- Flex target cable for use with the SOIC, SSOP, and QFP target head adapters
- Target MON08 cable

Table 3-6. Flex Target Cable (M68CBL05A) for SOIC/SSOP/QFP Target Head Adapters

SOIC/SSOP Package Pin No.	QFP Package Pin No.	M68ICS08RK2 Board Label	Target Head Adapter Pin No.	M68ICS08RK2 Connector J2 Pin No.
2	3	TGT-PTB0	1	1
1	2	TGT-PTA0	2	2
10	23	GND	3	3
20	1	PTA1	4	4
3	4	PTB1	5	5
19	32	PTA2	6	6
10	23	GND	7	7
18	31	PTA3	8	8
4	5	TGT-PTB2	9	9
17	30	PTA4	10	10
10	23	GND	11	11
16	29	PTA5	12	12
5	NC	PTB4	13	13
15	28	PTA6	14	14
10	23	GND	15	15
14	27	PTA7	16	16
6	NC	PTB5	17	17
10	23	GND	18	18
7	20	TGT-PTB3	19	19
10	23	GND	20	20

Table 3-6. Flex Target Cable (M68CBL05A) for SOIC/SSOP/QFP Target Head Adapters (Continued)

SOIC/SSOP Package Pin No.	QFP Package Pin No.	M68ICS08RK2 Board Label	Target Head Adapter Pin No.	M68ICS08RK2 Connector J2 Pin No.
10	23	GND	21	21
10	23	GND	22	22
8	21	OSC1	23	23
11	24	TGT-VDD	24	24
13	26	$\overline{\text{TGT-RST}}$	25	25
11	24	TGT-VDD	26	26
12	25	$\overline{\text{TGT-IRQ}}$	27	27
11	24	TGT-VDD	28	28
NC	NC	GND	29	29
NC	NC	GND	30	30
NC	NC	GND	31	31
NC	NC	GND	32	32
NC	NC	GND	33	33
NC	NC	GND	34	34
NC	NC	GND	35	35
NC	NC	GND	36	36
NC	NC	GND	37	37
NC	NC	GND	38	38
NC	NC	GND	39	39
NC	NC	GND	40	40

Section 4. Using the MON08 Interface

4.1 Contents

4.2 Overview 45
4.3 Target System Header Placement and Layout 45
4.4 Connecting to the In-Circuit Simulator. 47
4.5 Disabling the Target-System Interface 47
4.6 Sample Application. 48

4.2 Overview

The MON08 debugging interface may be used to debug and program a target system's MCU directly. The target system must be connected to the M68ICS08RK in-circuit simulator board's MON08 interface connector. This section explains how to connect to the MON08 interface on the target board.

4.3 Target System Header Placement and Layout

Two headers must be placed on the target board:

- P1 — 16-pin header such as Berg Electronics part number 67997-616
- P2 — 1-pin header such as Berg Electronics part number 68001-601

Table 4-1 and **Table 4-2** show the target-system interconnections for P1 and P2. **Figure 4-1** shows the pin layouts for P1 and P2. Additional information about the connections on the RKICS board can be found in **Appendix B. Technical Reference and Troubleshooting**.

Table 4-1. MON08 Target System Connector P1

Pin No.	M68ICS08RK2 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 +7.5 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 +7.5 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. MON08 Target System Connector P2

Pin No.	M68ICS08RK2 Label	Direction	Target System Connection
1	$\overline{RST}$	Bidirectional	Connect to MCU $\overline{RST}$ pin and P2 pin 4. No other target system logic should be tied to this signal. It will swing from 0 to +7.5 Vdc.

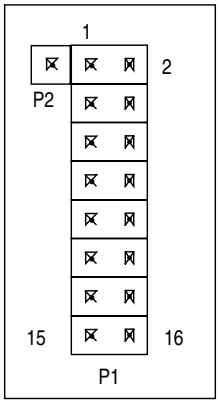


Figure 4-1. MON08 Target System Connector Layout

4.4 Connecting to the In-Circuit Simulator

Using the 16-pin cable provided with the RKICS kit, connect one end of the cable to the RKICS board at J2. 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 RKICS board.

4.5 Disabling the Target-System Interface

To use the target system in a stand-alone fashion (without the RKICS board connected), jumper the pins on the target board's connectors, as shown in **Figure 4-2**. This reconnects the target MCU to the appropriate circuits on the target system.

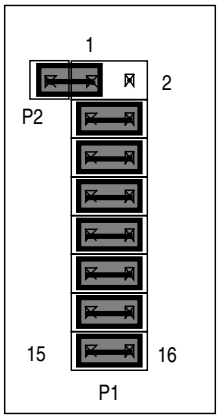


Figure 4-2. Target System Stand-Alone Connection

For production boards, a further enhancement of this scheme would be to include cutable traces between the pins of P1 and P2, as shown in **Figure 4-2**. The traces may be cut when debugging is necessary. To return the board to stand-alone use, jumpers may be installed as shown.

4.6 Sample Application

The circuit shown in **Figure 4-3** intercepts the mode select and communication signals for in-circuit debugging and programming.

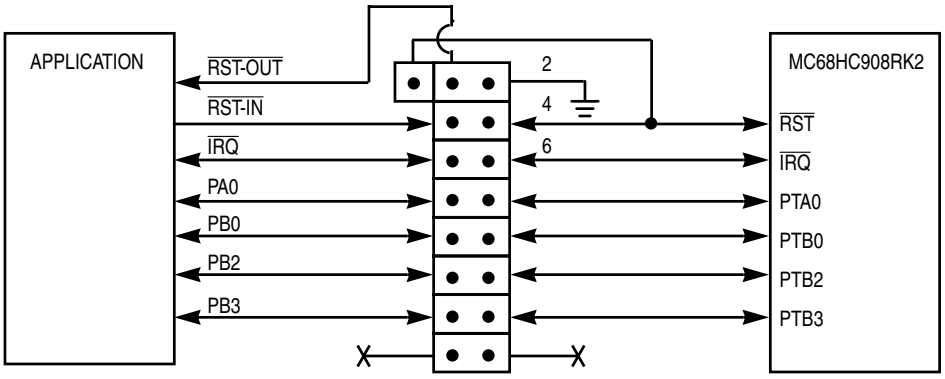


Figure 4-3. Application Designed with a Circuit for MON08

Appendix A. S-Record Information

A.1 Contents

A.2 Overview 49
A.3 S-Record Contents 49
A.4 S-Record Types 51
A.5 S Record Creation 52
A.6 S-Record Example 52
A.6.1 S0 Header Record 52
A.6.2 First S1 Record 53
A.6.3 S9 Termination Record 54
A.6.4 ASCII Characters 54

A.2 Overview

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 provides for editing of the S records and monitoring the cross-platform transfer process.

A.3 S-Record Contents

Each S record 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.

The five fields that comprise an S record are shown in **Table A-1**.

Table A-1. S-Record Fields

Type	Record Length	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 line number generated by some time-sharing systems.

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

A.4 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 ICS08RKZ 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. 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 ICS08RKZ
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.5 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.6 S-Record Example

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

Example:

```
S00600004844521B
S1130000285F245F2212226A00042429008237C2A
S113001000020008000825290018538123410018 13
S113002041E900084#42234300182342000824A952
S107003000144ED492
S9030000FC
```

In the example, the format consists of:

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

A.6.1 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; zeroes

Table A-4. S0 Header Record (Continued)

Field	S-Record Entry	Description
Code/Data	48 44 52	Descriptive information identified these S1 records: ASCII H D R — “HDR”
Checksum	1B	Checksum of S0 record

A.6.2 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.6.3 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
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.6.4 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 here.

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	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

Appendix B. Technical Reference and Troubleshooting

B.1 Contents

B.2 Overview 55
B.3 RKICS Board 56
B.4 Troubleshooting the Start Up 58
B.5 Troubleshooting MON08 Mode 64
B.6 Parts List 68
B.7 Board Layout and Schematic Diagrams 71

B.2 Overview

This appendix provides technical support information for the M68ICS08RK in-circuit simulator kit, including:

- Functional description of the kit
- Troubleshooting the start up procedure
- Troubleshooting MON08 mode
- Connector and cable pin assignments
- Parts list
- Board layout diagram
- Schematic diagrams

Quick Start instructions are included in P&E's software manual.

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 printed circuit boards 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.3 RKICS Board

The core component of the board is the MC68HC908RK2 MCU. This MCU resides either on the RKICS board or on a target system.

When the MCU resides on the board, the board may be used as an in-circuit emulator or simulator for the MC68HC908RK2. For this configuration, a target cable is run from the board to the target system.

The M68ICS08RK supports two kinds of target cables:

- A 20-pin ribbon cable (Motorola part number 01-RE9065W01), terminating in a 20-pin male DIP (dual in-line package) header

NOTE: *The 20-pin ribbon cable is included in the kit and cannot be ordered separately.*

- A flexible target head adapter cable (Motorola part number M68CBL05A), terminating in connectors for one of three target head adapters. This cable must be ordered separately. Four target head adapters (THAs) are available for use with the flex cable:
  - For a 20-pin SOIC-package (small outline integrated circuit) MCU on the target system, use Motorola THA model number M68DIPSOIC.
  - For a 20-pin SSOP-package (shrink small outline package) MCU on the target system, use Motorola THA model number M68DIP20SSOP.
  - For a 32-pin QFP-package (quad flat pack) MCU on the target system, use Motorola THA model number M68TC08RKFU32.
  - For a 20-pin SOIC- or SSOP-package MCU on the target system, use Motorola THA M68TC08RK2P20. Use only with flex cable A, Motorola part number M68CBL05A.

Using a target cable is optional; the board may be utilized with flying leads to other circuits. The MCU can be either the 20-pin SOIC or SSOP version or the 32-pin QFP version. On the RKICS board, socket XU1 supports the 20-pin SSOP package, socket XU2 supports the 20-pin SOIC package, and socket XU3 supports the 32-pin QFP package.

When the MCU resides on a target system, the RKICS board can communicate with the MCU over a 16-pin MON08 cable (Motorola part number 01-RE91008W01). The MON08 cable supports the MCU installed in either the target system or the RKICS.

When using the ICS08RKZ 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 ICS08RKZ software on the host computer lets the host computer become a simulator. When the ICS requires port data, the computer requests the data through the host's serial connection to the core MCU. The core MCU responds by sending the data to the host via the serial connection. It is this arrangement that allows a real-world interface for the in-circuit simulator. The clock runs the MCU at a 4.9512-MHz external clock rate.

When using the PROG08SZ programming software, the MCU's FLASH memory can be programmed. Socket XU1 supports the 20-pin SSOP version of the part, socket XU2 supports the 20-pin SIOC version, and socket XU3 supports the 32-pin QFP version. Only one part may be programmed at a time. The M68ICS08RK also supports in-circuit programming of either version of the part through the MON08 cable.

The RKICS board also provides +5 Vdc power, +7.5 Vdc power for the V_{TST} voltage required to enter monitor mode, a 4.9152-MHz clock signal, and host PC RS-232 level translation.

When using the ICD08SZ debugging software, code can be run directly out of the MCU's internal FLASH at real-time speeds.

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

NOTE: *The M68ICS08RK's emulation of the MC68HC908RK2 is limited. Port A bit 0 (PTA0) is used for host-to-MCU communication. The port bit is not available for connection to a target system. Setting DDRA bits 0 to 1 will stop communications with the simulation or debugger software and will require a system reset to regain communication with the MCU. Port bits PTB0, PTB2, and PTB3 are temporarily disconnected from the target system during reset.*

Emulation of the MC68HC908RK2's $\overline{RST}$ signal is also limited in that the signal is not a bidirectional, open-drain signal. It is emulated as either an input or an output (determined by jumper header W3) when using the target connectors or as two pins (one input and one output) when using the MON08 cable.

B.4 Troubleshooting the Start Up

The installation procedure in **2.4 Installing the Hardware** describes how to prepare the M68ICS08RK for use in the modes where the MCU is installed on the RKICS board.

These modes include:

- Using the M68ICS08RK as an in-circuit simulator/emulator with a target cable
- Using the M68ICS08RK as a programmer
- Using the M68ICS08RK as a stand-alone system without a target board

If difficulties are experienced when starting the kit using the procedure outlined in **2.4 Installing the Hardware**, follow these steps:

1. In these modes, do not use the MON08 cable to a target system. Use the MON08 cable connection only when the MCU is on the target system. Troubleshooting information for the MON08 mode is found in **B.5 Troubleshooting MON08 Mode**.
2. Disconnect any target cables from the board. These troubleshooting steps assume that no target system connections are present.
3. Make sure that the MCU is installed correctly. Verify that only one MCU is installed in either XU1, XU2, or XU3. Insert the MCU with the orientation notch and pin 1 to the upper left in the respective socket.
4. Make sure the board is getting power:
    a. Check the power at the output of the adapter. First disconnect the RKICS from the power supply, then measure the power at the wall adapter's output connector to confirm that it produces +5 Vdc. The

outer barrel of the connector is ground, and the inner sleeve is +5 Vdc. If there is no power at the connector, verify that the adapter is getting power from the ac power outlet.

b. Check the power at the RKICS board. Plug the adapter's output connector into the RKICS. The MCU power LED should light. Check for +5 Vdc at the RKICS's fuse F1. If the LED does not light or if +5 Vdc is not present on fuse F1, check the fuse in the RKICS. If more than +6.2 Vdc or reverse voltage is applied to the RKICS, the fuse will blow.

c. Check the RKICS MCU PWR. Disconnect the RKICS from the power supply and from the host PC. Configure the RKICS board to the factory defaults. Reconnect the power supply to the RKICS. The MCU power LED should light. If the LED does not light, there may be a problem with the RKICS causing too much of a drain on the +5 Vdc supply.

d. Check the MCU power at test point TP2 (MCU VDD) located just below oscillator Y1. Using the side pin on P2 (DC INPUT jack) as the ground reference, check for +3.3 Vdc at TP2.

e. Check the RKICS board's V_{TST} power with the host disconnected. With the RKICS board powered and no host connection to the RKICS, check for the following voltages on the J7 connector of the RKICS board, using the side pin on P1 (DC INPUT jack) as the ground reference:

   Approximately +7.5 Vdc at TP1 — If this voltage is not present when the MCU power LED is lit, there may be a problem with the RKICS's internal step-up power supply.

f. Check the RKICS board's V_{TST} power with the host connected. First, exit any ICS08RKZ software that may be running on the host PC. Then disconnect power from the RKICS. Ensure that the RKICS board is configured for the factory default settings. Ensure that there is an MCU in either XU1, XU2, or XU3 and that it is inserted correctly. Connect the serial cable between the host PC and the RKICS. Apply power to the RKICS. At this point, the RKICS power LED should be lit, and the MCU power LED should be off. If the MCU power LED is on, there may be a problem with the host PC's serial port or the serial cable. See step 5 for communications

problems. If the MCU power LED is off, start the ICS08RKZ simulator software as described in **2.4 Installing the Hardware** while watching the MCU power LED.

If the MCU power LED does not light at all, there may be a problem with the host PC communicating with the board. Refer to step 5.

If the MCU power LED flickers a few times and then goes out, the host PC is able to control the power to the RKICS board but communications may still not be established with the MCU. As the flickering of the MCU power LED indicates, the host PC is applying and removing power to the RKICS board during this period. Use an oscilloscope to view the voltages on TP1 and TP2 as the software tries to establish communication with the MCU. Restart or retry the ICS08RKZ software while looking at the signals. Using the side pin on P1 (DC INPUT jack) as the ground reference, check for a signal that varies between 0 and +3.3 Vdc at TP2 (MCU V_{DD}) and between 0 and +7.5 Vdc at TP1 (V_{TST}). If these voltages are present, the power is good, but communication problems should be investigated as described in step 5.

If the MCU power LED comes on and stays on, communication probably was established with the MCU. Check for the following voltages, using the side pin on P1 (DC INPUT jack) as the ground reference:

- ◊ Approximately +7.5 Vdc at TP1
- ◊ Approximately +3.3 Vdc at TP2

  If these voltages are present, the power is good, and the problem lies elsewhere.

5. Make sure that the host PC can communicate with the MCU:

   a. The MCU's PTA0 pin is used for host communications. DDRA bit 0 should never be set to 1 as this interrupts monitor-mode communications. The target connector PTA0 pins (J3 pin 1, J1 pin 2, and J2 pin 7) are never connected to the MCU's PTA0 pin. They are wired only for probing purposes.

   b. Make sure that the serial cable is correctly attached to the RKICS and to the correct serial port on the host computer.

c. Make sure that the cable is a straight-through cable supporting all nine pins of the serial port connection.

d. Make sure that no hardware security key or other devices are attached to the serial port or cable.

e. Make sure that the host PC supports the minimum speed requirements of the ICS08RKZ software.

f. Make sure to use the correct security code to access the MCU. If the security bytes have been programmed previously, the part will not unlock and enter monitor mode unless the correct security code is sent to the MCU.

g. Check for data at the RKICS end of the serial cable. Pin 3 of this connector carries RS-232 data into the RKICS; pin 2 carries RS-232 data out of the RKICS. Pin 4 controls the MCU power. Pin 5 is ground. While the ICS08RKZ software is trying to establish communications, pins 3 and 4 should both toggle between +10 Vdc and –10 Vdc (or +12 Vdc and –12 Vdc). If these signals are not seen at the cable end, the problem is on the PC and cable side of the system. When connected to the RKICS, a +10 Vdc signal on pin 4 should activate the MCU V_{DD} test point and the MCU power LED.

h. Make sure the serial data is getting to the MCU's PTA0 pin. First, exit any ICS08RKZ software that may be running on the host PC. Then disconnect power from the RKICS. Ensure that the RKICS board is configured for the factory default settings. Ensure that there is an MCU in either XU1, XU2, or XU3 and that it is inserted correctly. Connect the serial cable between the host PC and the RKICS. Apply power to the RKICS. Start the ICS08RKZ simulator software as described in **2.4 Installing the Hardware**. Probe the PTA0 pin (XU1 pin 1, XU2 pin 1, or XU3 pin 2) for the serial data. Since the board power is turned off and on several times during the connecting phase, the data observed at the MCU's PTA0 pin is affected also.

6. Make sure that the MCU has a good clock source. Use an oscilloscope to check the OSC1 input at the MCU (XU1 pin 8, XU2 pin 8, or XU3 pin 21). Set the oscilloscope to 0.1 ms per division. The oscillator should run when the MCU power LED is on. Approximately two divisions per cycle should be observed. This corresponds to a 4.9152-MHz signal, the

frequency required for a 9600-baud communications rate. If the clock signal is not present, check to see that a jumper is installed on Wr. This selects the RKICS as the source of the OSC1 signal.

7. Make sure that the MCU can enter and remain in monitor mode. For this to happen, the following conditions must occur:

   a. At the rising edge of $\overline{RST}$, $\overline{IRQ}$ must be at V_{TST} (+7.5 Vdc). Using a dual-trace oscilloscope, trigger channel 1 on the rising edge of $\overline{RST}$ (XU1 pin 13, XU2 pin 13, or XU3 pin 26) and read the $\overline{IRQ}$ pin (XU1 pin 12, XU2 pin 12, or XU3 pin 25) with Channel 2. Start the ICS08RKZ software as described in **2.4 Installing the Hardware** and verify that the $\overline{IRQ}$ signal is approximately +7.5 Vdc when $\overline{RST}$ rises. If $\overline{IRQ}$ is not at +7.5 Vdc, there may be a problem with the RKICS board's IRQ circuit. Check R1, R15, R16, R27, U10, U12, and U13 for the proper signals to keep $\overline{IRQ}$ at +7.5 Vdc during the period where $\overline{RST}$ is low.

   b. At the rising edge of $\overline{RST}$, PTA0, PTB0, PTB2, and PTB3 must be held at logic values 1, 1, 0, and 0, respectively. The logic levels are +3.3-volt CMOS logic levels (with the factory default setting of +3.3 V on jumper J4). Using a dual-trace oscilloscope, trigger channel 1 on the rising edge of $\overline{RST}$ (XU1 pin 13, XU2 pin 13, or XU3 pin 26) and read the corresponding MCU pin with channel 2. PTA0 (XU1 pin 1, XU2 pin 1, or XU3 pin 2) is the serial data pin to and from the host PC and should be around +3.3 Vdc at the rising edge of $\overline{RST}$. PTB0 (XU1 pin 2, XU2 pin 2, or XU3 pin 39), PTB2 (XU1 pin 4, XU2 pin 4, or XU3 pin 5), and PTB3 (XU1 pin 7, XU2 pin 7, or XU3 pin 20) are controlled by analog switch U2 and should be approximately +3.3 V, 0 V, and 0 V, respectively, at the rising edge of $\overline{RST}$. Port pins PTB0, PTB2, and PTB3 are connected to the target connector pins after the rising edge of $\overline{RST}$ and are then available for target system connections. The MCU's PTA0 pin is never connected to the target pins, as it is used for host communication.

   c. Either $\overline{RST}$ or $\overline{IRQ}$ must remain at +7.5 Vdc to hold the MCU in monitor mode. The RKICS board has an interrupt lock out feature to keep $\overline{IRQ}$ at 7.5 Vdc when the $\overline{RST}$ or $\overline{RST\text{-}IN}$ signal is asserted

(low) and keep it at +7.5 Vdc until after $\overline{RST}$ goes high. The $\overline{TGT\text{-}IRQ}$ signal is allowed to control the $\overline{IRQ}$ signal when $\overline{RST}$ is not asserted.

8. Make sure that external circuitry does not interfere with the monitor mode communications. When connecting external circuitry to the RKICS board, use only the target system connectors J1 and J3. This ensures that the target system will not interfere with the communications and setup of the MCU's monitor mode by allowing the RKICS to disconnect some target system components during monitor mode entry.
9. Make sure that W2 (bus clock divisor) is jumpered correctly. W2 is provided to allow the user to select whether the MCU bus clock is equal to CGMXCLK/2 (W2 pins 2 and 3 jumpered, default setting) or CGMXCLK/4 (W2 pins 1 and 2 jumpered). This jumper affects the power-on baud rate in monitor mode. Currently, only 9600-baud serial communications rate is supported for monitor mode by the simulation and debugging software.
10. When connecting to a target system, observe the setting of W3 (target $\overline{RST}$ direction). W3 is provided to allow the user to select whether the target system can reset the MCU on the RKICS (jumper between pins 2 and 3) or whether the target system receives a reset signal from the RKICS (jumper between pins 1 and 2). $\overline{RST}$ is not a bidirectional, open-drain signal at the target connectors. Removing the jumper leaves the $\overline{RST\text{-}IN}$ signal pulled up to +3.3 V (or the current MCU V_{DD} voltage setting).
11. When connecting to a target system, observe the setting of W1 (target V_{DD} disconnect). W1 is provided to allow the user to select whether the RKICS powers the target system's MCU and external circuitry (W1 jumper on) or whether the target provides the power for its MCU and circuitry (W1 jumper off). The target system V_{DD} must match the RKICS operating voltage in either case.

CAUTION: *Before powering up, remove the W1 jumper if the target system is powered by a source other than the RKICS. Failure to remove the jumper in this case will cause the two power supply outputs to be connected together, possibly causing large currents to flow over the target cable. If the RKICS is to provide power to the target system, ensure that the current drain on the target connector's V_{DD} pins is kept under 100 mA.*

B.5 Troubleshooting MON08 Mode

This section describes the troubleshooting steps for the instances where the MCU is installed on a target system and the RKICS is used to interact with the target system through the MON08 cable. These instances include in-circuit simulation/emulation and FLASH memory programming through the MON08 cable.

1. Disconnect the target system and make sure that the RKICS operates correctly when configured as described in the start up instructions. See **2.4 Installing the Hardware** and **B.4 Troubleshooting the Start Up**.
2. If the start up works, the RKICS should be functioning well enough to place the MCU on the target system into monitor mode.
3. Prepare the RKICS for use with the MON08 cable. Turn off the power to the target system. Exit the RKICS software. Remove the power plug from the RKICS. Remove any MCU from sockets XU1, XU2, and XU3. Install a jumper on W5 unless the target system is using a 4.9152-MHz oscillator. Jumper selections on W1 and W3 have no effect when using the MON08 cable. Install a jumper on W2 pins 2 and 3.
4. Connect the 16-pin cable from J6 on the RKICS to the target system's MON08 connector. Details on designing a MON08 connector for the target system are given in **Table 4-1. MON08 Target System Connector P1**. If cutable jumpers were used on the target board, the jumpers must be cut before using the MON08 cable.
5. The target system (including the MCU) must be externally powered. The target system's MCU V_{DD} must match the MCU V_{DD} setting on the RKICS to communicate with the RKICS. Do not apply power to the target system at this time.
6. Exit any ICS08RKZ software that may be running on the host PC. Connect the serial cable between the host PC and the RKICS. Apply power to the RKICS by connecting the wall adapter's output jack to the RKICS. At this point, the ICS power LED should be lit, and the MCU power LED should be off. If the MCU power LED is on, there may be a problem with the host PC's serial port or the serial cable. Refer to step 9 for information on host communications.

7. Apply power to the target system. At this point, the target MCU should be powered. Check for the appropriate voltage at the MCU's V_{DD} pin. The RKICS should leave the target MCU in reset with approximately 0 Vdc at the MCU's $\overline{RST}$ pin. Verify this at the target MCU's $\overline{RST}$ pin and at J2 pin 4. If $\overline{RST}$ floats too high, the MCU may start and execute code out of its FLASH memory. The RKICS should reset the MCU again in step 8 when the software is started.

8. Start the ICS08RKZ simulator software as described in **2.4 Installing the Hardware** while watching the MCU power LED.

   If the MCU power LED does not light, there may be a problem with the host PC communicating with the RKICS. Continue with step 9.

   If the MCU power LED flickers a few times and then goes out, the host PC is able to control the RKICS but communications may still not be established with the MCU on the target system. As the flickering of the MCU power LED indicates, the host PC is applying and removing power to the RKICS board during this period. Use an oscilloscope to view the voltages on TP1 and TP2 as the software tries to establish communication with the MCU. Restart or retry the ICS08RKZ software while looking at the signals. Using the side pin on P1 (DC INPUT jack) as the ground reference, check for a signal that varies between 0 and +7.5 Vdc at TP1 (V_{TST}) and between 0 and +3.3 Vdc at TP2 (MCU V_{DD}). If these voltages are present, the power is good, but the MCU is not being placed in monitor mode. Continue with step 9.

   If the MCU power LED comes on and stays on, communication probably was established with the MCU. Check for the following voltages, using the side pin on P1 (DC INPUT jack) as the ground reference:

   - ◊ Approximately +7.5 Vdc at TP1
   - ◊ Approximately +3.3 Vdc at TP2

   If these voltages are present, the RKICS power is good. Continue with step 9.

9. Make sure the host PC can communicate with the MCU:

   a. The MCU's PTA0 pin is used for host communications. DDRA bit 0 should never be set to 1, as this interrupts monitor-mode communications. The MON08 pin TGT-PTA0 (J6 pin 7) is never connected to the MCU's PTA0 pin. It is wired to J3 pin 15, J1 pin

10, and J6 pin 7 for probing purposes. On the MON08 connector J6, pin 8 is wired to the MCU's PTA0 pin. Driving this signal with external logic on the target system will interrupt communications.

b. Make sure that the MON08 cable is installed properly between the RKICS and the target system. Pin 1 of each connector on the cable must go to pin 1 of the headers on the RKICS and target system.

c. Make sure that the serial cable is correctly attached to the RKICS and to the correct serial port on the host computer.

d. Make sure that the cable is a straight-through cable supporting all nine pins of the serial-port connection.

e. Make sure that no hardware security key or other device is attached to the serial port or cable.

f. Make sure that the host PC supports the minimum speed requirements of the ICS08RKZ software.

g. Make sure to use the correct security code to access the MCU. If the security byte has been programmed previously, the part will not unlock and enter monitor mode unless the correct security code is sent to the MCU.

h. Make sure the serial data is getting to the MCU's PTA0 pin. Re-start the ICS08RKZ simulator software as described in steps 3 and 4 of the start up instructions. Probe the PTA0 pin of the target MCU for the serial data. Since the board power is turned off and on several times during the connecting phase, the data observed at the MCU's PTA0 pin is also affected.

i. Make sure that the target MCU has a good clock source. Use a clock rate that gives a 9600-baud serial communications rate for monitor mode on the target system. Use an oscilloscope to check the OSC2 output at the MCU. Set the oscilloscope to 0.1 ms per division. The oscillator should run when the MCU power LED is on. There should be approximately two divisions per cycle. This corresponds to a 4.9152-MHz signal, the frequency required for a 9600-baud communications rate. If the clock signal is not present, check to see that a jumper is installed on W5. This selects the RKICS as the source of the OSC1 signal.

10. Make sure that the MCU can enter and remain in monitor mode. For this to happen, the following conditions must occur:

    a. At the rising edge of $\overline{RST}$, the target MCU's $\overline{IRQ}$ pin must be at V_{TST} (+7.5 Vdc). Using a dual-trace oscilloscope, trigger channel 1 on the rising edge of the MCU's $\overline{RST}$ pin and read the $\overline{IRQ}$ pin with channel 2. Start the ICS08RKZ software as described in **2.4 Installing the Hardware** and verify that the $\overline{IRQ}$ signal is approximately +7.5 Vdc when $\overline{RST}$ rises.

    b. At the rising edge of $\overline{RST}$, PTA0, PTB0, PTB2, and PTB3 must be held at logic values 1, 1, 0, and 0, respectively. The logic levels are +3.3-volt CMOS logic levels (with the factory default setting of +3.3 V on jumper J4). Using a dual-trace oscilloscope, trigger channel 1 on the rising edge of $\overline{RST}$ and read the corresponding MCU pin with channel 2. PTA0 is the serial data pin to and from the host PC and should be around +3.3 Vdc at the rising edge of $\overline{RST}$. PTB0, PTB2, and PTB3 are controlled by analog switch U2 on the RKICS and should be approximately +3.3 V, 0 V, and 0 V, respectively, at the rising edge of $\overline{RST}$. After the rising edge of $\overline{RST}$, the MCU pins PTB0, PTB2, and PTB3 are connected (by the RKICS) to the MON08 connector pins TGT-PTB0, TGT-PTB2, and TGT-PTB3, respectively. The MCU's PTA0 pin is never connected to the target pins because it is used for host communication.

    c. Either $\overline{RST}$ or $\overline{IRQ}$ must remain at +7.5 Vdc to hold the MCU in monitor mode. The RKICS board has an $\overline{IRQ}$ lockout feature to keep $\overline{IRQ}$ at 7.5 Vdc when the $\overline{RST}$ or $\overline{RST\text{-}IN}$ signal is asserted (low) and to keep it at +7.5 Vdc until after $\overline{RST}$ goes high. The $\overline{TGT\text{-}IRQ}$ signal is allowed to control the $\overline{IRQ}$ signal when $\overline{RST}$ is not asserted.

11. Make sure that the target circuitry does not interfere with the monitor mode communications. When connecting target circuitry to the MCU, be sure to connect the circuits through the RKICS by connecting to the $\overline{RST\text{-}OUT}$, $\overline{RST\text{-}IN}$, $\overline{TGT\text{-}IRQ}$, TGT-PTA0, TGT-PTB0, TGT-PTB2, and TGT-PTB3 pins of the MON08 connector. These signals will be connected by the RKICS to the corresponding pins of the MCU through the corresponding MON08 connector pins — $\overline{RST}$, $\overline{IRQ}$, PTB0, PTB2,

and PTB3 — after monitor mode is established. TGT-PTA0 is never connected to PTA0 because the PTA0 signal is used for host communications.

B.6 Parts List

The parts list for the RKICS board is given in **Table B-1**.

Table B-1. RKICS Parts List (Sheet 1 of 3)

Reference Designator	Description	Manufacturer	Part Number
C1, C2, C3, C4, C6, C7, C8, C10, C11, C12, C13, C15, C17, C19, C20, C21, C22, C23, C26, C27, C29, C31, C33, C34, C35	Capacitor, 0.1 µF, ceramic, Z5U, 50 V	AVX Corporation	08055E104ZAT2A
C14, C18, C28, C30	Capacitor, 10 µF, tantalum, 16 V	AVX Corporation	TAJC106M016
C16	Capacitor, 2.2 µF, tantalum, 16 V	AVX Corporation	AJA225M016
C24	Capacitor, 560 pF, ceramic, COG, 50 V	AVX Corporation	08055A561KAT2A
C9, C25	Capacitor 47 µF, tantalum, low-ESR, 16 V	AVX Corporation	TPSD476M016R0150
D1	Diode, zener, 1SMA6.0AT3	Motorola	1SMA6.0AT3
D2, D8	Diode, Schottky, MBRA130	Motorola	MBRA130
D3	LED, green, 5 mm	Dialight Corp.	521-9173
D4, D5, D6	Diode, MMBD914	Motorola	MMBD914
D7	Diode, Schottky MBR0520	Motorola	MBR0520
D9	LED, yellow, 5 mm	Dialight Corp.	521-9174
F1	Fuse clip, 5X20 mm, GMA1A	Littelfuse	100-054
J1	Header, 2X20, 100 MIL, SHRD	3M	2540-6002-UG
J2	Header, 8X2, 100 MIL, SHRD	3M	2516-6002-UG
J3	Socket, DIP, narrow, MACH, 20P	Aries Electronics	20-3518-10
J4	Header, 2X6, 100 MIL	3M	2412-6122-TG
L1	Inductor, 180 H	Murata Electronics	LQH4N181K04

Table B-1. RKICS Parts List (Sheet 2 of 3)

Reference Designator	Description	Manufacturer	Part Number
L2	Inductor, 10 H	Murata Electronics	LQH1N100K04
P1	Power jack, 2.5 mm	CUI Stack	PJ-002B
P2	Connector, DB9, socket, R/A	AMP, Inc.	AMP-9726-A
Q1	Transistor, NPN, MMBT3904	Motorola	MMBT3904
Q2, Q4	Transistor, PNP, MMBT3906	Motorola	MMBT3906
Q3, Q5	Transistor, PFET, MMBF0201	Motorola	MMBF0201
R1, R8, R9, R12, R14, R15, R16, R24, R27, R28	Resistor, 3.3 kΩ, 5%	Dale	CRCW0805332J
R17	Resistor,10.7 kΩ, 1%	Dale	CRCW08051072F
R18, R19, R20	Resistor, 10 Ω, 1%	Dale	CRCW080510R0F
R2, R22	Resistor, 470 Ω, 5%	Dale	CRCW0805471J
R21	Resistor, 150 Ω, 1%	Dale	CRCW08051500F
R23	Resistor, 53.6 kΩ, 1%	Dale	CRCW08055362F
R29,R35	Resistor, 205 Ω, 1%	Dale	CRCW08052050F
R30	Resistor, 124 Ω, 1%	Dale	CRCW08051240F
R31	Resistor, 1.21 kΩ, 1%	Dale	CRCW08051211F
R32	Resistor, 499 Ω, 1%	Dale	CRCW08054990F
R33	Resistor, 316 Ω, 1%	Dale	CRCW08053160F
R34	Resistor, 118 Ω, 1%	Dale	CRCW0805118R0F
R37	Resistor, 100 kΩ, 5%	Dale	CRCW0805104J
R39	Potentiometer, 5 kΩ, 3/8-inch square, single-turn, CERMET	Bourns, Inc.	3386P-1-502
R4, R7, R11	Resistor, 1 kΩ, 5%	Dale	CRCW0805102J
R5, R6, R10, R13, R25, R26, R36	Resistor, 10 kΩ, 5%	Dale	CRCW0805103J
U11	IC, single NAND gate, open drain	Fairchild Semiconductor	NC7SZ38M5

Table B-1. RKICS Parts List (Sheet 3 of 3)

Reference Designator	Description	Manufacturer	Part Number
U12	IC, single OR gate	Fairchild Semiconductor	NC7SZ32M5
U14	IC, switch mode controller	Motorola	MC34063AD
U15	IC, voltage regulator	Linear Technology	LT1086CM
U2	IC, quad, analog, multiplexer	Analog Devices, Inc.	ADG774BR
U3	IC, RS323-XCVR	Linear Technology	LTC1386CS
U4, U5, U10, U16	IC, single inverter	Fairchild Semiconductor	NC7SZ04M5
U6	IC, single TS buffer	Fairchild Semiconductor	NC7SZ125M5
U7, U9, U13	IC, single inverter, open drain	Fairchild Semiconductor	NC7SZ05M5
U8	IC, low-voltage detect	Motorola	MC34164D-5
W1, W5	HDR, 2X1,100 MIL	3M	2402-6112-TG
W2, W3	HDR, 3X1, 100 MIL	3M	2403-6112TG
XF1	Fuse, 1 amp, fast acting	Littelfuse	217001
XU1	Socket, SSOP20, open top	Enplas-Tesco, Inc.	OTS-20(34)-0.65-01
XU2	Socket, SOIC20, open top	Enplas-Tesco, Inc.	OTS-20(28)-1.27-04
XU3	Socket, QFP32, clam shell	Yamaichi Electronics USA, Inc.	IC51-0324-354
XY1	Socket, DIP8, XTAL	Aries Electronics	1108800
Y1	XTAL, 9.8304-MHz, 1/2 metal CAN	ECS, Inc.	OECS-98.3-2-A401A
N/A	Printed wiring board		84-RE32430W01 Rev. 0

B.7 Board Layout and Schematic Diagrams

Figure B-1 shows the RKICS board layout and component locations.

The RKICS schematic diagrams follow.

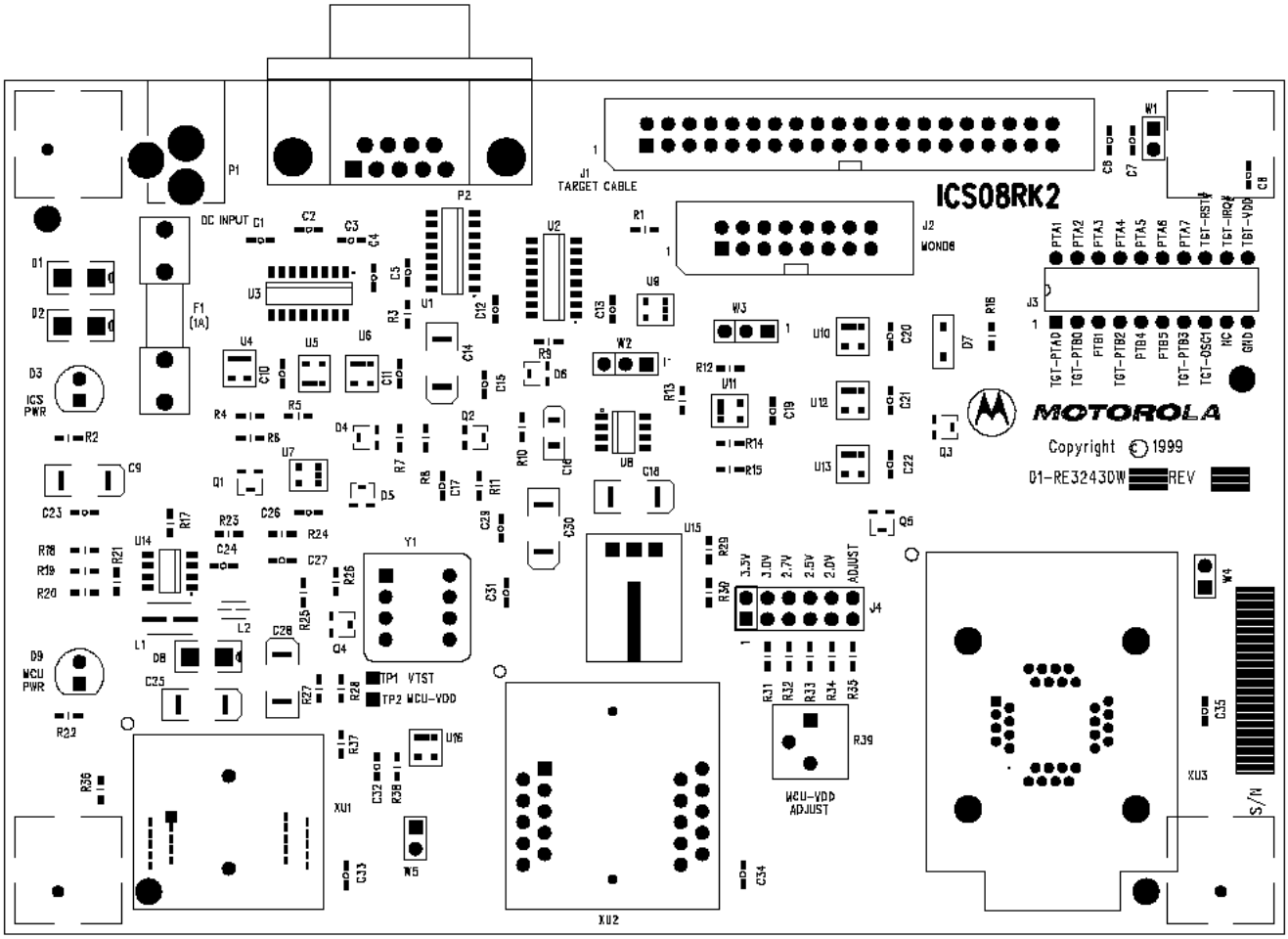


Figure B-1. M68IC508RK2 Board Layout

Modified after FAT and Motorola testing

REVISIONS

ZONE	REV	DESCRIPTION	DATE	APPROVED
	O	Original Revision	02/24/99	
	A	Modified after FAT and Motorola testing	04/21/99	
	B	Changed oscillator (Y1) to 9.8304 MHz.	12/29/99	Bill Kauffman

NOTES:

1. UNLESS OTHERWISE SPECIFIED:
   RESISTANCE VALUES ARE IN OHMS.
   RESISTORS ARE 1/8 WATT, 5%.
   CAPACITANCE VALUES ARE IN MICROFARADS.
2. INTERRUPTED LINES CODED WITH THE SAME LETTER OR LETTER COMBINATIONS ARE ELECTRICALLY CONNECTED.
3. DEVICE TYPE NUMBER IS FOR REFERENCE ONLY. THE NUMBER VARIES WITH THE MANUFACTURER.
4. SPECIAL SYMBOL USAGE:
   # DENOTES ACTIVE-LOW SIGNAL.
   [ ] DENOTE VECTORED SIGNALS.
5. INTERPRET DIAGRAM IN ACCORDANCE WITH AMERICAN NATIONAL STANDARDS INSTITUTE SPECIFICATIONS, CURRENT REVISION, WITH THE EXCEPTION OF LOGIC BLOCK SYMBOLOGY.

CHANGES:

03/22/99 - Replaced Motorola logo in title blocks

04/06/99 - Added 'B' to document number size code field

04/21/99 - Added jumper W5, U16, R39-40 and C34-35

04/27/99 - Back-annotated resequenced reference designators from PCB

12/29/99 - Changed oscillator (Y1) to 9.8304MHz.

COMPUTER GENERATED DRAWING : DO NOT REVISE MANUALLY

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.

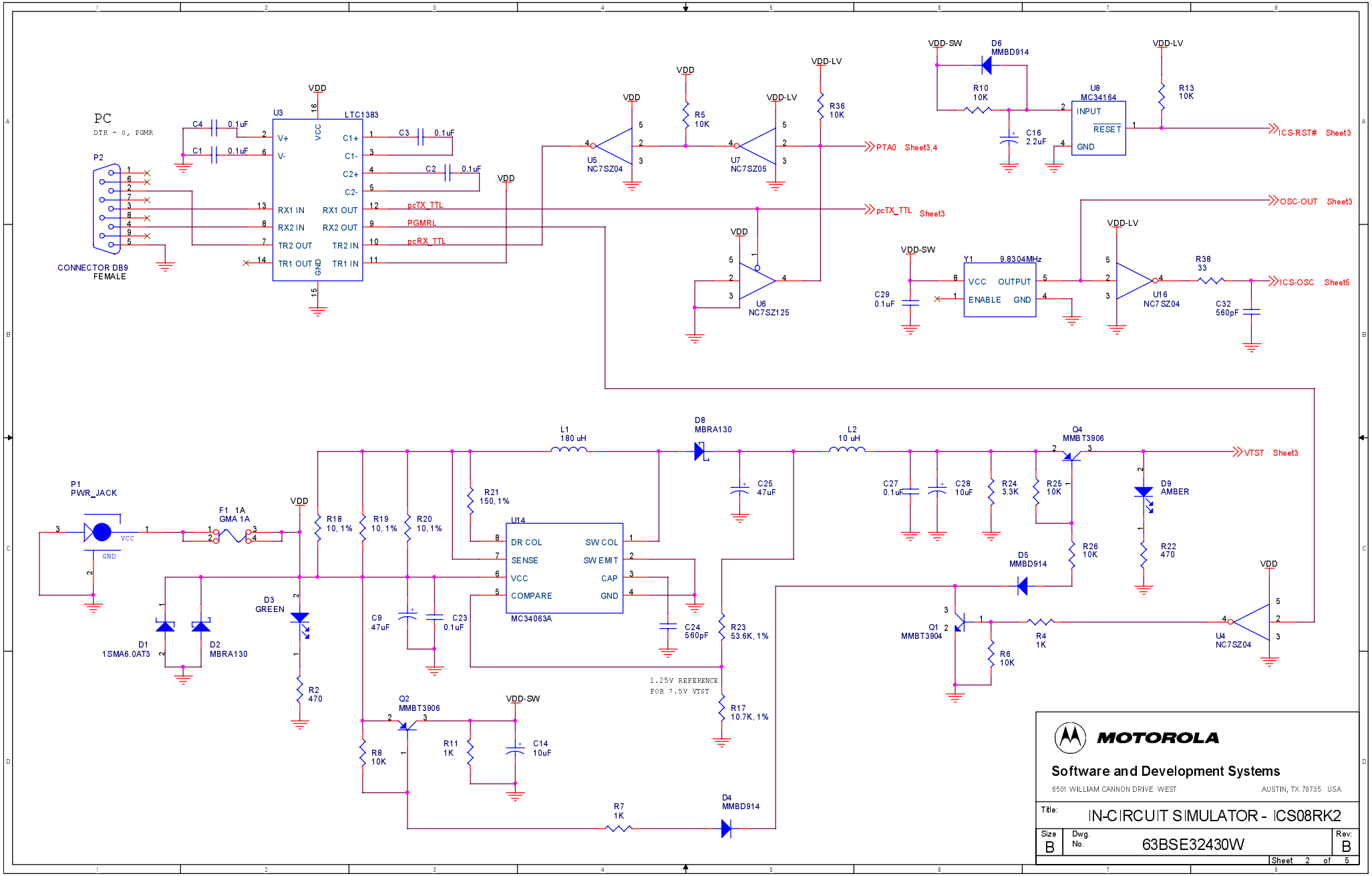
DRAWN:	MIKE SMITH	DATE:	04/21/99
CHECKED:		DATE:	
APPROVED:		DATE:	

MOTOROLA
Software and Development Systems
6501 WILLIAM CANNON DRIVE WEST AUSTIN, TX 78735 USA

Title: IN-CIRCUIT SIMULATOR - ICS08RK

Size	Dwg. No.	Rev:
B	63BSE32430W	B

Sheet 1 of 5

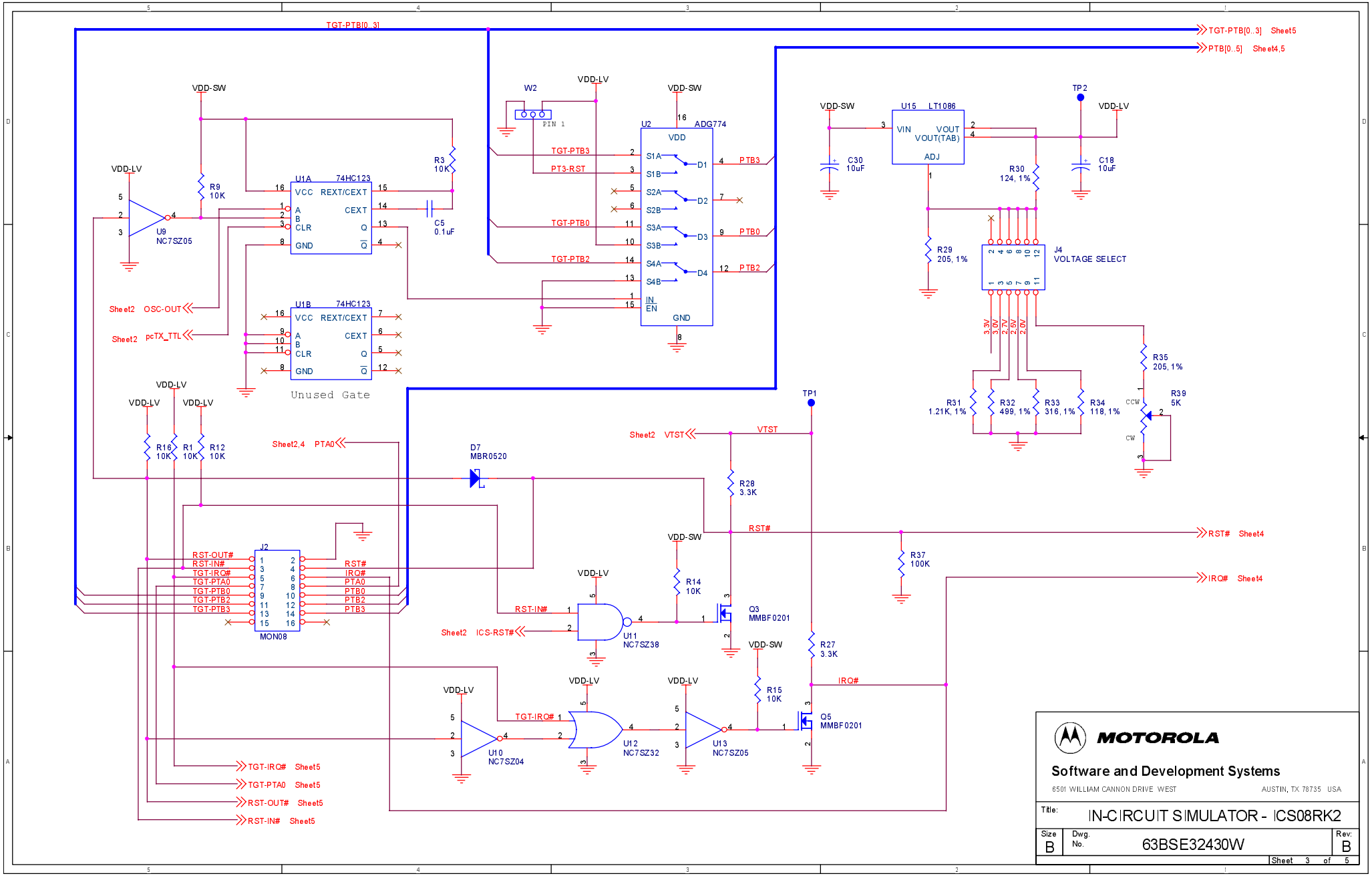


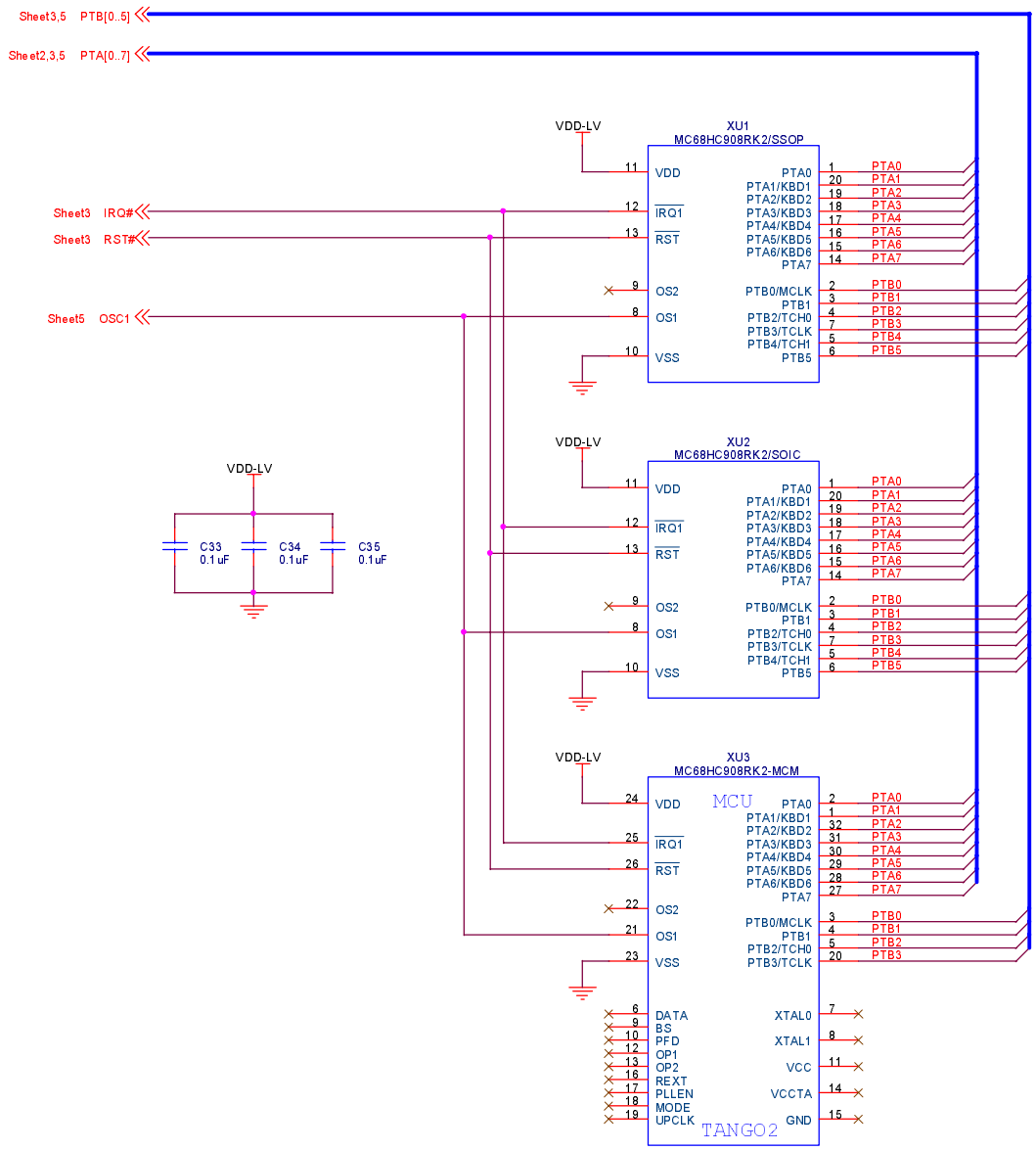
MOTOROLA

Software and Development Systems

6501 WILLIAM CANNON DRIVE WEST AUSTIN, TX 78735 USA

Title: IN-CIRCUIT SIMULATOR - ICS08RK2		
Size B	Dwg. No. 63BSE32430W	Rev: B
		Sheet 2 of 5


TGT-PTB[0..3]
TGT-PTB[0..3] Sheet5
PTB[0..5] Sheet4,5
VDD-SW
VDD-LV
W2
PIN 1
U2 ADG774
VDD
S1A
S1B
S2A
S2B
S3A
S3B
S4A
S4B
IN
EN
GND
D1
D2
D3
D4
TGT-PTB3
PT3-RST
TGT-PTB0
TGT-PTB2
PTB3
PTB0
PTB2
U15 LT1086
VIN
VOUT
VOUT(TAB)
ADJ
TP2
C30 10uF
C18 10uF
R30 124, 1%
R29 205, 1%
J4 VOLTAGE SELECT
3.3V
3.0V
2.7V
2.5V
2.0V
R35 205, 1%
R39 5K
CCW
CW
R31 1.21K, 1%
R32 499, 1%
R33 316, 1%
R34 118, 1%
R9 10K
U9 NC7SZ05
U1A 74HC123
VCC
REXT/CEXT
CEXT
CLR
GND
R3 10K
C5 0.1uF
U1B 74HC123
Unused Gate
Sheet2 OSC-OUT
Sheet2 pcTX_TTL
R16 10K
R1 10K
R12 10K
Sheet2,4 PTA0
D7 MBR0520
TP1
Sheet2 VTST
VTST
R28 3.3K
RST#
RST# Sheet4
R37 100K
IRQ# Sheet4
J2
MON08
RST-OUT#
RST-IN#
TGT-IRQ#
TGT-PTA0
TGT-PTB0
TGT-PTB2
TGT-PTB3
RST#
IRQ#
PTA0
PTB0
PTB2
PTB3
R14 10K
Q3 MMBF0201
U11 NC7SZ38
Sheet2 ICS-RST#
R27 3.3K
R15 10K
Q5 MMBF0201
U10 NC7SZ04
U12 NC7SZ32
U13 NC7SZ05
TGT-IRQ# Sheet5
TGT-PTA0 Sheet5
RST-OUT# Sheet5
RST-IN# Sheet5
MOTOROLA
Software and Development Systems
6501 WILLIAM CANNON DRIVE WEST
AUSTIN, TX 78735 USA
Title: IN-CIRCUIT SIMULATOR - ICS08RK2
Size B
Dwg. No. 63BSE32430W
Rev: B
Sheet 3 of 5

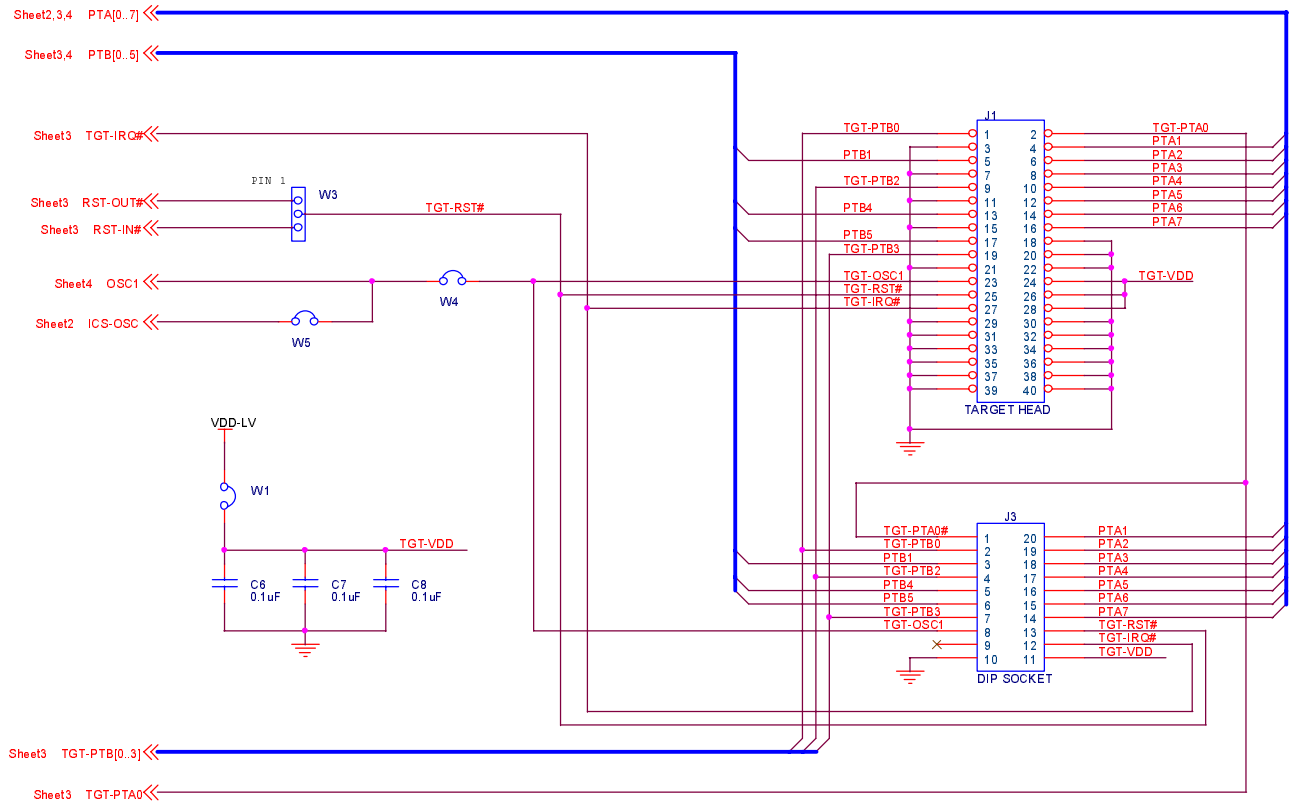


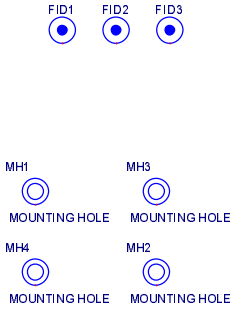
MOTOROLA

Software and Development Systems

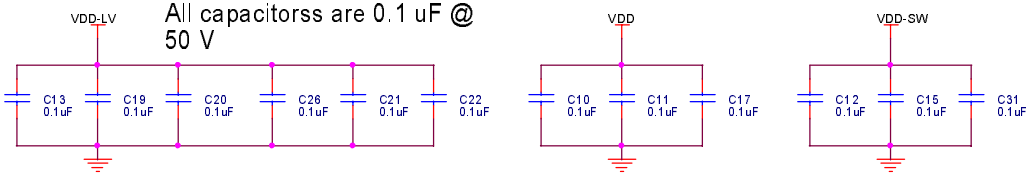
6501 WILLIAM CANNON DRIVE WEST AUSTIN, TX 78735 USA

Title:	IN-CIRCUIT SIMULATOR - ICS08RK2	
Size B	Dwg. No. 63BSE32430W	Rev: B
		Sheet 4 of 5





Decoupling capacitors for ICs
All capacitorss are 0.1 uF @ 50 V



MOTOROLA

Software and Development Systems

6501 WILLIAM CANNON DRIVE WEST AUSTIN, TX 78735 USA

Title: IN-CIRCUIT SIMULATOR - ICS08RK2		
Size B	Dwg. No. 63BSE32430W	Rev: B
		Sheet 5 of 5

Glossary

8-bit MCU — A microcontroller whose data is communicated over a data bus made up of eight separate data conductors. Members of the MC68HC908 Family of microcontrollers are 8-bit MCUs.

A — An abbreviation for the accumulator of the MC68HC908RK2 MCU.

accumulator — An 8-bit register of the MC68HC908RK2 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.

C — An abbreviation for carry/borrow in the condition codes register of the MC68HC908RK2. When adding two unsigned 8-bit numbers, the C bit is set if the result is greater than 255 ($FF).

CCR — An abbreviation for condition code register in the MC68HC908RK2. The CCR has five bits (H, I, N, Z, and C) that can be used to control conditional branch instructions. The values of the bits in the CCR are determined by the results of previous operations. For example, after a load accumulator (LDA) instruction, Z will be set if the loaded value was $00.

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.

condition codes register — The CCR has five bits (H, I, N, Z, and C) that can be used to control conditional branch commands. The values of the bits in the CCR are determined by the results of previous operations. For example, after a load accumulator (LDA) instruction, Z will be set if the loaded value was $00.

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 MC68HC908RK2, 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.

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.

H — Abbreviation for half-carry in the condition code register of the MC68HC908RK2. This bit indicates a carry from the low-order four bits of an 8-bit value to the high-order four bits. This status indicator is used during BCD calculations.

I — Abbreviation for interrupt mask bit in the condition code register of the MC68HC908RK2.

index register — An 8-bit CPU register in the MC68HC908RK2 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.

MSB — Most significant bit.

N — Abbreviation for negative, a bit in the condition code register of the MC68HC908RK2. 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 MC68HC908RK2 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 MC68HC908RK2.

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

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 MC68HC908RK2 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.

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 MC68HC908RK2 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.

$\mathbf{V_{DD}}$ — The positive power supply to a microcontroller (typically 5 volts dc).

V_{SS} — 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 MC68HC908RK2.

Z — Abbreviation for zero, a bit in the condition code register of the MC68HC908RK2. 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 54
assembler
description. 16

B

baud rates 23
board layout 20
bus frequency 23
bus frequency selection. 23

C

cables
flex target 42
target head adapter 56
CASM08Z. 16
description. 16
checksum. 49
communications 23
communications, board-to-host. 17
connectors
Target 37
Target DIP. 40

D

debuggers
ICD08SZ
description. 17
MON08 interface 33, 45
drive space 17

E

electrostatic discharge 55
emulation
    RF 23
examples
    S records 52

F

features 16
FLASH memory 14, 16, 33, 57, 64, 65
function keys 17

H

hardware
    installation 20
    requirements 17
    specifications 18
hotkeys. See function keys
humidity 18

I

I/O 14
ICD08SZ
    description 16
ICG
    MON08 limitations to 23
ICS PWR LED 32
ICS08RK software 15
ICS08RKZ
    description 16
integrated development environment 16
interactive mode 20

L

LED
    ICS power 32, 64
    MCU power 59, 60, 61, 64, 66
    RKICS power 59

limitations
ICG MON08 23
LVI MON08 23
RKICS 23
LVI
MON08 limitations to 23

M

M68ICS08RK in-circuit simulator
components 15
description 13
features 16
hardware 16
hardware specifications 18
M68ICS08RK2 board 15, 20, 32, 33, 36, 42
board layout 20
MC68HC908RFRK2 23
MC68HC908RK2 MCU 15, 35
MCU 13, 15, 35
MCU subsystem
clocks 26
host system connector 31
level translation 36
memory
address 49
system requirements 17
MON08 . . . . . 15, 33, 42, 57, 58, 64, 65, 67

O

operating systems 17

P

parts list 68
port A0 23
ports, serial 17
power
connector 32
LEDs
ICS PWR LED 64
MCU PWR LED 59, 60, 61, 64, 66

RKICS PWR LED 59
requirements 18
power connector 32
PROG08SZ
description. 16

Q

QFP socket 23

R

RAM 17
record length 49
record type 49
relative humidity 18
requirements
hardware 17
host computer 17
operating system 17
software. 17
RF emulation. 23
RF features
MC68HC908RFRK2 23
RKICS
limitations 23
software. 15
RKICS board. 58, 59, 64, 71
components 56–58
parts list. 68
RKICS limitations. 23
RS-232 serial connector 32

S

S records 49–54
serial port
connector. 32
use 17
software
ICS08RK (or RKICS software) 16
requirements 17
RKICS (or ICS08RK software) 16

specifications, hardware 18
S-record
content. 49
creating 52
field contents. 50
fields 50
overview 49
S0 record. 52
S1 record. 53
S9 record. 54
termination record. 54
types 51
stand-alone mode 20

T

target system
cables. 42–43, 56
connecting to. 14, 33, 45, 47, 56, 57
connectors 36–42, 45
description. 14
MCU 56, 57
MON08 interface 45, 58, 64
power source 63
stand-alone mode 47
target head adapter 56
temperature
operating 18
storage. 18

W

Windows 95 14
Windows 98 14
WinIDE. 16
description. 16

Index

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

© Motorola, Inc., 2000

M68ICS08RKUM/D