Dr.63514

In-Circuit Emulator

User's Manual

ML63512/514 Program Development Support System

Second Edition, Feb 5, 1999

♦ ***This manual contains important information pertaining to the safe use of the above product. Before using the product, read these safety notes thoroughly and then keep this manual handy for immediate reference.***

Oki Electric Industry Co., Ltd.

This manual describes the setup and operation of the Dr.63514 in-circuit emulator, the hardware portion of the Dr.63514 development support system for developing user application programs for Oki Electric's ML63512/514 of 4-bit CMOS microcontrollers.

NOTICE

1. The information contained herein can change without notice owing to product and/or technical improvements. Before using the product, please make sure that the information being referred to is up-to-date.
2. The outline of action and examples for application circuits described herein have been chosen as an explanation for the standard action and performance of the product. When planning to use the product, please ensure that the external conditions are reflected in the actual circuit and assembly designs.
3. When developing and evaluating your product, please use our product below the specified maximum ratings and within the specified operating ranges including, but not limited to, operating voltage, power dissipation, and operating temperature.
4. **OKI assumes no responsibility or liability whatsoever for any failure or unusual or unexpected operation resulting from misuse, neglect, improper installation, repair, alteration or accident, improper handling, or unusual physical or electrical stress including, but not limited to, exposure to parameters beyond the specified maximum ratings or operation outside the specified operating range.**
5. Neither indemnity against nor license of a third party's industrial and intellectual property right, etc. is granted by us in connection with the use of product and/or the information and drawings contained herein. No responsibility is assumed by us for any infringement of a third party's right which may result from the use thereof.
6. The products listed in this document are intended only for use in development and evaluation of control programs for equipment and systems. These products are not authorized for other use (as an embedded device and a peripheral device).
7. Certain products in this document may need government approval before they can be exported to particular countries. The purchaser assumes the responsibility of determining the legality of export of these products and will take appropriate and necessary steps at their own expense for these.
8. No part of the contents contained herein may be reprinted or reproduced without our prior permission.
9. MS-DOS is a registered trademark of Microsoft Corporation.

Table of Contents

Preface ---------- 0-1
1. Product Inquiries ---------- 0-2
2. Using this Product Safely and Properly ---------- 0-3
2.1 Important Safety Notes ---------- 0-4
3. Notation ---------- 0-6
4. Manual Organization ---------- 0-7
5. Package Contents ---------- 0-8
5.1 Verify Shipping Contents ---------- 0-8

Chapter 1. Overview ---------- 1-1
1. Overview ---------- 1-2
2. Package Components ---------- 1-3
3. Configurations ---------- 1-4
4. Names of Parts ---------- 1-5

Chapter 2. Functions ---------- 2-1
1. Emulator Specifications ---------- 2-2
2. Functions ---------- 2-5
2.1 Configuring for Target Device ---------- 2-5
2.2 Evaluation Operation ---------- 2-6
2.2.1 Overview ---------- 2-6
2.2.2 Operation ---------- 2-6
2.3 Emulation Operation ---------- 2-8
2.3.1 Single-Step Emulation ---------- 2-8
2.3.2 Real-time Emulation ---------- 2-9
2.3.2.1 Breaks with Parameters 2-11
2.3.2.2 Breaks on Specific Conditions 2-16
2.3.2.3 Forced Breaks 2-20
2.4 Code Memory Operations ---------- 2-22
2.4.1 Data Operations between Code Memory and Disk Files ---------- 2-22
2.4.2 Data Operations between Code Memory and EPROMs ---------- 2-22
2.4.3 Displaying/Changing/Moving Code Memory ---------- 2-23
2.4.4 Code Memory Backup ---------- 2-23
2.4.5 Expanding Code Memory ---------- 2-24
2.5 Real-time Tracing ---------- 2-25
2.5.1 Trace Entries ---------- 2-26
2.5.2 Real-time Trace Control ---------- 2-26
2.5.3 Displaying/Searching Trace Entries ---------- 2-29
2.6 Profiling ---------- 2-30
2.6.1 Instruction Executed (IE) Bits ---------- 2-30
2.6.2 Cycle Counter ---------- 2-31

2.7 Probe Cable ---------- 2-33
2.7.1 Break Signal Input ---------- 2-33
2.7.2 Sync Out Signal ---------- 2-34
2.7.3 Trace Inputs ---------- 2-35
2.8 Clock Switching ---------- 2-36
2.9 Reset Input Switching ---------- 2-38
2.10 Mask Option ---------- 2-39
2.11 Comparator Selection Function ---------- 2-40
2.12 Backup Switching ---------- 2-41
2.13 Package Type Setting ---------- 2-42
2.14 Operating power supply selection function ---------- 2-43
2.15 Internal Signal Monitoring ---------- 2-44
2.16 BAUD Switches ---------- 2-45
2.17 LED Indicator ---------- 2-46
2.18 Power Supplies ---------- 2-47

Chapter 3. Setting Up and Starting Up ---------- 3-1
1. Setting up and turning on the Dr.63514 in-circuit emulator ---------- 3-2
1.1 Device configuration ---------- 3-2
1.2 Switch and Jumper Settings ---------- 3-3
1.3 Emulator Connections ---------- 3-6
1.4 Powering Up ---------- 3-7

Chapter 4. Additional Usage Notes ---------- 4-1
1. Debugging Notes ---------- 4-2
2. Initialization ---------- 4-10
3. Operation Timing ---------- 4-11

Appendices ---------- A-1
1. User Cable Connector Layout ---------- A-2
2. User cable layout ---------- A-3
3. User cable pin arrange ---------- A-4
4. Probe Cable Connectors and Pin Layout ---------- A-6
5. RS232C Cable Wiring Diagrams ---------- A-8
6. RS232C Interface Circuit ---------- A-10
7. Installing EPROMs ---------- A-11
8. If Emulator Doesn't Start ---------- A-13

Preface

1. Product Inquiries

Thank you for purchasing the Dr.63514 Development Support System. Please direct any comments or questions that you may have about this product to your nearest Oki Electric Industry representative.

2. Using this Product Safely and Properly

This User's Guide uses various labels and icons that serve as your guides to operating this product safely and properly so as to prevent death, personal injury, and property damage. The following table lists these labels and their definitions.

Labels

Warning	This label indicates precautions that, if ignored or otherwise not completely followed, could lead to death or serious personal injury.
Caution	This label indicates precautions that, if ignored or otherwise not completely followed, could lead to personal injury or property damage.

Icons

A triangular icon draws your attention to the presence of a hazard. The illustration inside the triangular frame indicates the nature of the hazard—in this example, an electrical shock hazard.

A circular icon with a solid background illustrates an action to be performed. The illustration inside this circle indicates this action—in this example, unplugging the power cord.

A circular icon with a crossbar indicates a prohibition. The illustration inside this circle indicates the prohibited action—in this example, disassembly.

2.1 Important Safety Notes

Please read this page before using the product.

Warning	
Use only the specified voltage. Using the wrong voltage risks fire and electrical shock.	
At the first signs of smoke, an unusual smell, or other problems, unplug the emulator and disconnect all external power cords. Continued use risks fire and electrical shock.	
Do not use the product in an environment exposing it to moisture or high humidity. Such exposure risks fire and electrical shock.	
Do not pile objects on top of the product. Such pressure risks fire and electrical shock.	
At the first signs of breakdown, immediately stop using the product, unplug the emulator, and disconnect all external power cords. Continued use risks fire and electrical shock.	

Please read this page before using the product.

⚠ Caution	
Do not use this product on an unstable or inclined base as it can fall or overturn, producing injury.	
Do not use this product in an environment exposing it to excessive vibration, strong magnetic fields, or corrosive gases. Such factors can loosen or even disconnect cable connectors, producing a breakdown.	
Do not use this product in an environment exposing it to temperatures outside the specified range, direct sunlight, or excessive dust. Such factors risk fire and breakdown.	
Use only the cables and other accessories provided. Using non-compatible parts risks fire and breakdown.	

Please read this page before using the product.

⚠ Caution	
Do not use the cables and other accessories provided with other systems. Such improper usage risks fire.	
Do not exceed the rated input voltage for the user cable VDD pin. Doing so risks fire and breakdown.	
Always observe the specified order for turning equipment on and off. Using the incorrect order risks fire and breakdown.	
Always cut the power to the emulator before altering connections. Connection or disconnection with the power on risks fire and breakdown.	
Always cut the power to the emulator and the user application system before altering connections between the two. Connection or disconnection with the power on risks fire and breakdown.	

3. Notation

This User's Guide uses the following notational conventions.

Type	Notation	Meaning
Numerals	xxh, xxH	Hexadecimal number
	xxb	Binary number
Units	W (word)	1 word = 2 bytes = 4 nibbles = 16 bits
	B (byte)	1 byte = 2 nibbles = 8 bits
	N (nibble)	1 nibble = 4 bits
	M (mega-)	10^6
	K (kilo-)	1024 — only in KB (kilobytes) and KW (kilowords)
	k (kilo-)	$10^3 = 1000$
	m (milli-)	10^{-3}
	μ (micro-)	10^{-6}
	n (nano-)	10^{-9}
	s	second(s)
Terms	"H" level	High signal level — that is, the VDD voltage level.
	"L" level	Low signal level — that is, the VSS voltage level.
Cross References	■ **Reference** ■	This notation gives a cross-reference to related material elsewhere in this manual.
	(See Note n)	This notation refers the reader to a numbered note providing supplementary information later in the same Section.
	■ **Note n** ■	This notation introduces a numbered note providing supplementary information.

4. Manual Organization

This manual consists of the following four chapters.

Chapter 1. Overview
This chapter introduces the emulator and its parts.

Chapter 2. Functions
This chapter describes the functions of the emulator.

Chapter 3. Setting and Starting Up
This chapter describes configuring the emulator and powering it up.

Chapter 4. Additional Usage Notes
This chapter contains important usage notes. Be sure to read it before using the emulator.

Appendices

5. Package Contents

5.1 Verify Shipping Contents

When you receive your Dr.63514 development support system, check the package contents against the Dr.63514 packing list.

Oki Electric has every confidence that the contents are both complete and undamaged. Should a component be damaged or missing, however, please contact your nearest Oki Electric representative.

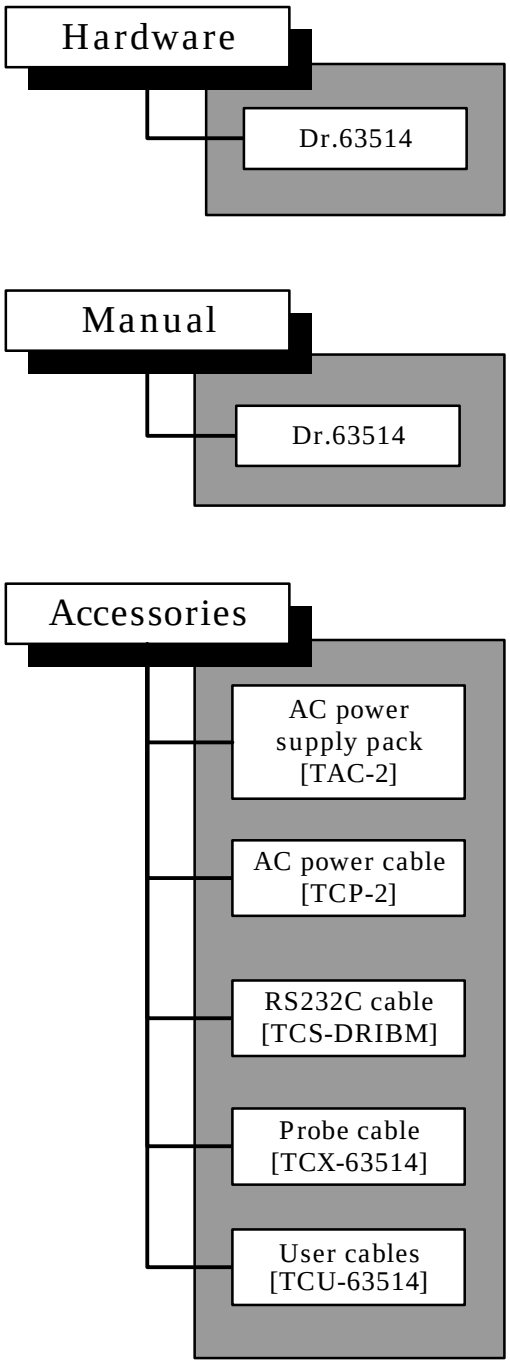
Chapter 1. Overview

1. Overview

The Dr.63514 in-circuit emulator supports the development of user application programs for the Oki ML63512/514 of CMOS 4-bit microcontroller.

2. Package Components

The package contains the components listed below.



This is the emulator unit.

This, the document that you are now reading, is the manual for the package.

Power supply for the Dr.63514 in-circuit emulator.

Cable that connects to the AC power supply pack.

This cable connects the host computer to the emulator.

This cable plugs into the probe cable connector on top of the emulator.

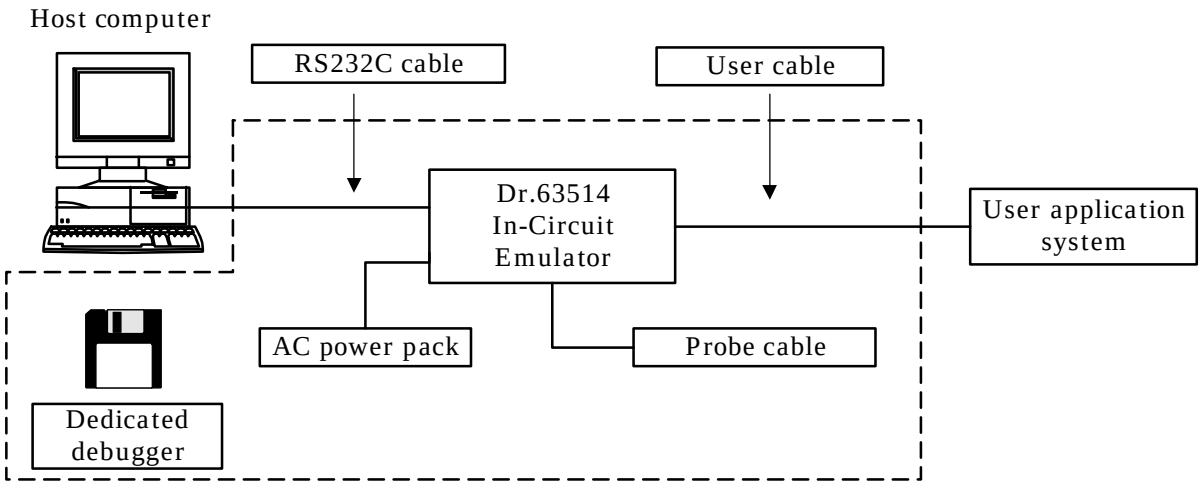
This cable connects the user cable connector to the user application system.

3. Configurations

The emulator is used in the two configurations shown below.

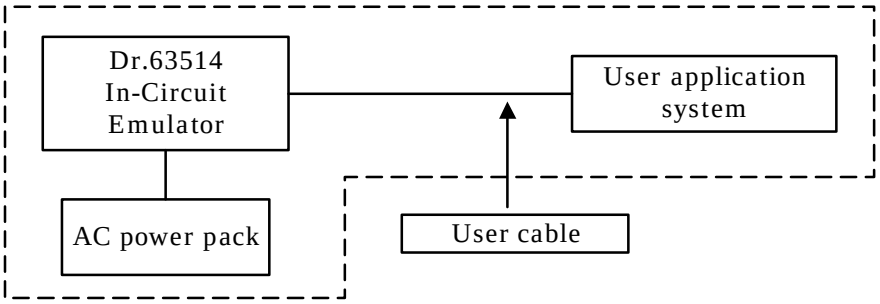
(1) Emulation

This configuration is for high-level debugging using a dedicated debugger running on a development host.



(2) Evaluation

This configuration is for stand-alone execution of the user application program from EPROMs.

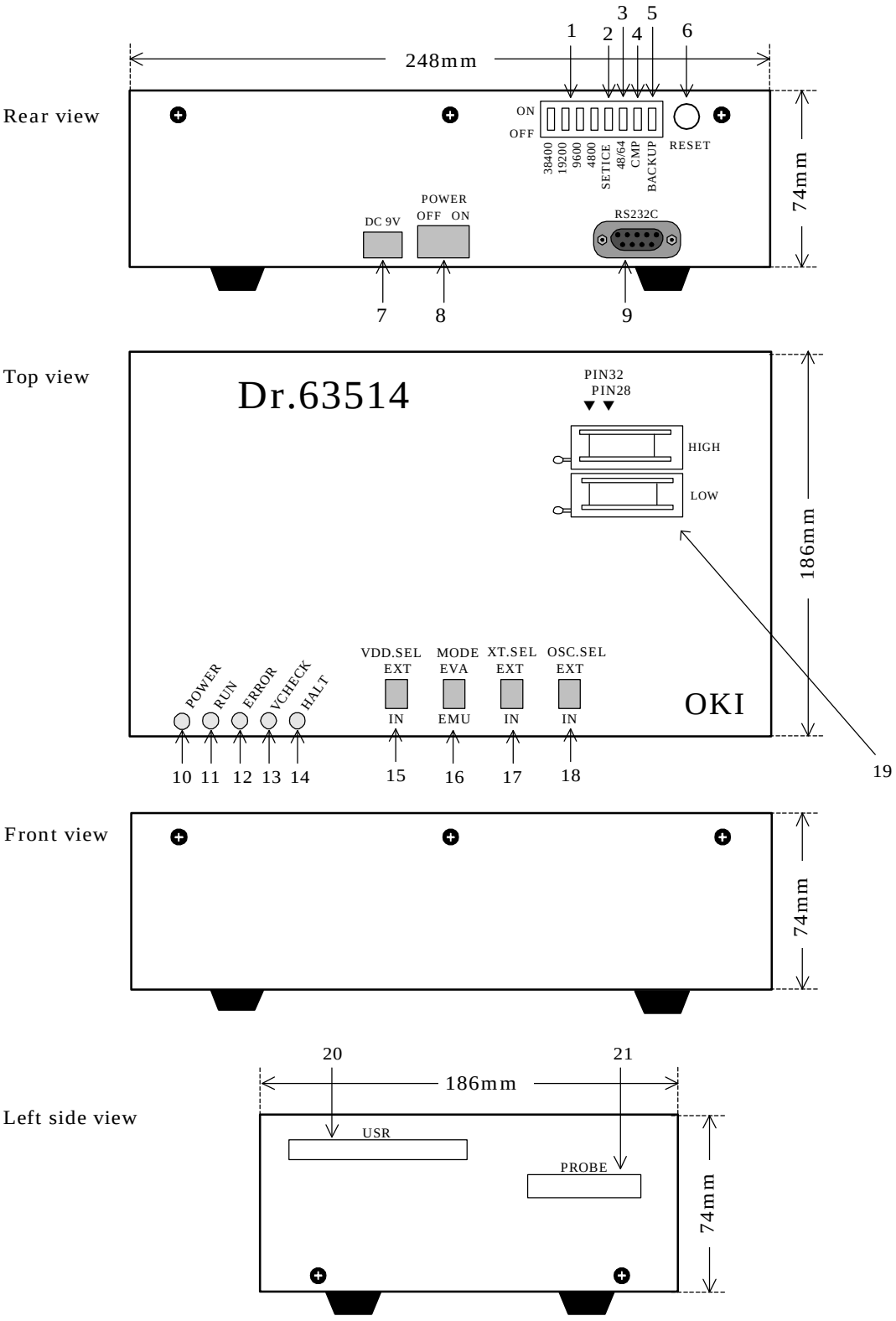


Caution

Use only the cables and other accessories provided.

Using non-compatible parts risks fire and breakdown.

4. Names of Parts



1. Baud rate switches (BAUD)

These switches specify the baud rate for the serial interface to the development host.

2. Device configuration switch (SETICE)

This switch is used when specifying the target microcontroller with the dedicated emulator setup utility.

3. Package selection switch (48/64)

Switch that enables or disables the ports (P6, P9, PA). The ability to use these ports as I/O pins depends on the package type.

4. Comparator setting switch (CMP)

Switch that enables or disables the comparator.

5. Backup setting switch (BACKUP)

Turns the backup function on or off.

6. Reset switch (RESET)

This switch resets the emulator and initializes its firmware.

7. DC power input jack (DC9V)

Jack for DC power input. Connect the DC power cable from the AC power pack provided as an accessory with this product.

8. Power switch (POWER)

This switch controls power to the emulator. Rapid switching on and off can prevent the main control CPU from resetting properly, leading to faulty emulator operation.

9. RS232C connector (RS232C)

This connects to the RS232C cable provided.

10. Power supply LED (POWER)

This red LED lights when power is being supplied to the emulator.

11. Execution LED (RUN)

This green LED lights during real-time emulation. It also may flash during emulator initialization.

12. Error LED (ERROR)

This red LED lights when a problem within the emulator prevents normal operation from proceeding. It also may flash during emulator initialization.

13. Voltage check LED (VCHECK)

This red LED lights when VDD, positive power supply voltage, falls below 0.7V.

14. Halt mode LED (HALT)

This orange LED lights when the evaluation chip is in halt mode.

15. Positive power supply voltage switch (VDD.SEL)

This switches VDD, the positive power supply voltage, between internal and external sources.

16. Evaluation/emulation switch (MODE)

This switches emulator operation between evaluation and emulation.

17. Low-speed clock switch (XT.SEL)

This switches the low-speed (XT) clock between internal and external sources.

18. High-speed OSC clock switch (OSC.SEL)

This switches the high-speed (OSC) clock between internal and external sources.

19 EPROM sockets (EPROM.HIGH and EPROM.LOW)

These accept EPROMs containing the user application program.

20. User cable connector (USR)

This connects to the user cable provided.

21. Probe connector (PROBE)

This connects to the probe cable provided.

Chapter 2. Functions

1. Emulator Specifications

Function	Specification
Interface	serial interface 4800/9600/19200/38400/51200/57600/76800/115200 bps, 8 bits, no parity, 1 stop bit, XON/XOFF flow control
Program size	**Code memory size:** up to 64 KW, depending on the microcontroller **Memory backup:** approx. 1 day
Data storage	Depending on the microcontroller
Emulation	Real-time emulation (evaluation and emulation configurations) Single-step emulation (emulation configuration only)
Breaks	**Breaks with parameters:** Address break Address pass count break RAM data match break RAM address match break Internal ROM table data match break Internal ROM table address match break **Breaks on specific conditions:** Breakpoint break Trace memory full break Cycle counter overflow break External break HALT break Call stack overflow break Register stack overflow break **Forced breaks:** N area access break User break
Real-time tracing	**Trace memory size:** 8192 entries **Trace conditions:** Free-running trace Trigger trace **Trace data:** PC, A, FLAG, CBR, EBR, HL, XY, SP, RSP, MI, MD, XP, RAM address, RAM data
Cycle counter	**Counter:** One 24-bit counter **Count conditions:** Free-running count Trigger count

Function	Specification
Coverage functions	**Monitored space:** Program memory address space **Monitored condition:** Instruction fetch **Coverage information:** Address access information
Probe cable I/O	• Break (EXT.BRK) input • Synchronous (SYNC.OUT) output • Trace (PROBE0 to PROBE3) inputs
LEDs	POWER, RUN, ERROR, VCHECK, HALT
Voltage switching (User cable)	• Choice of internal (1.5 V) or external power with VDD.SEL switch • External power supply voltage range: 0.9 to 5.5 V
Clock switching	• Choice of internal or external low-speed clock with XT.SEL switch • Choice of internal or external high-speed clock with OSC.SEL switch
Mask option selection function	Switches between the internal and an external pull-up resistor for the reset pin.
Package selection function	Switches between 48-pin (ports 6, 9, and A disabled) and 64-pin (ports 6, 9, and A enabled) operation under control of the 48/64 switch.
User interface cables	Flat cables with 60 pins (pitch = 2.54 mm)
Comparator selection function	Switches between comparator enabled and comparator disabled operation under control of the CMP switch.
Backup selection function	Switches between backup on and backup off operation under control of the BACKUP switch.
Power supplies	**In:** 100 to 240 V AC, 50/60 Hz (9 V DC) **Input voltage:** AC adapter
Operating conditions	**Temperature:** 5 to 50°C **Humidity:** 30 to 80%
External dimensions and weight	**Dimensions:** 248 (W) 186 (D) 74 (H) mm **Weight:** 1.8 kg

■ **Note 1** ■

Although the ML63512 and ML63514 have four mask options (low-speed oscillator clock, high-speed oscillator clock, and reset), the Dr.63514 in-circuit emulator only supports the reset-related mask options.

■ **Note 2** ■

The ML63512 and ML63514 are available in two packages, a 48-pin TQFP and a 64-pin TQFP. These packages differ in whether or not the port 6, 9, and A pins can be used for I/O. The Dr.63514 in-circuit emulator supports that difference with a "package selection switch".

■ **Note 3** ■

The Dr.63514 in-circuit emulator does not support a user interface for the package type.

■ **Note 4** ■

The circuits that correspond to the ML63512/514 for the Dr.63514 in-circuit emulator consist of the nX-4/250 core evaluation chip, which corresponds to the ML63512/514 CPU core block, and a block that corresponds to the ML63512/514 I/O block. Note that since the I/O block is formed from normal discrete components, the electrical characteristics of the ports and other circuits differ somewhat from those of the ML63512/514. See chapter 4, "Additional Usage Notes", for details on these differences.

2. Functions

2.1 Configuring for Target Device

The emulator is used to develop user application program for the Oki ML63512/514 of 4-bit microcontroller even though the individual devices have different ROM sizes and onboard peripherals.

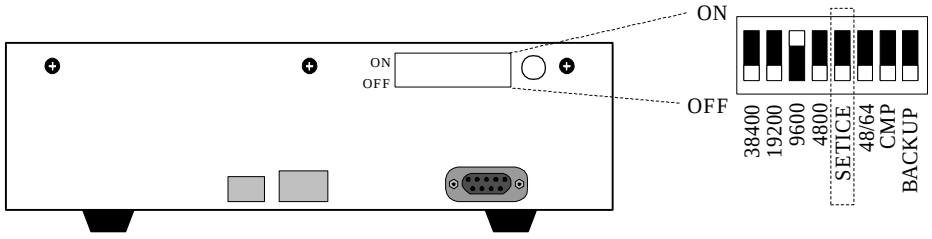
A dedicated emulator setup utility running on the development host configures the emulator to cover these differences by downloading the contents of the device information file (.TCD file) for the target microcontroller to an EEPROM inside the emulator. These settings take effect the next time that the power is applied or the reset switch is pressed.

<The settings in this file>

- Specify the highest code memory address in the target microcontroller's ROM and thus the number of breakpoint bits, trace enable bits, instruction executed (IE) bits, and sync out bits
- Enable and disable RAM and SFR addresses
- Specify the sizes of the call and register stacks

Changing these settings requires setting the emulator's SETICE switch to its ON position. Always set this switch to its OFF position for debugging.

Operation	SETICE switch
Device configuration	ON
Debugging	OFF



■ **Reference** ■

Refer to the setup utility’s manual for the detailed operating procedure.

2.2 Evaluation Operation

2.2.1 Overview

The emulation configuration is for high-level debugging using a dedicated debugger running on a development host; the evaluation configuration, for stand-alone execution of the user application program from EPROMs.

The MODE switch switches between the two, taking effect the next time that the power is applied or the reset switch is pressed.

MODE switch	Configuration
EMU	Emulation
EVA	Evaluation

2.2.2 Operation

The evaluation configuration produces real-time emulation of the user application program from EPROMs. The two EPROM sockets, labeled EPROM.HIGH and EPROM.LOW, accept the following types.

- MSM27512 and compatibles
- MSM27101 and compatibles

The emulator cannot program EPROMs. Use a commercial EPROM writer to write the two object files generated by the dedicated assembler to separate EPROMs.

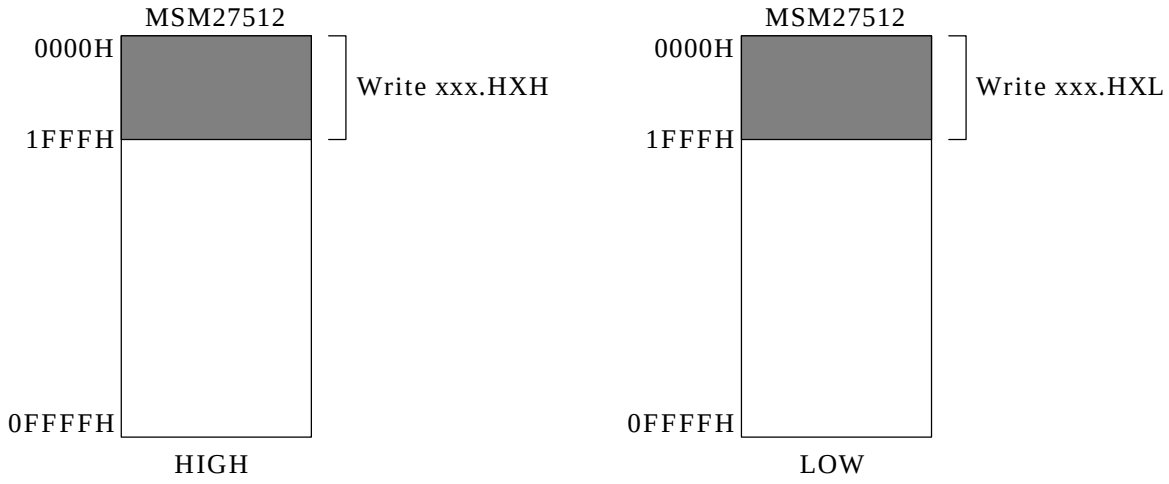


Figure 2-1 Address Ranges for MSM27512

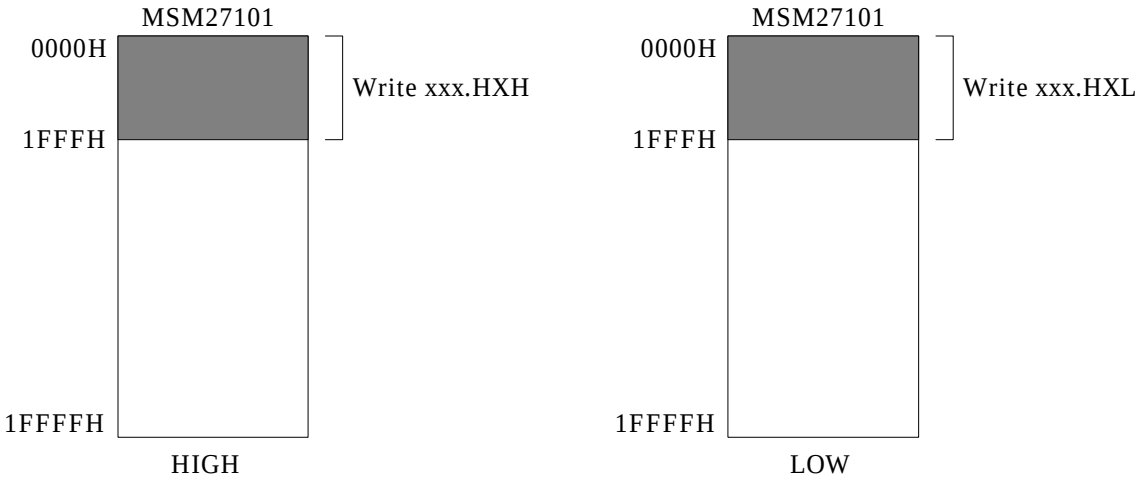


Figure 2-2 Address Ranges for MSM27101

Figures 2-1 and 2-2 show the ML63514 address ranges. The extension .HXH indicates the upper 8-bit object file from the assembler; .HXL, lower 8-bit object file.

In the stand-alone evaluation configuration, turning on the emulator or pressing the reset switch resets the evaluation chip and starts real-time emulation from address 0. The RUN LED then lights.

■ **Note 1** ■

The ML63512/514 test data area is handled as an area in which program execution is not possible (an N area) in the Dr.63514 in-circuit emulator.

■ **Note 2** ■

After starting a real-time emulation, if an area in which program execution is not possible (an N area) (including the test data area) is accessed, the RUN LED will be turned off and real-time emulation will be forcibly terminated. If real-time emulation is forcibly terminated, an input to the RESETB pin will not restart real-time emulation. To restart real-time emulation, press the reset switch on the Dr.63514 in-circuit emulator itself.

■ **Note 3** ■

The RESETB pin is only enabled during periods when the RUN LED is lit. In particular, reset signal input to the RESETB pin from the user application system is disabled while the Dr.63514 in-circuit emulator is initializing and after real-time emulation has been forcibly terminated.

2.3 Emulation Operation

Emulation involves running a user application program, under the control of software on a development host, in real time at the same speed and with electrical characteristics approaching those of the volume production masked ROM version. These last two characteristics distinguish it from simulation, the replacement of hardware with software running on a development host.

Two types of emulation are available: real-time and single-step. The former runs nonstop up until a break. The latter pauses after each instruction to permit such debugging operations as examining and modifying register contents.

Control from the host is possible because the evaluation chip inside the emulator has no masked ROM. Instead there are data and address buses to RAM and other external devices plus related control circuits. These modifications permit the microcontroller to execute the user application program in real time while still allowing the emulator (as thus the control software) debugging access to the device's memory, registers, and flags. The microcontroller uses this additional hardware to read instructions; the emulator, to control user application program execution and access these internal device components. The pins that the evaluation chip shares with the volume production masked ROM version are connected to the user application system through the user cables.

■ **Note 1** ■

The circuits that correspond to the ML63512/514 for the Dr.63514 in-circuit emulator consist of the nX-4/250 core evaluation chip, which corresponds to the ML63512/514 CPU core block, and a block that corresponds to the ML63512/514 I/O block. Note that since the I/O block is formed from normal discrete components, the electrical characteristics of the ports and other circuits differ somewhat from those of the ML63512/514. See chapter 4, "Additional Usage Notes", for details on these differences.

2.3.1 Single-Step Emulation

Single-step emulation pauses after each instruction to permit such debugging operations as examining and modifying register contents.

<During single-step emulation>

- Emulation aborts if the program counter (PC) strays into the nonexistent code memory area (N area).
- HALT is just another instruction. It produces a temporary transition to halt mode followed by an immediate return.
- Real-time tracing and the cycle counter are disabled.

- Instruction executed (IE) bit updates and sync out output continue.
- User cable reset (RESETB) input from user cable does nothing.
- The pauses after each instruction prevent operation of serial ports and other time-sensitive portions.

■ **Note 1** ■

Do not press the emulator's reset switch during single-step emulation. Doing so invalidates code memory contents.

2.3.2 Real-time Emulation

Real-time emulation runs nonstop or until there is a break from the following list.

<Breaks with parameters>

- Address break
- Address pass count break
- RAM data match break
- RAM address match break
- Internal ROM table data match break
- Internal ROM table address match break

<Breaks on specific conditions>

- Breakpoint break
- Trace memory full break
- Cycle counter overflow break
- External break
- HALT break
- Call stack overflow break
- Register stack overflow break

<Forced breaks>

- N area access break
- User break

■ **Note 1** ■

Evaluation, in contrast, only offers N area access breaks.

These break conditions generate a break request. Acceptance terminates the real-time emulation.

■ **Note 2** ■

The second group uses parameters that the user must set in advance.

Figure 2-3 shows the interaction between these break conditions and the break condition register.

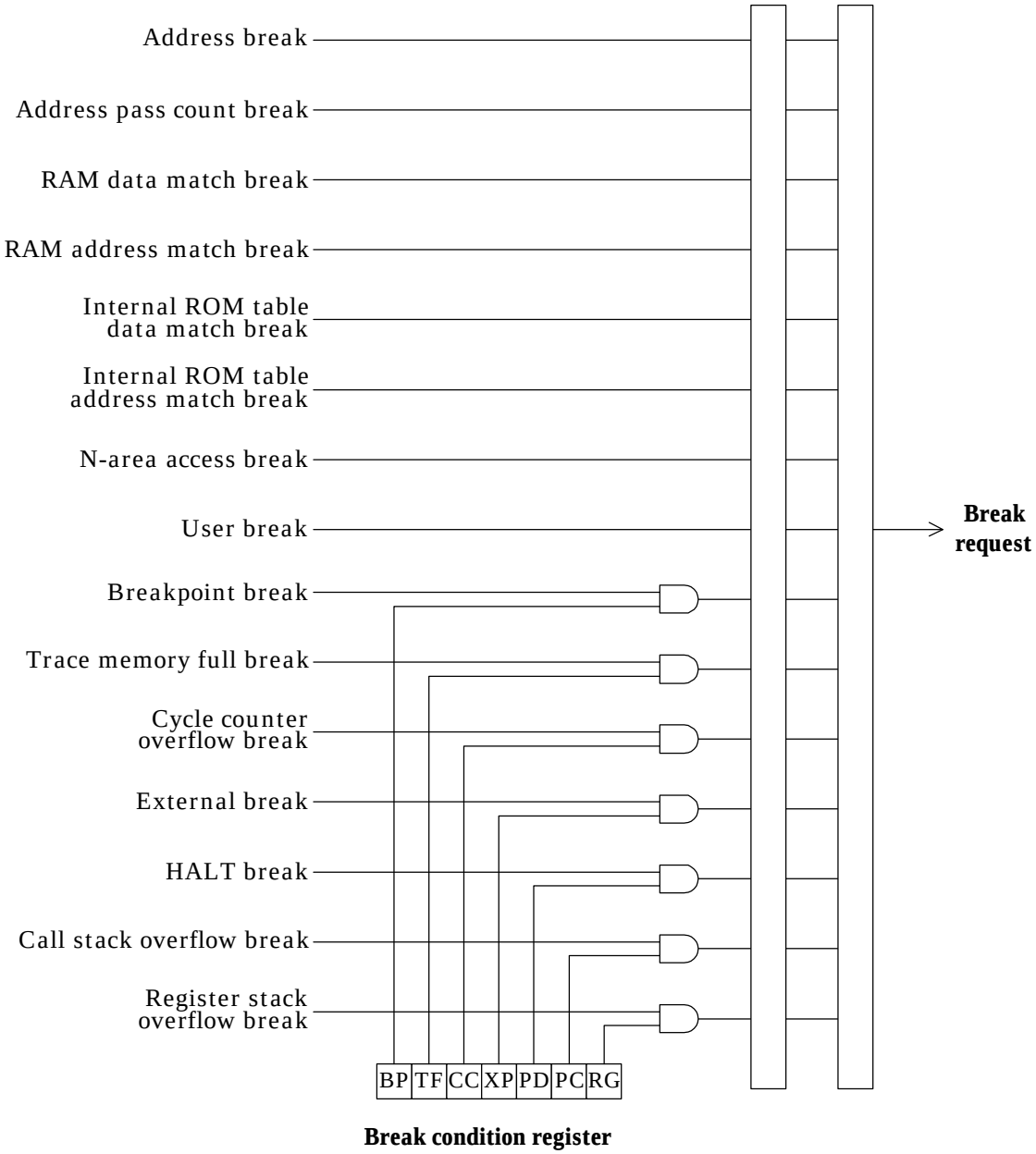
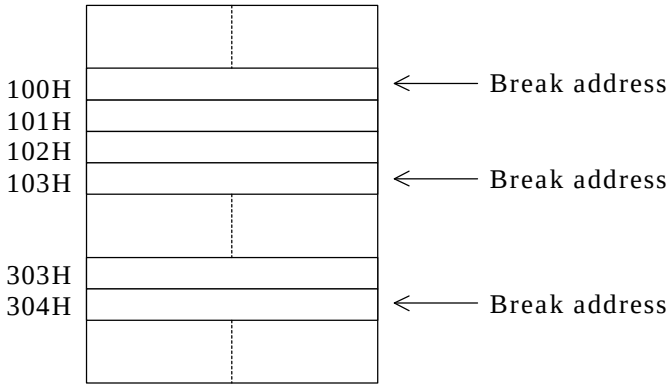


Figure 2-3 From Break Condition to Break Request

2.3.2.1 Breaks with Parameters

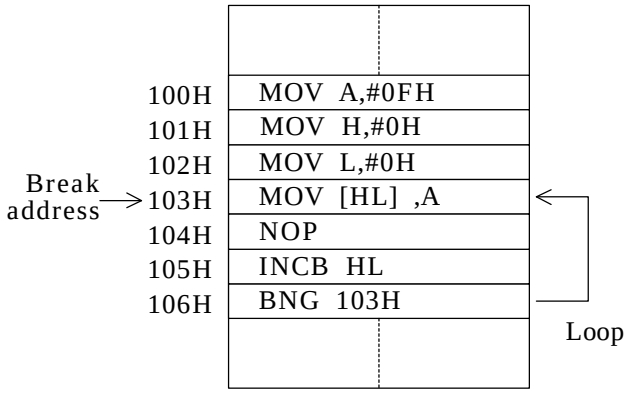
(1) Address break

Execution breaks after the instruction at the specified break address has executed.



(2) Address pass count break

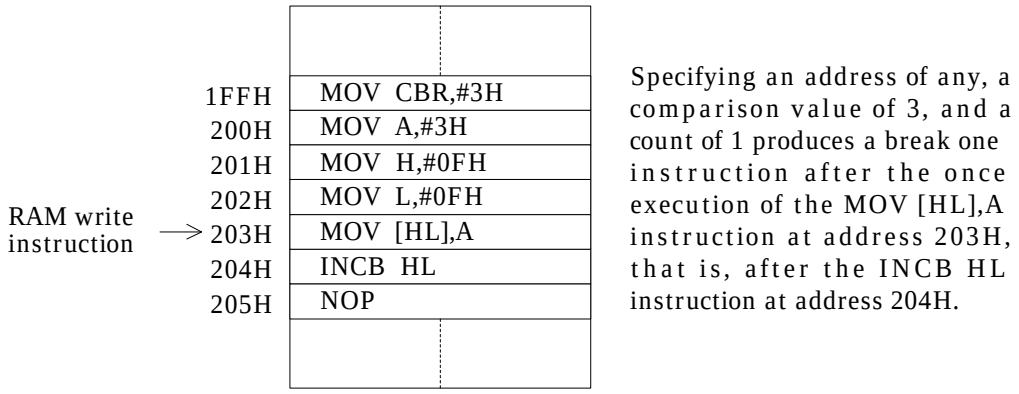
Execution breaks after the instruction at the specified address has executed the specified number of times.



Specifying a break address of 103H and a count of 5 produces a break after the fifth execution of the MOV [HL],A instruction at address 103H.

(3) RAM data match break

Execution breaks one instruction after instructions have written the specified data the specified number of times to either any data memory address or the specified data memory address.



The break timing is not immediately after the instruction satisfying the break condition, but an additional instruction later.

There is a bit mask parameter for extending data checking to multiple (or even all) comparison values.

Specifying a code memory address produces a break only if the instruction at that address writes to a data memory address.

■ **Note 1** ■

RAM data match breaks are available over the entire data memory address space—even SFR addresses with reserved bits (bits that ignore writes and always return “1”) and addresses with read-only bits.

■ **Note 2** ■

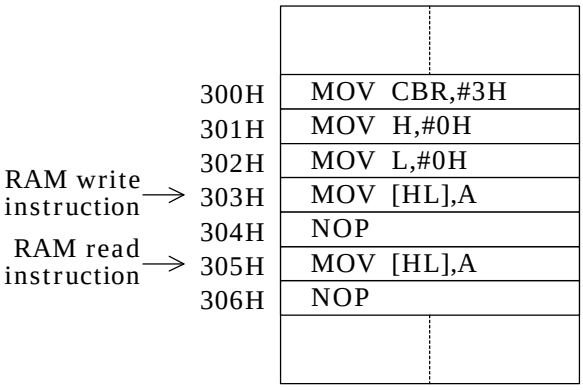
RAM data match breaks check only writes by instructions. RAM modifications by timers and other circuits are ignored.

■ **Note 3** ■

A RAM data match break request remains in effect until the next write instruction. To resume emulation under the same break conditions, write somewhere in RAM using data that does not produce another break.

(4) RAM address match break

Execution breaks one instruction after instructions have written the specified number of times to the specified data memory address.



Specifying an address of 300H and a count of 2 produces a break one instruction after the MOV [HL],A instructions at addresses 303H and 305H, that is, after the NOP instruction at address 306H.

The break timing is not immediately after the instruction satisfying the break condition, but an additional instruction later.

There is a bit mask parameter for extending address checking to multiple data memory addresses.

■ **Note 4** ■

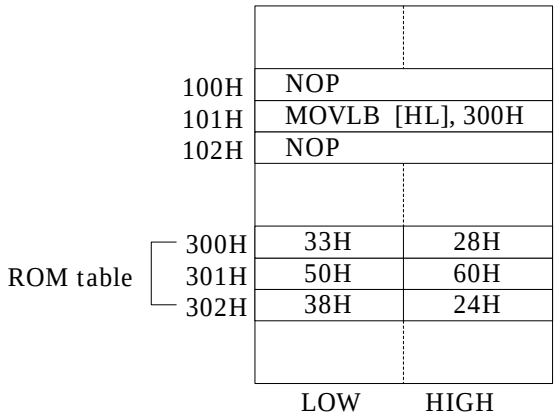
RAM address match breaks are available over the entire data memory address space—even SFR addresses.

(5) Internal ROM table data match break

Execution breaks when a ROM table reference instruction (MOVHB or MOVLB) reads either any data or the specified data from the specified code memory address.

There is a bit mask parameter for extending data checking to multiple (or even all) comparison values.

This type of break provides no count parameter.



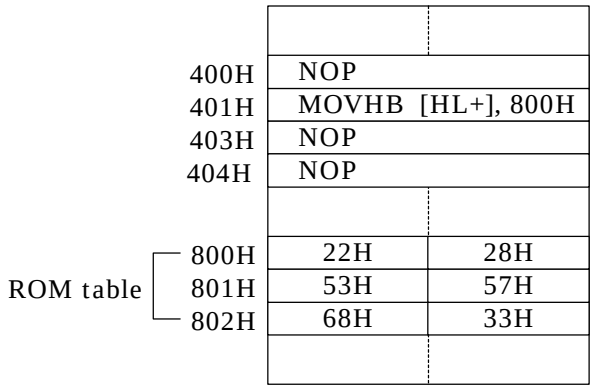
Specifying a ROM table address of 300H and a comparison value of 33H produces a break one instruction after the MOVLB [HL],300H instruction at address 101H, that is, after the NOP instruction at address 102H.

The break timing is not immediately after the instruction satisfying the break condition, but an additional instruction later.

Specifying a code memory address produces a break only if the instruction at that address is a ROM table reference instruction.

(6) Internal ROM table address match break

Execution breaks when a ROM table reference instruction (MOVHB or MOVLB) reads from the specified code memory address.



Specifying a ROM table address of 800H produces a break one instruction after the MOVHB [HL+],800H instruction at address 401H, that is, after the NOP instruction at address 403H.

The break timing is not immediately after the instruction satisfying the break condition, but an additional instruction later.

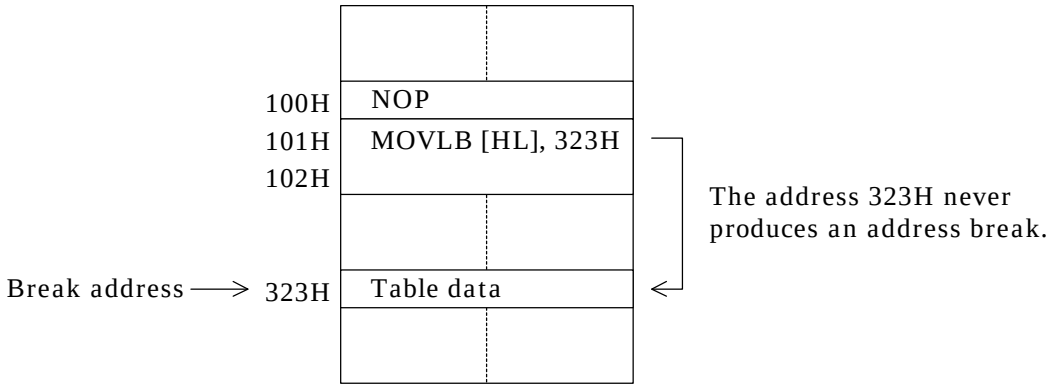
There is a bit mask parameter for extending address checking to multiple ROM table addresses.

This type of break provides no count parameter.

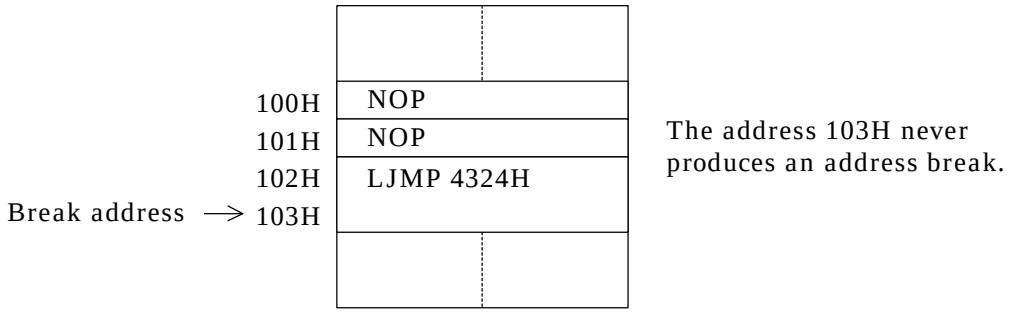
■ **Note 5** ■

Address settings at the following addresses never produce address breaks or address pass count breaks—unless there is something seriously wrong with the user application program.

(a) ROM table locations



(b) The second word of a 2-word instruction



■ **Note 6** ■

Some break types support bit masks for extending data or address matches.

(a) Data match:

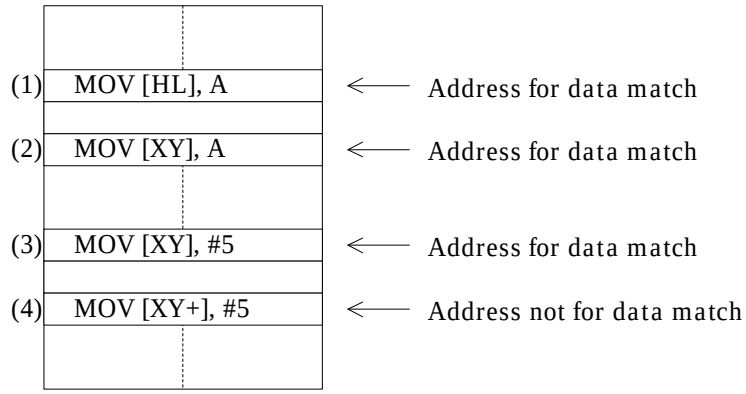
Specifying a data memory address of 200H, a comparison value of 4H, and a mask of 0111B produces a match whenever 4H, 0CH is written to data address 200H.

(b) Address match:

Specifying a RAM address of 120H and a mask of 0FFF0H produces a match for all addresses from 120H to 12FH (among others).

■ **Note 7** ■

Specifying a code memory address produces data matching for the instruction at that address. In the following example, only the first three specifications produce data matches.

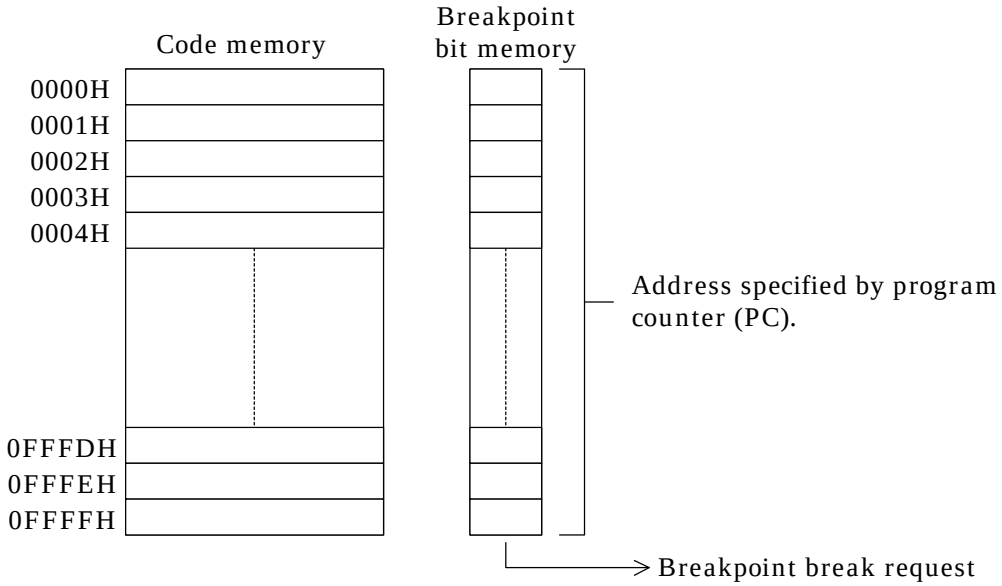


2.3.2.2 Breaks on Specific Conditions

These breaks are the result of specific conditions involving flag bits and counters.

(1) Breakpoint break

For each code memory address, the emulator provides a breakpoint bit for enabling these breaks. Setting a breakpoint at a code memory address sets the corresponding breakpoint bit to “1.”

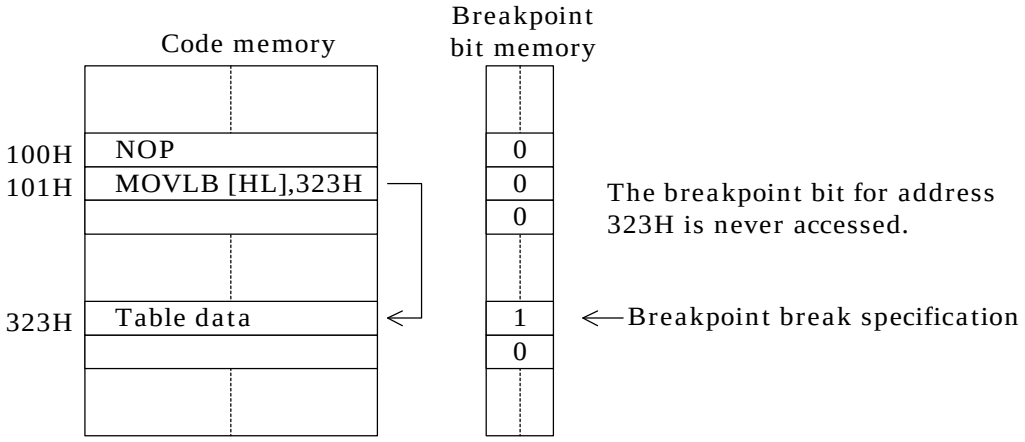


If these breaks are enabled, execution of the instruction at that address produces a break request of this type.

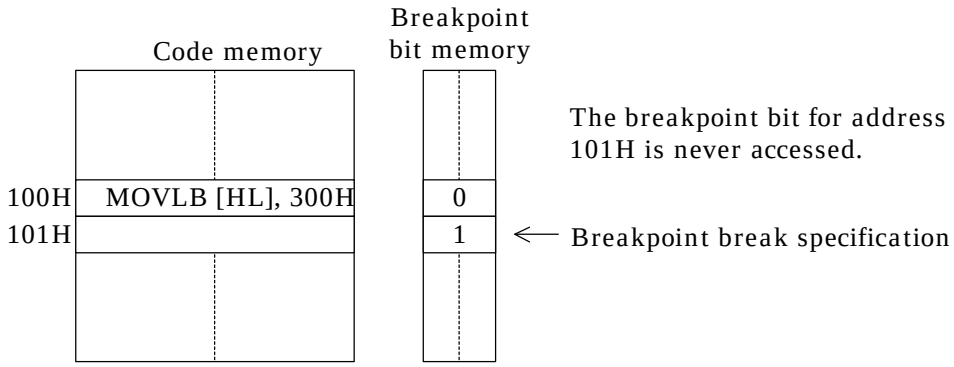
■ **Note 1** ■

There are no limits on the number of breakpoints or their locations in the code memory space. Breakpoint settings at the following addresses, however, never produce breaks.

(a) ROM table locations



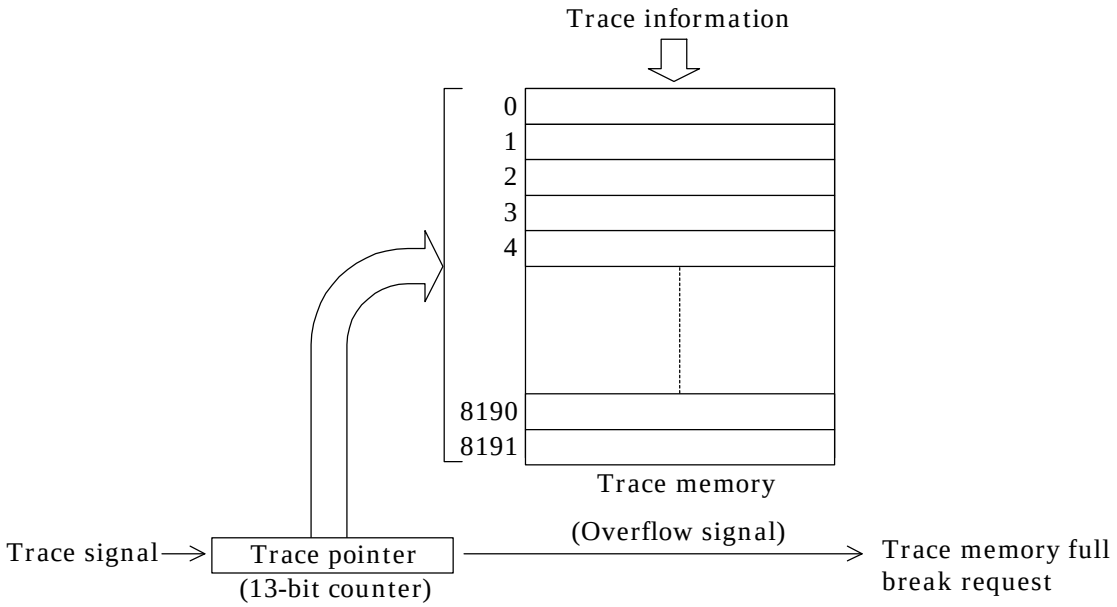
(b) The second word of a 2-word instruction



(2) Trace memory full break

If these breaks are enabled, overflow during real-time emulation of the trace pointer, a 13-bit counter giving the location of the next entry to be written within its 8192-entry trace table produces a break request of this type.

The emulator has room for 8192 trace entries.



If trace memory full breaks are enabled, overflow during real-time emulation of the trace pointer, a 13-bit counter, produces a break request of this type.

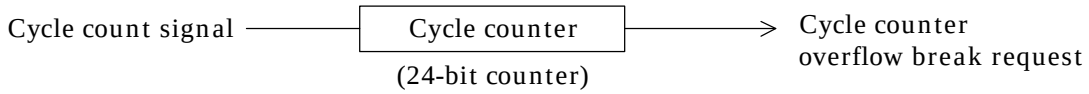
■ **Note 2** ■

Resuming real-time emulation after this break automatically cancels this request. The next

break of this type is not until the next overflow.

(3) Cycle counter overflow break

If these breaks are enabled, overflow during real-time emulation of the cycle counter, a 24-bit counter summing the machine cycles for instructions executed produces a break request of this type.



■ **Note 3** ■

Resuming real-time emulation after this break automatically cancels this request. The next break of this type is not until the next overflow.

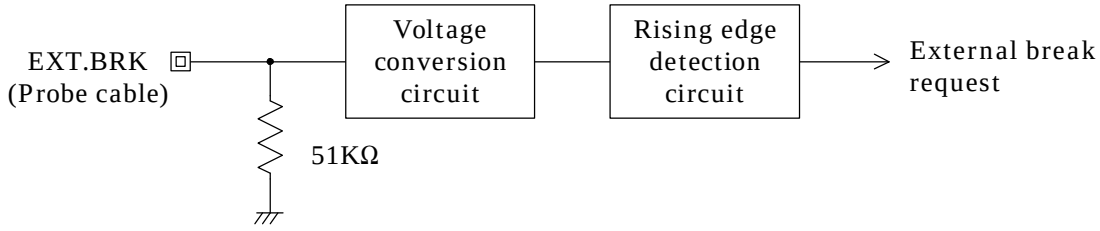
■ **Note 4** ■

The cycle counter value after the break varies between one and the execution time of the instruction producing the overflow. It is always 1 for a single-cycle instruction, but could be 1, 2, or 3 for a 3-cycle one.

(4) External break

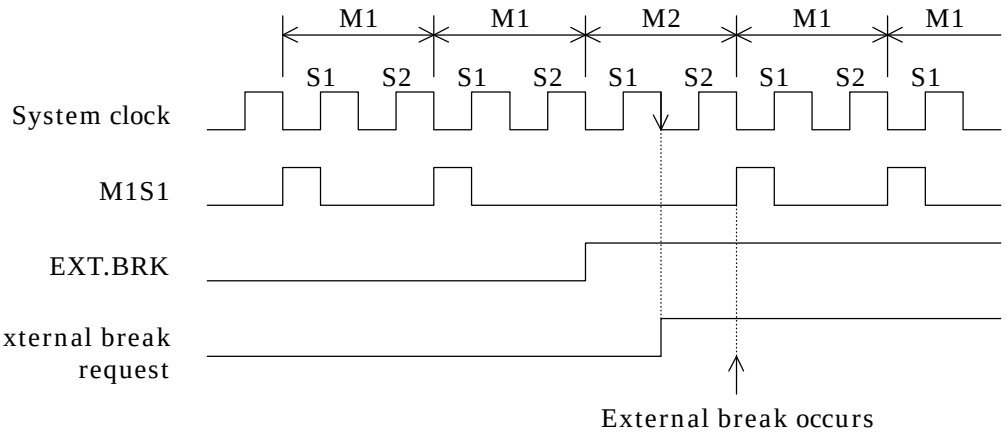
If these breaks are enabled, a rising edge in the probe cable break (EXT.BRK) input during real-time emulation produces a break request of this type.

The signal uses VDD, the positive power supply voltage, for its “H” level.



■ **Note 5** ■

The external break request and acceptance coincide with the beginning and end of an instruction S2 cycle.



(5) Power down (HALT) break

If these breaks are enabled, a HALT instruction produces a break request of this type.

■ **Note 6** ■

The HALT instruction produces a temporary transition to halt mode followed by an immediate return. Restarting real-time emulation without specifying a starting address causes execution to resume from the instruction after the HALT instruction.

(6) Call stack overflow break

If these breaks are enabled, stack over- or underflow in the call stack pointer (SP) as the result of pushing onto or popping from that stack during real-time emulation produces a break request of this type.

■ **Note 7** ■

When a call stack push/pop is performed, if the emulator detects a stack pointer overflow or underflow, then it will output a call stack overflow break request.

■ **Reference** ■

The stack size appears in the user's manual for the target microcontroller.

(7) Register stack overflow break

If these breaks are enabled, stack over- or underflow in the register stack pointer (RSP) as the result of pushing onto or popping from that stack during real-time emulation produces a break request of this type.

■ **Note 8** ■

When a register stack push/pop is performed, if the emulator detects a stack pointer overflow or underflow, then it will output a register stack overflow break request.

■ **Reference** ■

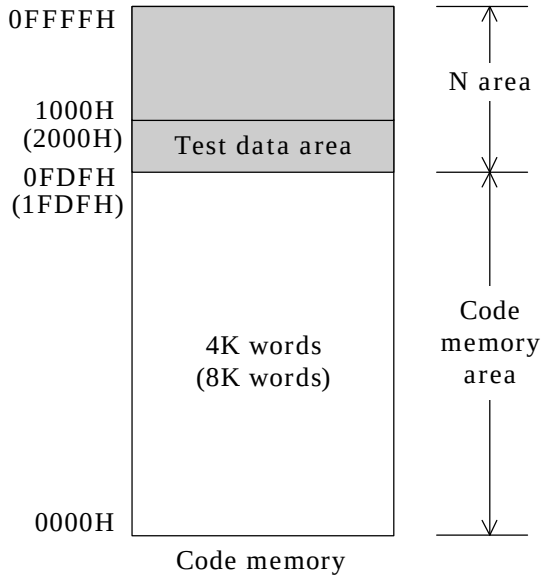
The stack size appears in the user's manual for the target microcontroller.

2.3.2.3 Forced Breaks

Forced breaks are breaks that do not depend on parameters or specific conditions. They force immediate termination of real-time emulation.

(1) N area access break

An attempt to read an instruction or ROM table data from a code memory address not physically present produces a break request of this type.



■ **Note 1** ■

The former is especially problematical because the break occurs after the microcontroller attempts to execute the indeterminate data from the invalid address.

Real-time emulation immediately terminates.

■ **Note 2** ■

The emulator considers the microcontroller's test data area part of the N area.

■ **Note 3** ■

When code memory has been expanded to 64K words using the code memory expansion function, only the test data area will be an N area.

(2) User break

User break input from the keyboard produces a break request of this type.

Real-time emulation immediately terminates.

■ **Note 4** ■

This break also terminates halt mode if it is in effect. Restarting real-time emulation without specifying a starting address causes execution to resume from the instruction after the HALT instruction.

2.4 Code Memory Operations

Code memory is a 16-bit address space that corresponds to the masked ROM of the volume production device. The emulator starts with a code memory area the size of the ROM in the microcontroller specified with the dedicated emulator setup utility.

2.4.1 Data Operations between Code Memory and Disk Files

These operations include copying data in either direction between code memory and disk files.

They always involve simultaneous use of a pair of object files: one for the upper 8 bits and another for the lower 8 bits.

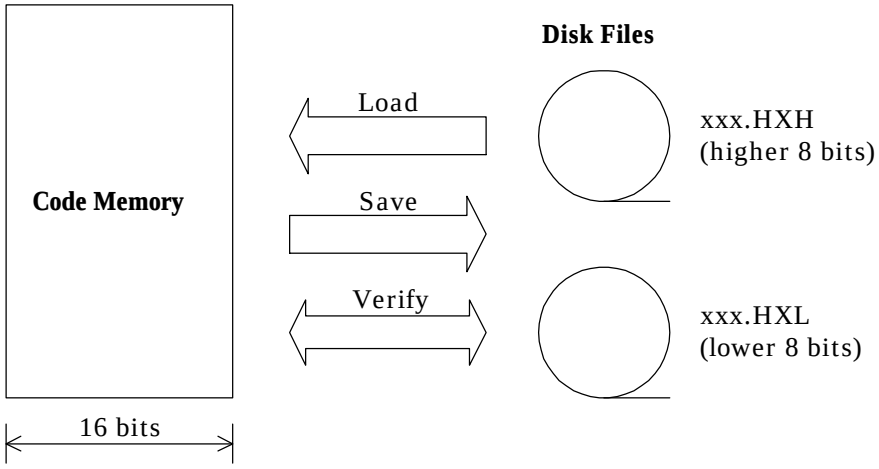


Figure 2-4 Data Operations between Code Memory and Disk Files

2.4.2 Data Operations between Code Memory and EPROMs

These operations limit copying data to one direction: from EPROMs in the two sockets on top of the emulator to code memory.

The socket labeled EPROM.HIGH is for the upper 8-bit object file (.HXH) from the assembler; EPROM.LOW, the lower 8-bit object file (.HXL).

The emulator cannot program EPROMs. Use a commercial EPROM writer to write the two object files generated by the dedicated assembler to separate EPROMs.

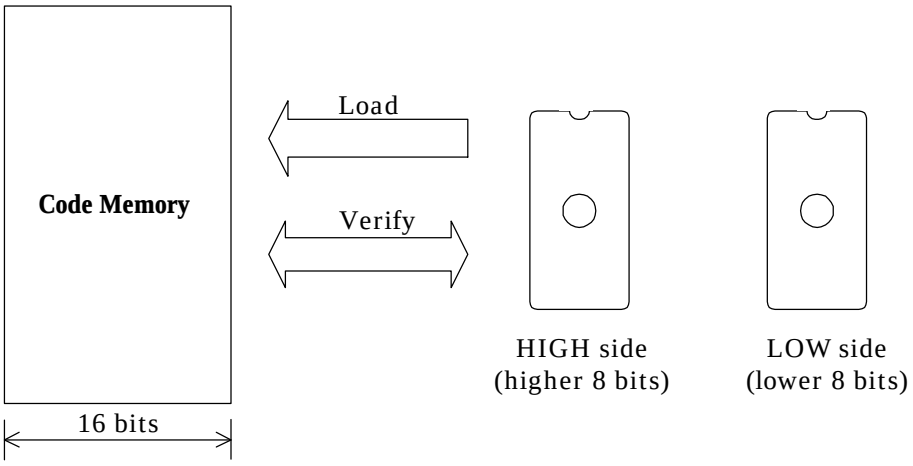


Figure 2-5 Data Operations between Code Memory and EPROMs

■ **Note 1** ■

Stand-alone, evaluation operation does not use code memory. The user application program executes directly from the EPROMs in the EPROM sockets.

2.4.3 Displaying/Changing/Moving Code Memory

Displaying/changing code memory can be performed at the instruction code level or instruction mnemonic level. Moving code memory can be performed at the instruction code level.

2.4.4 Code Memory Backup

A large capacitor inside the emulator maintains code memory contents for up to 24 hours (one day).

■ **Note 1** ■

The backup interval depends on how long the capacitor has been charged—that is, how long the emulator power has been on—and varies with ambient conditions.

2.4.5 Expanding Code Memory

The EXPAND command temporarily expands the code memory to the full address space (64 KW) for debugging a user application program that is too large for the user application program memory.

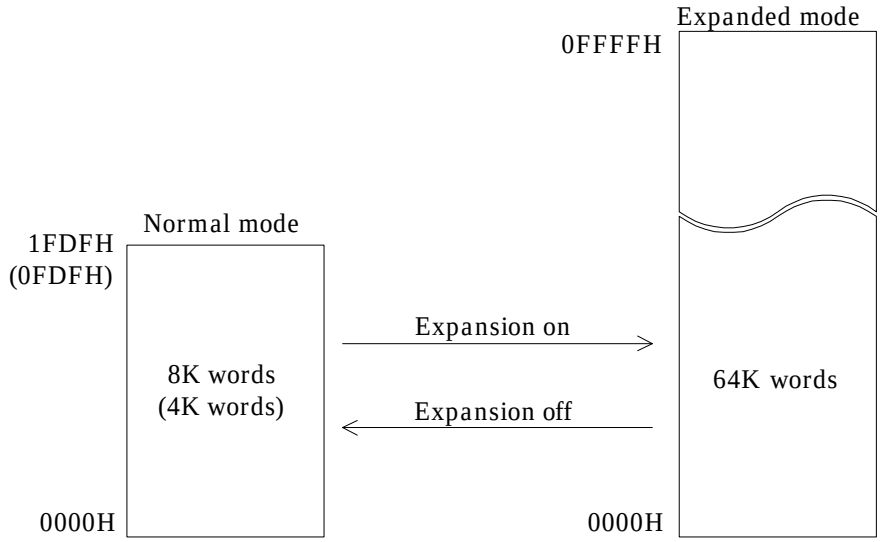


Figure 2-6 Code Memory Expansion

■ **Note 1** ■

Be sure to disable code memory expansion before starting final verification of user application program operation.

■ **Note 2** ■

Changing the code memory expansion setting resets the evaluation chip.

■ **Note 3** ■

Test data areas exist at locations 0FE0H to 0FFFH or 1FE0H to 1FFFH, regardless of whether the system is in normal mode or expanded mode. If expanded mode is used, the user must design and code applications so that they do not execute locations in the test data area. A forcible N area break will occur if a test data area location is executed.

2.5 Real-time Tracing

Real-time tracing stores the current instruction address, the contents of ACC and other registers, flag states, etc. for post mortem analysis of real-time emulation.

The emulator uses the 13-bit trace pointer to keep track of the location of the next entry to be written within its 8192-entry trace table. When the trace pointer reaches that number, it recycles, overwriting the oldest entry first.

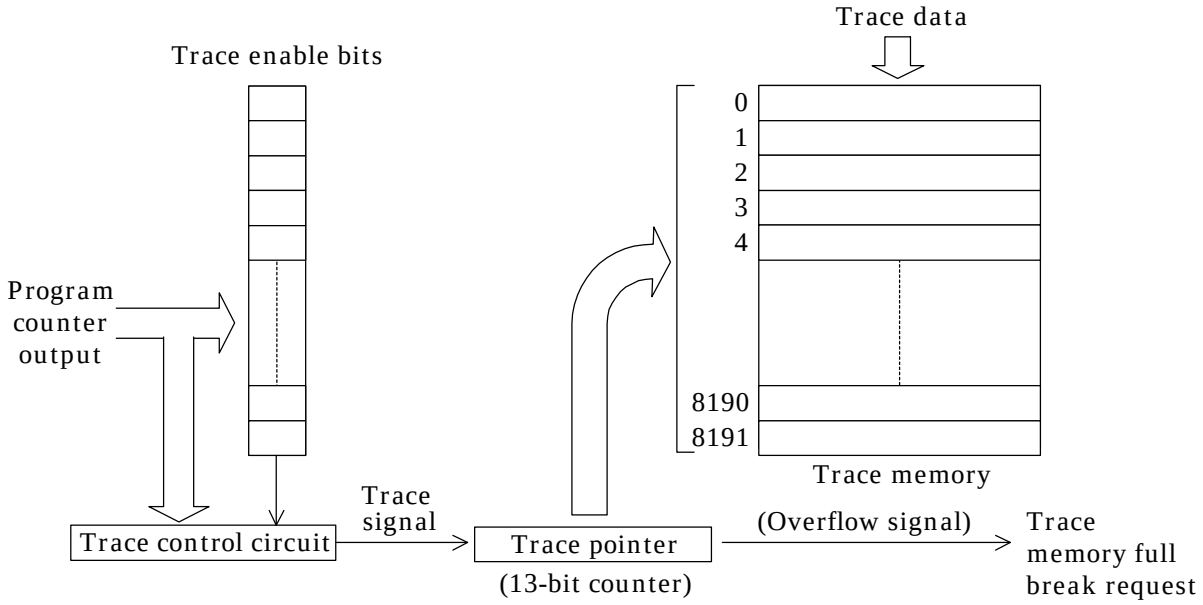


Figure 2-7 Real-Time Tracing

For each code memory address, the emulator provides a trace enable bit for use in limiting tracing to specific addresses.

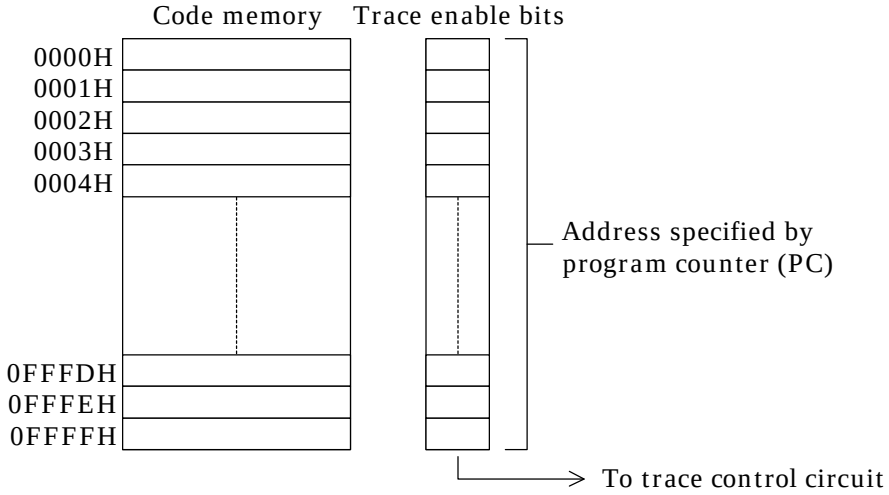


Figure 2-8 Trace Enable Bits

2.5.1 Trace Entries

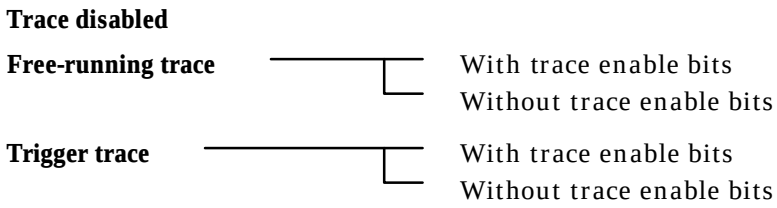
Trace entries track the following data items.

Label	Description	Bits
ADRS	Execution address	16
A	Accumulator	4
RADR	RAM address	12
RD	RAM data	4
CGZ	Flag register	3
MI	Master interrupt enable flag	1
MD	Melody request flag	1
CB	Current bank register	4
EB	Extra bank register	4
HL	HL register	8
XY	XY register	8
SP	Stack pointer	5
RS	Register stack pointer	4
XP	External probe data	4

In addition to the above, the emulator also traces interrupt requests for post mortem analysis of interrupts during real-time emulation.

2.5.2 Real-time Trace Control

Real-time tracing offers three operating modes and an option for limiting tracing to code memory addresses with “1” in their trace enable bits.



(1) Trace disabled

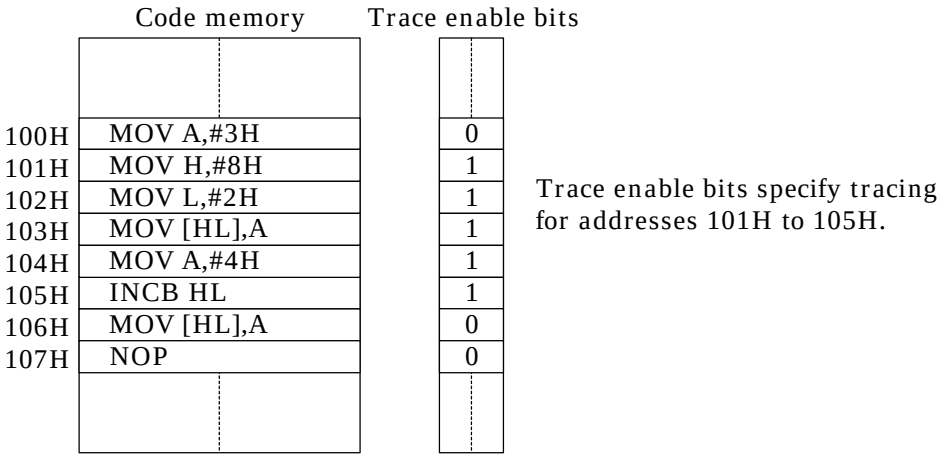
No instructions are traced.

(2) Free-running trace

Instructions are traced for all code memory addresses—unless limited with the trace enable bit option.

a. With trace enable bits

Tracing depends on the trace enable bit contents. Only addresses with a trace enable bit of 1 are traced.



b. Without trace enable bits

Tracing ignores the contents of the trace enable bits.

(3) Trigger trace

All instructions between start and stop triggers based on code memory addresses are traced—unless limited with the trace enable bit option.

There are three possible trigger combinations:

a. Specifying both a start and stop address

Tracing starts when real-time emulation visits the former and stops when the program counter (PC) hits the latter.

b. Specifying a start address only

Tracing starts when real-time emulation visits the former and continues until there is a break.

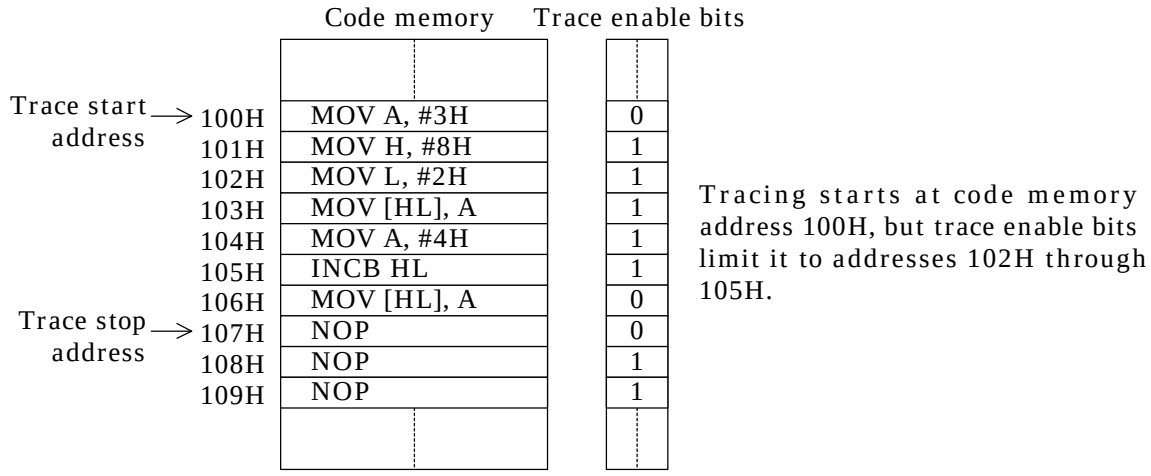
c. Specifying a stop address only

Tracing starts simultaneously with real-time emulation and stops when the program counter (PC) hits the latter.

This example shows such tracing together with the trace enable bit option.

a. With trace enable bits

Tracing depends on the trace enable bit contents. Only addresses with a trace enable bit of 1 are traced.



b. Without trace enable bits

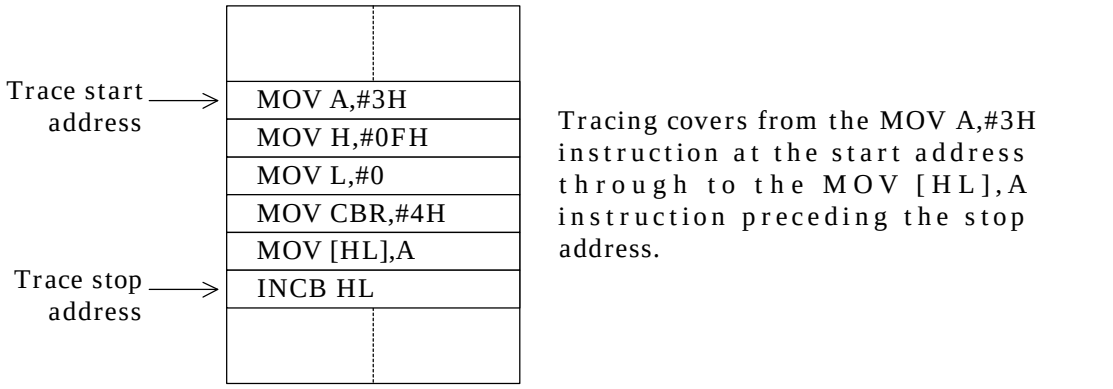
Tracing ignores the contents of the trace enable bits.

■ **Note 1** ■

Real-time tracing is enabled during real-time emulation and disabled during single-step emulation.

■ **Note 2** ■

The trigger fires just before execution of the instruction at the corresponding address, so the instruction at the start address is traced, but not the one at the stop address.



■ **Note 3** ■

Only start and stop address specifications corresponding to the first word of an instruction produce results. One specifying the second word of a two-word instruction or an entry in the ROM table yield a trigger that never fires.

■ **Note 4** ■

Only the trace enable bits corresponding to the first word of an instruction produce results. One specifying the second word of a two-word instruction or an entry in the ROM table is ignored. Disabling the option produces tracing regardless of the enable bit contents.

■ **Note 5** ■

The RAM address and RAM data fields contain indeterminate data until an instruction writes to a data memory address. They also repeat between such instructions.

■ **Note 6** ■

Flag (C, Z, G, MIEF) changes appear in the entry for the instruction preceding the one where the flag actually changes.

■ **Note 7** ■

The ROM table reference instruction (MOVHB, and MOVLB) consume two trace entries each.

2.5.3 Displaying/Searching Trace Entries

The emulator can display the real-time trace entries as a group or individually.

2.6 Profiling

The emulator supports two types of profiling:

- Checking code memory addresses accessed with instruction executed (IE) bits
- Measuring execution times with the cycle counter

2.6.1 Instruction Executed (IE) Bits

For each code memory address, the emulator provides an instruction executed (IE) bit for use in tracking instruction access during execution.

Each access to a code memory address during real-time emulation sets the corresponding IE bit to “1.” Examining these bits then reveals which instructions were executed during the emulation.

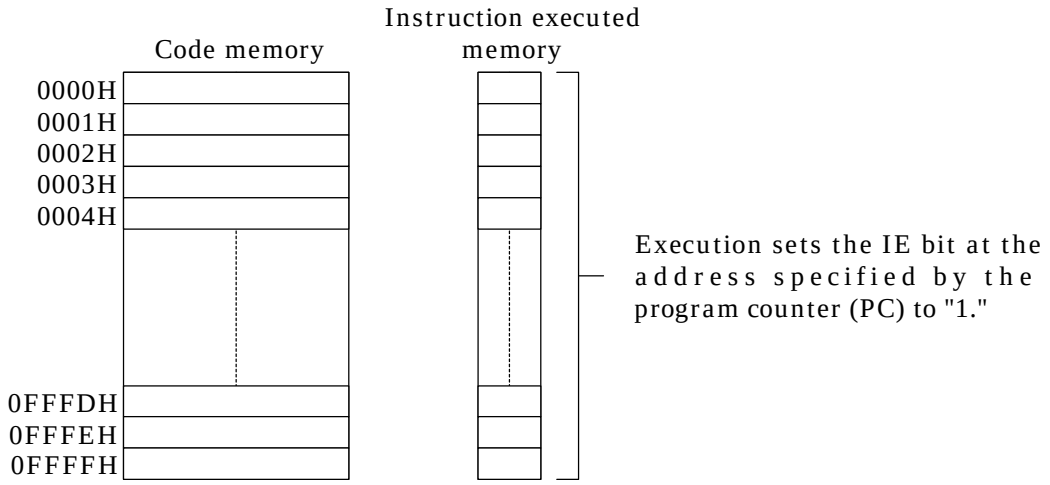


Figure 2-9 IE Bits

■ **Note 1** ■

Single-step emulation does not set IE bits.

2.6.2 Cycle Counter

The 24-bit cycle counter tracks the machine cycles of each instruction executed as a yardstick to user application program execution times.

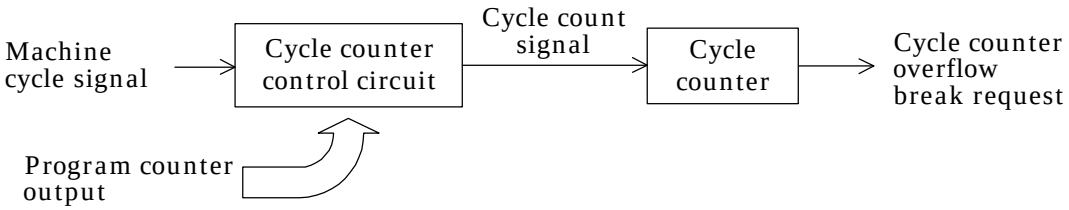


Figure 2-10 Cycle Counter

This counter offers three modes of operation.

- Count disabled
- Free-running count
- Trigger count

(1) Count disabled

No instructions are counted.

(2) Free-running count

Instructions are counted for all code memory addresses.

(3) Trigger count

All instructions between start and stop triggers based on code memory addresses are counted. There are three possible trigger combinations:

a. Specifying both a start and stop address

Counting starts when real-time emulation visits the former and stops when the program counter (PC) hits the latter.

b. Specifying a start address only

Counting starts when real-time emulation visits the former and continues until there is a break.

c. Specifying a stop address only

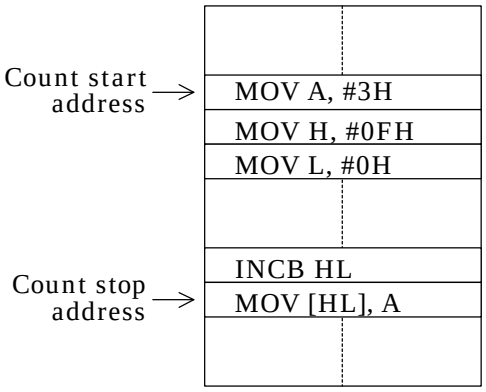
Counting starts simultaneously with real-time emulation and stops when the program counter (PC) hits the latter.

■ **Note 1** ■

Counting is enabled during real-time emulation and disabled during single-step emulation.

■ **Note 2** ■

The trigger fires just before execution of the instruction at the corresponding address, so the instruction at the start address is counted, but not the one at the stop address.



Counting covers from the MOV A,#3H instruction at the start address through to the INCB HL instruction preceding the stop address.

■ **Note 3** ■

Only start and stop address specifications corresponding to the first word of an instruction produce results. One specifying the second word of a two-word instruction or an entry in the ROM table yield a trigger that never fires.

2.7 Probe Cable

The emulator's probe cable carries the following six signals.

- Break (EXT.BRK) input
- Sync out (SYNC.OUT) output
- Trace (PROBE0 to PROBE3) inputs

2.7.1 Break Signal Input

If external breaks are enabled, a rising edge in this signal produces a break request.

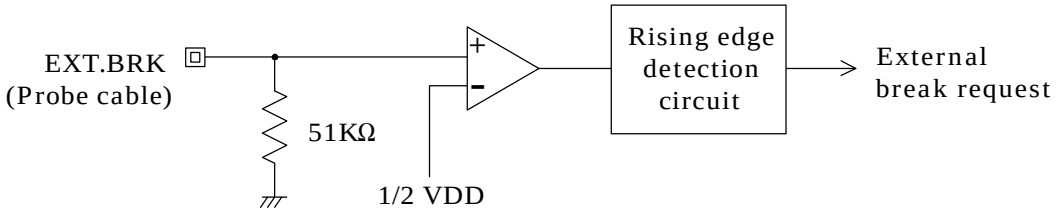


Figure 2-11 Break Signal Input

A built-in voltage conversion circuit converts the break (EXT.BRK) input level of VDD, the port interface power supply voltage (0.9 to 5.5 V), to the internal "H" level.

The break request coincides with the beginning of an instruction S2 cycle.

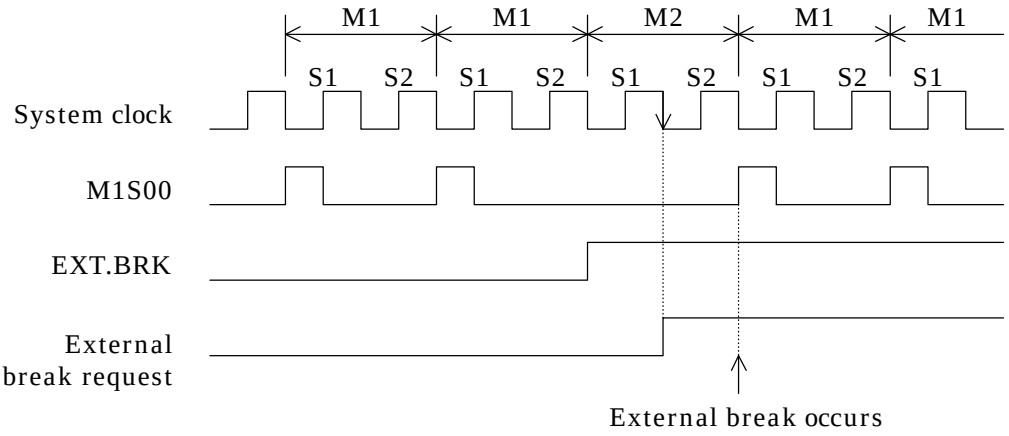


Figure 2-12 External Break Input Timing

2.7.2 Sync Out Signal

For each code memory address, the emulator provides a sync out bit that controls this output.

If a bit is “1,” execution of the instruction at the corresponding address pulls the probe cable sync out (SYNC.OUT) output to “L” level for the first half of an instruction S1 cycle.

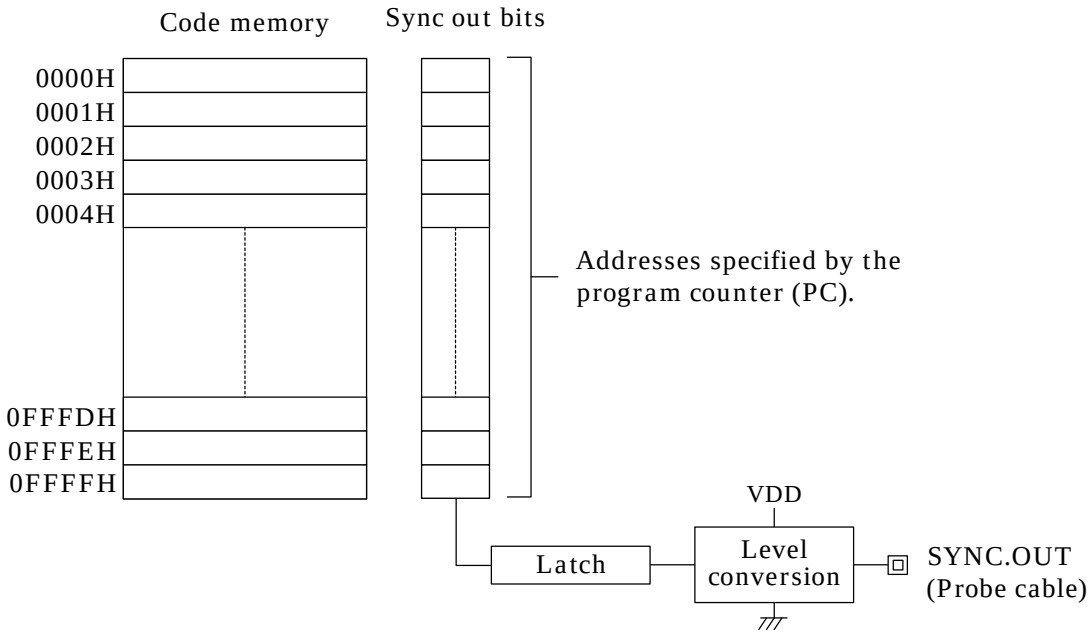


Figure 2-13 Sync Out Bits

A built-in voltage conversion circuit converts the sync out (SYNC.OUT) signal “H” level to VDD, the positive power supply voltage (0.9 to 5.5 V), for output.

■ **Note 1** ■

Only the sync out bit specifications corresponding to the first word of an instruction produce results. One specifying the second word of a two-word instruction or an entry in the ROM table is ignored.

2.7.3 Trace Inputs

These inputs are for tracing external signals during real-time emulation.

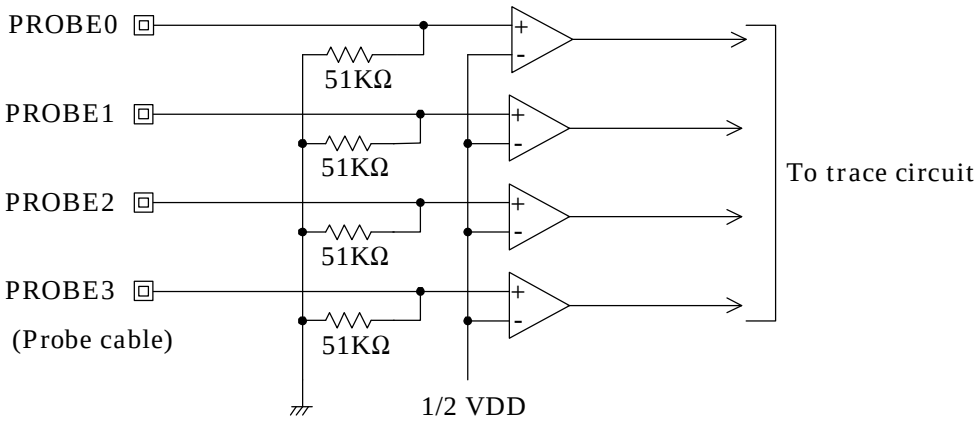


Figure 2-14 Trace Inputs

Built-in voltage conversion circuits convert the trace (PROBE0 to PROBE3) inputs to the internal “H” level from VDD, the positive power supply voltage (0.9 to 5.5 V).

Input is latched at the beginning of an instruction S1 cycle.

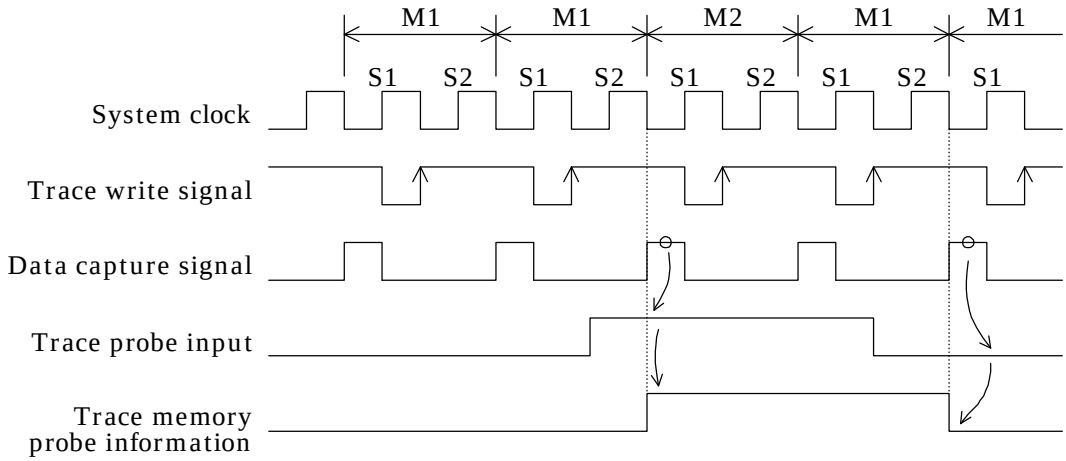


Figure 2-15 Trace Input Timing

2.8 Clock Switching

The emulator offers the choice of internal or external clock signals to the XT (low-speed) and OSC (high-speed) pins on the evaluation chip. These clocks can be selected from the following: the Dr.63514 clock and the clocks input to the USR.TXT and USR.OSC pins on the user cable.

Low-speed (XT)
- Internal (32.768kHz)
- External (user cable USR.XT pin)

High-speed (OSC)
- Internal (2MHz)
- External (user cable USR.OSC pin)

The microcontroller's time-base counter always uses the XT input to generate clock signals for the onboard peripherals. Its CPU, however, offers a choice of clock speeds: XT or OSC.

Using the faster (OSC) clock signal, however, introduces the risk of losing synchronization with the lower (XT) during single-step emulation or in the course of repeated breaks during real-time emulation. Results from timers and other onboard peripherals can therefore differ from those obtained during continuous execution with real-time emulation.

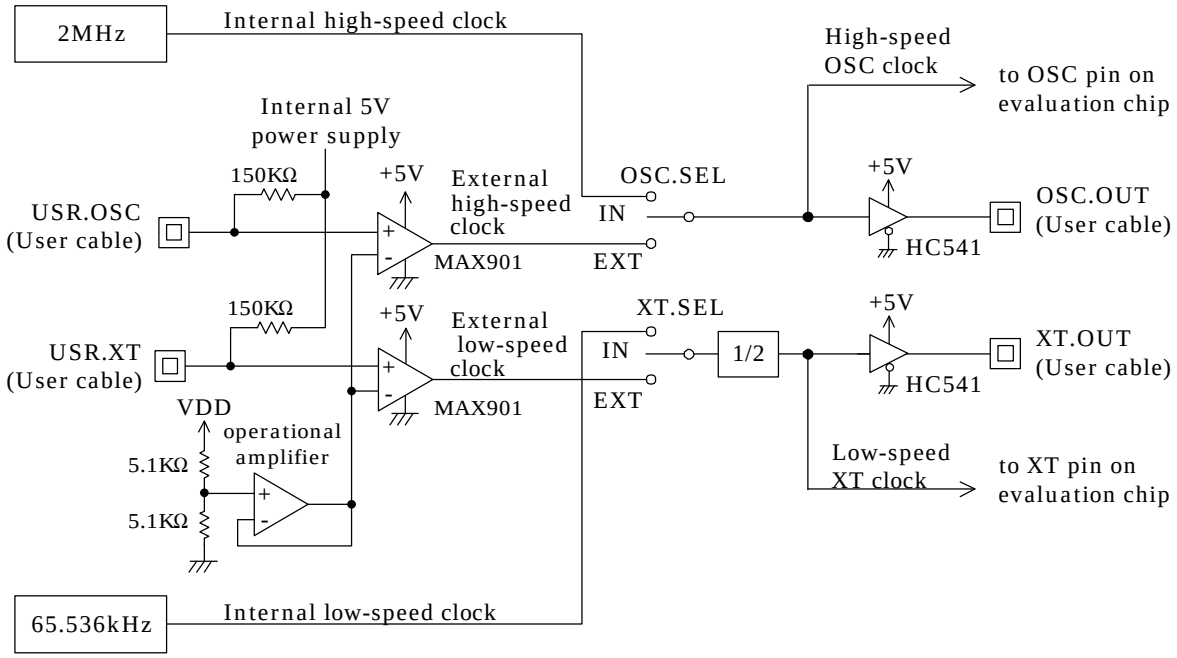
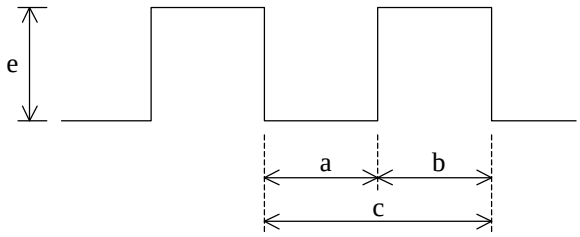


Figure 2-17 Clock Circuits

The XT.SEL and OSC.SEL switches control clock signal selection.

High-speed (OSC) clock		Low-speed (XT) clock	
OSC.SEL	**Selected clock**	**XT.SEL**	**Selected clock**
IN	Internal high-speed clock	IN	Internal low-speed clock
EXT	External high-speed clock	EXT	External low-speed clock

External high-speed (USR.OSC) and low-speed (USR.XT) clock signals must have the following waveform.



Duty ratio: a:b = 1:1
Voltage: e = 0.9 V ~ 5.5 V
Frequency: USR.XT c = 60 kHz to 160 kHz
USR.OSC c = 350 kHz to 2 MHz

■ **Note 1** ■

In the ML63512/514, after starting the high-speed clock oscillator has been specified (by setting the FCON ENOSC bit to 1), a period of at least 300 μs in RC oscillator mode and 10 ms in ceramic oscillator mode is required until the circuit enters the stable oscillation state. On the other hand, since the oscillator is started after power is applied when the Dr.63514 in-circuit emulator is used, the clock signal is supplied immediately when oscillator start has been specified by setting the FCON ENOSC bit to 1.

■ **Note 2** ■

The Dr.63514 in-circuit emulator does not support the mask options related to clocks.

■ **Note 3** ■

The low-speed clock uses a signal that is created by dividing the input clock frequency by 2. (The internal low-speed clock is 32.768kHz signal that is created by dividing 65.536kHz signal by 2.) Therefore, be sure to supply a clock whose frequency is twice the desired frequency when providing from external low-speed clock.

■ **Note 4** ■

The operating voltage of the XT.OUT and OSC.OUT pins is +5 V.

2.9 Reset Input Switching

The reset signal to the evaluation chip in the emulator normally comes from the emulator's main control CPU. There is, however, a setting for adding user cable reset (RESETB) input.

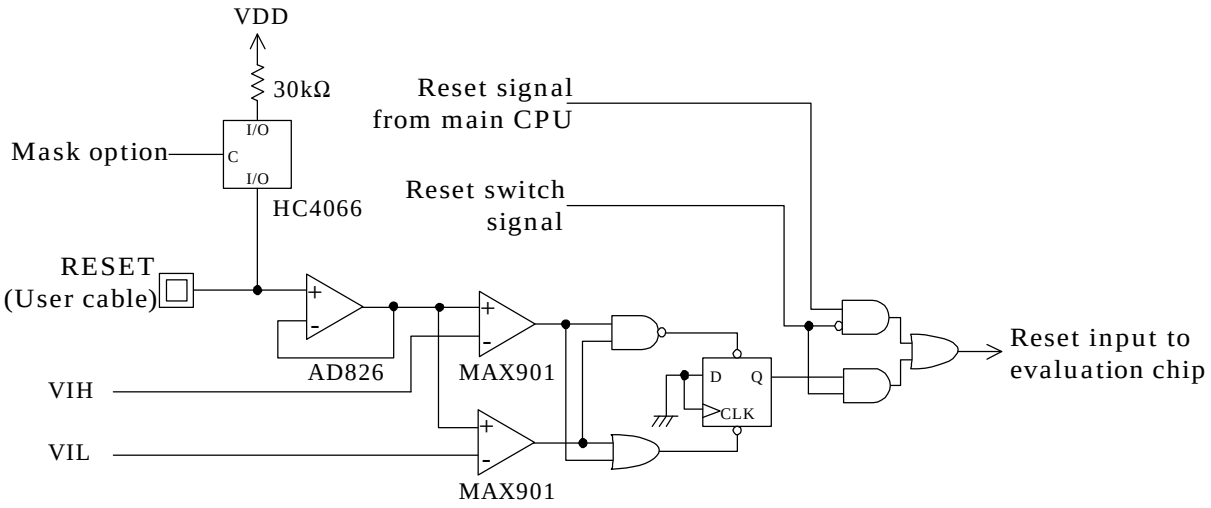


Figure 2-17 Reset Input

A built-in voltage conversion circuit converts the user cable reset (RESETB) input to the internal “H” level from VDD, the positive power supply voltage (0.9 to 5 V).

■ **Note 1** ■

User cable reset (RESETB) input is only relevant during evaluation and real-time emulation. It is always prohibited during single-step emulation.

■ **Note 2** ■

The RESETB pin is only enabled during periods when the RUN LED is lit. In particular, reset signal input to the RESETB pin from the user application system is disabled while the Dr.63514 in-circuit emulator is initializing and after real-time emulation has been forcibly terminated.

2.10 Mask Option

(1) RESET Terminal

In the ML63512/514, a internal pull-up resistor or an external pull-up resistor can be selected for the reset pin in the mask options. This option is set by defining location 0FE2H (1FE2H) in the mask option allocated data area in the test data area. On the other hand, in the Dr.63514 in-circuit emulator, this is implemented by defining location 0FE2H (1FE2H) in code memory in the same manner as in the ML63512/514. Note that the mask option allocated data area can be read and written with commands.

(2) CLOCK Terminal

In the ML63512/514, the following options can be specified in the mask options: a crystal oscillator or an RC oscillator for the low-speed clock, an RC oscillator or a ceramic oscillator for the high-speed clock, and an internal or external capacitor for the high-speed RC oscillator. However, these functions are not supported in the Dr.63514 in-circuit emulator.

■ **Reference** ■

See the ML63512/514 User's Manual for the procedures for defining the mask options.

2.11 Comparator Selection Function

The P7.0 and P7.1 pins in the ML63512/514 can be used as input ports or comparators. The Dr.63514 implements the input port block with discrete components and the comparator block with an ML63514 itself. However, the discrete components (circuits) in the input block influence the characteristics of the comparator. Therefore, be sure to disconnect the port circuit connection with the CMP switch when using the comparator. This will achieve comparator characteristics identical to those of the ML63512/514.

Used function	CMP switch
Input ports	OFF
Comparator	ON

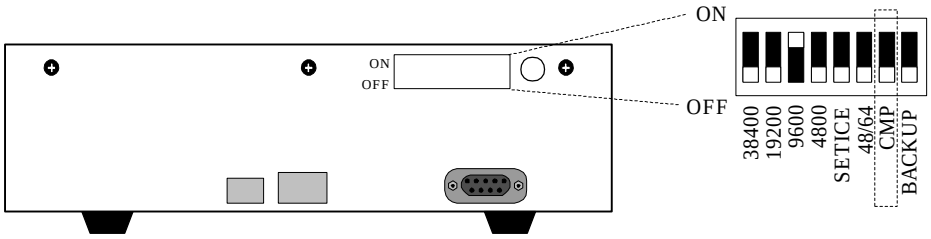


Figure 2-18 CMP Switch

■ **Note 1** ■

Since the P7.0 and P7.1 pin input port circuits are not connected when the comparator usage is set (i.e. when the CMP switch is on), they cannot be used as input ports in this mode.

■ **Note 2** ■

With the input port usage setting (i.e. when the CMP switch is off), although comparator can be used, the Dr.63514 cannot provide the same characteristics as the ML63512/514.

2.12 Backup Switching

The ML63512/514 includes a built-in backup circuit that doubles the power-supply voltage. The backup circuit is used when the positive power-supply voltage (VDD) falls below 1.8 V. The Dr.63514 in-circuit emulator includes an ML63514 to implement the level detector and comparator functions. The positive power-supply voltage (VDD) in this ML63514 depends on the setting of the VDD.SEL switch, and has the same external input voltage range as the ML63512/514, namely 0.9 to 5.5 V. The backup circuit on/off setting for the ML63514 used in the Dr.63514 in-circuit emulator must be set appropriately as well. The BACKUP switch sets the backup state of the ML63514 used in the Dr.63514 in-circuit emulator.

operating condition for the operating voltage	Backup Switch
1.5 V internally or 0.9 to 1.8 V externally	ON
1.8 to 5.5 V externally	OFF

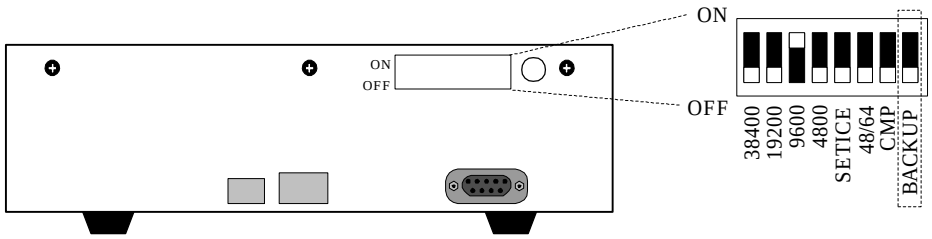


Figure 2-19 Backup Switch

■ **Note 1** ■

This BACKUP switch is completely unrelated to the backup function of the ML63512/514 being evaluated. The ML63512/514 backup function is implemented using only the BUPCON register in the Dr.63514 in-circuit emulator.

Caution

The BACKUP switch must be set to the operating voltage used. If the switch is set to the wrong position, the ML63514 used in the Dr.63514 in-circuit emulator may be destroyed.

2.13 Package Type Setting

The ML63512/514 is provided in two different packages with differing pin counts (a 48-pin TQFP package and a 64-pin TQFP package), and due to that difference, ports 6, 9, and A are either available as I/O pins or are not available. The Dr.63514 in-circuit emulator supports this difference with the 48/64 switch. When the 48/64 switch is turned off, the connections to the port 6, 9, and A pins in the user cable are disconnected and signal I/O is disabled. Inversely, when set to the on position, the pins are connected and signal I/O is enabled. Set this switch according to the package type you will be using.

PORT6,9,A	Package	48/64 Switch
None	48-pin TQFP	OFF
Present	64-pin TQFP	ON

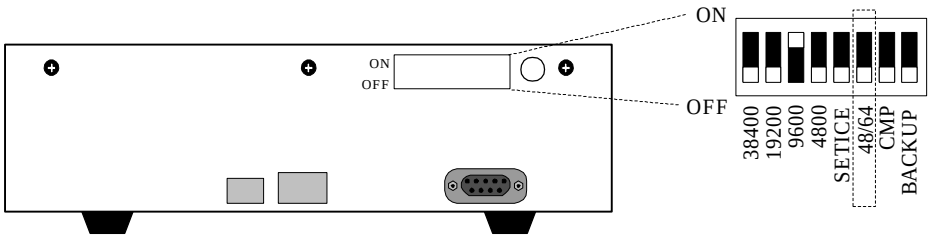


Figure 2-20 48/64 Switch

2.14 Operating power supply selection function

Except for port 8 and the monitor pins, the voltage on all pins in the user cable and all signal in the probes depend on this operating power supply setting.

- emulator internal power supply (1.5 V)
- external power supply (power is taken from the USR.VDD pin in the user cable)

Operating power supply selection function: Except for port 8 and the monitor pins, the voltage on all pins in the user cable and all signal in the probes depend on this operating power supply setting.

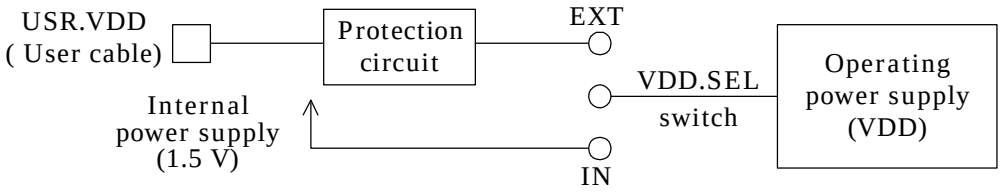


Figure 2-21 Operating power supply selection circuit

The VDD.SEL switch switches between internal and external power supply operation. When the VDD.SEL switch is set to the IN position, the emulator internal power supply (1.5 V) is selected, and when it is set to the EXT position, the external power supply operation is selected, i.e., power is taken from the USR.VDD pin in the user cable. The user cable USR.VDD pin protection circuit is provided to prevent internal damage to the emulator if the power-supply voltage is applied to the user cable USR.VDD pin before the emulator power supply is turned on.

Caution

The USR.VDD input voltage must be between 0.9 and 5.5 V. Using a voltage outside this range risks damaging the evaluation chip.

2.15 Internal Signal Monitoring

The user cables provide pins for monitoring the following internal signals.

- Halt mode (HALT.OUT) signal
- Low-speed clock (XT.OUT) signal
- High-speed clock (OSC.OUT) signal

(1) Halt mode (HALT.OUT) signal

“H” level output indicates that the evaluation chip is in halt mode.

(2) Low-speed clock (XT.OUT) signal

This pin monitors the low-speed (XT) clock signal to the evaluation chip.

(3) High-speed clock (OSC.OUT) signal

This pin monitors the high-speed (OSC) clock signal to the evaluation chip.

■ **Note 1** ■

These three signals use the emulator's internal operating voltage (5 V) for their “H” level.

2.16 BAUD Switches

The BAUD switches at the rear of the emulator offer a choice of eight baud rates from 4800 to 115200 bps.

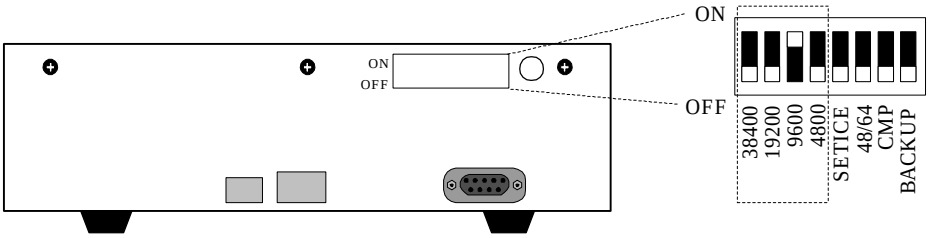


Figure 2-22 BAUD Switch

The baud rate settings are as follow.

BAUD switch / baud rate	38400	19200	9600	4800
4,800bps	OFF	OFF	OFF	ON
9,600bps	OFF	OFF	ON	OFF
19,200bps	OFF	ON	OFF	OFF
38,400bps	ON	OFF	OFF	OFF
51,200bps	ON	ON	ON	OFF
57,600bps	ON	ON	OFF	ON
76,800bps	ON	OFF	ON	ON
115,200bps	OFF	ON	ON	ON

All other serial interface parameters are fixed: 8 bits, no parity, 1 stop bit, XON/XOFF flow control.

■ **Note 1** ■

The IBM PC/AT and compatibles do not support the 51,200bps and 76,800bps speeds. When turning on or resetting the Dr.63514, if an initialization message is not produced on the dedicated debugger's screen or a communication error occurs, lower both the Dr.63514 baud rate setting and the dedicated debugger's speed parameter.

2.17 LED Indicator

The emulator has five LEDs.

Label	Color	Meaning
POWER	Red	Power supply
RUN	Green	Execution
ERROR	Red	Error
VCHECK	Red	Voltage check
HALT	Orange	Halt mode

(1) POWER
This LED lights when power is being supplied to the emulator.

(2) RUN
This LED lights during real-time emulation.

(3) ERROR
This LED lights when an error within the emulator prevents correct operation.

(4) VCHECK
This LED lights when power supply voltage falls below 0.7V in USR.VDD pin.

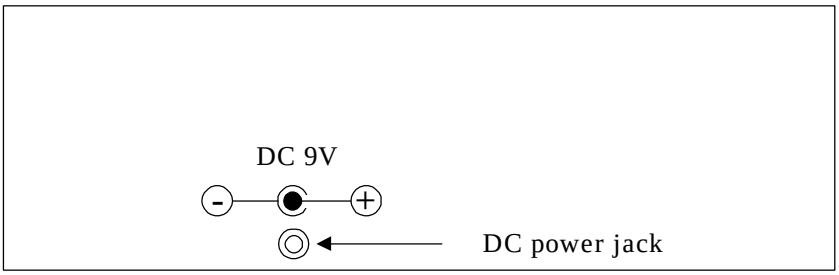
(5) HALT
This LED lights when the evaluation chip is in halt mode.

■ **Note 1** ■

If the ERROR LED lights, or if all LEDs other than HALT light, try the appropriate troubleshooting procedure from Appendix 8, “If Emulator Doesn't Start.”

2.18 Power Supplies

The Dr.63514 in-circuit emulator can be operated from a DC power supply. The TAC-2 AC adapter provide with the emulator can be used to provide the required 9 V DC power.



Caution

Do not use any DC power supply other than the provided AC adapter. Damage to the emulator or fire can result if power that does not meet the required specifications is used.

Caution

NEVER REVERSE THE POLARITY OF THE DC POWER SUPPLY INPUT. DOING SO DAMAGES THE EMULATOR.

Chapter 3. Setting Up and Starting Up

1. Setting up and turning on the Dr.63514 in-circuit emulator

This section describes the set up procedure that must be followed prior to turning on the Dr.63514 in-circuit emulator, and then the procedure for turning on the emulator.

1.1 Device configuration

Before using the emulator for debugging, use the dedicated emulator setup utility to configure it for the target microcontroller. This utility runs on the development host, transferring device information to the emulator over a serial cable. The BAUD switches at the rear of the emulator specify the transfer speed.

The emulator stores this configuration data in a built-in EEPROM, so reconfiguration is only necessary when the target microcontroller changes.

Before using the utility, set the SETICE switch at the rear of the emulator to its ON position, the one for updating this EEPROM.

SETICE switch	Operating mode
ON	Device configuration
OFF	Debugging (evaluation or emulation)

The ICE setup utility runs on the host computer, transferring device information to the Dr.63514 In-Circuit Emulator through the RS232C interface. The communication baud rate of the RS232C interface is set by the dipswitches (BAUD) on the rear panel of the Dr.63514 In-Circuit Emulator unit.

BAUD switch / baud rate	38400	19200	9600	4800
4,800bps	OFF	OFF	OFF	ON
9,600bps	OFF	OFF	ON	OFF
19,200bps	OFF	ON	OFF	OFF
38,400bps	ON	OFF	OFF	OFF
51,200bps	ON	ON	ON	OFF
57,600bps	ON	ON	OFF	ON
76,800bps	ON	OFF	ON	ON
115,200bps	OFF	ON	ON	ON

The emulator is now ready for the update.

■ **Note 1** ■

Set the emulator switches to the settings in the following table.

Switch label	Setting
MODE	EMU
VDD.VDDI.SEL	IN
OSC.SEL	IN
XT.SEL	IN
48/64	option
CMP	OFF
BACKUP	ON

■ **Note 2** ■

The IBM PC/AT and compatibles do not support the 51,200bps and 76,800bps speeds. When turning on or resetting the Dr.63514, if an initialization message is not produced on the dedicated debugger's screen or a communication error occurs, lower both the Dr.63514 baud rate setting and the dedicated debugger's speed parameter.

1.2 Switch and Jumper Settings

This section describes the switch and jumper settings needed before starting the emulator.

(1) MODE switch

Set the MODE switch to the desired mode.

Operating mode	MODE Switch
Emulation Mode	EMU
Evaluation Mode	EVA

(2) BAUD switch

Set the BAUD switch to the Host computer baud rate. However, note that this setting is not required if the emulator is operated in evaluation mode.

BAUD switch / baud rate	38400	19200	9600	4800
4,800bps	OFF	OFF	OFF	ON
9,600bps	OFF	OFF	ON	OFF
19,200bps	OFF	ON	OFF	OFF
38,400bps	ON	OFF	OFF	OFF
51,200bps	ON	ON	ON	OFF
57,600bps	ON	ON	OFF	ON
76,800bps	ON	OFF	ON	ON
115,200bps	OFF	ON	ON	ON

■ **Note 1** ■

The IBM PC/AT and compatibles do not support the 51,200bps and 76,800bps speeds. When turning on or resetting the Dr.63514, if an initialization message is not produced on the dedicated debugger's screen or a communication error occurs, lower both the Dr.63514 baud rate setting and the dedicated debugger's speed parameter.

(3) VDD.SEL switch

Set the VDD.SEL switch as follows according to the way the operating voltage is supplied.

Operating voltage usage conditions	VDD.SEL
Internal (1.5V)	IN
External (from user cable USR.VDD pin)	EXT

(4) XT.SEL switch

Select the low-speed (XT) clock source.

Clock source	XT.SEL
Internal (32.768kHz)	IN
External	EXT

(5) OSC.SEL switch

Select the high-speed (OSC) clock source.

Clock source	OSC.SEL
Internal (2MHz)	IN
External	EXT

(6) 48/64 switch

Set the 48/64 switch as follows according to the package type used.

Port 6, 9 and A	Package	48/64 switch
None	48-pin TQFP	OFF
Present	64-pin TQFP	ON

(7) CMP switch

Set the CMP switch as follows according to the comparator usage.

Used function	CMP switch
Input ports	OFF
Comparator	ON

(8) BACKUP switch

Set the BACKUP switch as follows according to the way the operating voltage is supplied.

Operating voltage usage conditions	BACKUP switch
Internal 1.5 V or external 0.9 to 1.8 V	ON
External 1.8 to 5.5 V	OFF

(9) EPROM Socket

Insert the low-side and high-side EPROMs into which the user program has been stored in the Dr.63514 in-circuit emulator low-side and high-side EPROM sockets. However, note that the EPROMs do not need to be inserted if the emulator is operated in emulation mode.

1.3 Emulator Connections

Connect the Dr.63514 in-circuit emulator to its accessories and peripherals as shown in the figure.

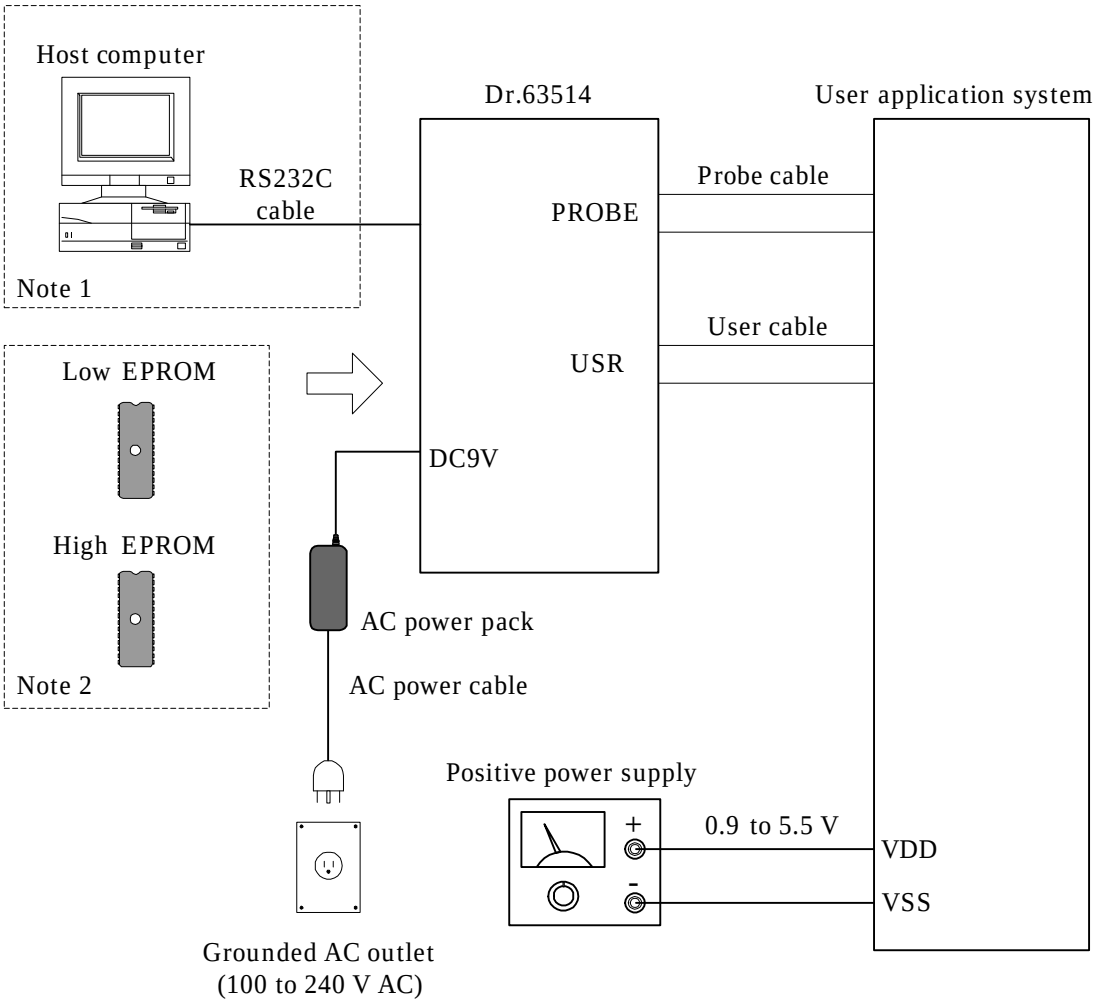


Figure 3-1 Connections for Emulation

■ **Note 1** ■

The host computer and the RS232C cable are not required when operating the emulator in evaluation mode.

■ **Note 2** ■

There is no need to insert the user program EPROMs when operating the emulator in emulation mode.

1.4 Powering Up

First make sure that

- the emulator is connected to the host computer,
- the emulator switches have been properly set,
- the emulator is connected to the user application system.

Follow the procedure below to start the emulator.

(1) Emulation mode

1. Load the dedicated debugger.
2. Turn on the power to the emulator.
3. Wait for the emulator's POWER LED to light.
4. Turn on the power to the user application system.

(2) Evaluation mode

1. Turn on the power to the emulator.
2. Wait for the emulator's POWER LED to light.
3. Wait for the emulator's RUN LED to light.
4. Turn on the power to the user application system.
5. Press the reset switch on the user application system.

Caution

Pay close attention to the sequence of applying power, or you could damage the emulator.

(1) When turning power on:
   a. Turn on power to theemulator.
   b. Turn on power to the user application system.
(2) When turning power off
   a. Turn off power to the user application system.
   b. Turn off power to the emulator.

Chapter 4. Additional Usage Notes

1. Debugging Notes

(1) Power on/off sequence

When a user application system is connected, always power up the emulator and then the user application system. Power down in the reverse order.

(2) Flag bits and start/stop addresses

Only breakpoint, trace enable, and sync out bit and start and stop address specifications corresponding to the first word of an instruction produce results. One specifying the second word of a two-word instruction or an entry in the ROM table is ignored.

(3) Terminating halt mode

A user break terminates halt mode if it is in effect. Restarting real-time emulation without specifying a starting address causes execution to resume from the instruction after the HALT instruction. Note that forcing resumption this way can produce different results from normal execution.

(4) User cable pins

In the Dr.63514 in-circuit emulator, the user-cable I/O pins have the following I/O circuits, and as a result, their I/O characteristics differ from the corresponding pins in the ML63512/514.

1. VDD,VSS terminal

The pins on the Dr.63514 in-circuit emulator user cable operate at the voltage level between the VDD (note 1) and the VSS pins. Either the emulator internal power supply (1.5 V) or an external power supply (input from the user cable USR.VDD pin) can be used for VDD. When using an external power supply, apply the desired operating voltage to the USR.VDD pin.

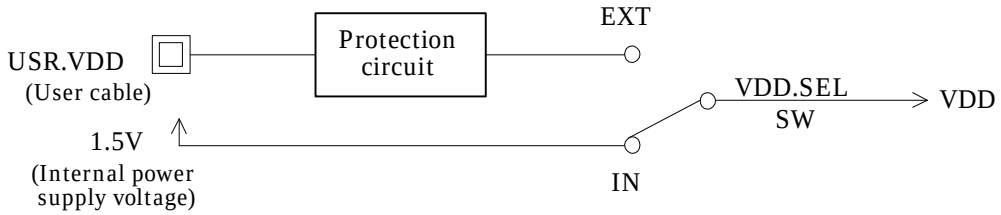


Figure 4-1 VDD circuit

■ **Note 1** ■

The term VDD used in this document refers to the voltage on the pins in the user cable as selected by the VDD.SEL switch.

Caution

The voltage supplied to the USR.VDD pin must be within the stipulated range (0.9 to 5.5 V). The Dr.63514 in-circuit emulator may be damaged or destroyed if a voltage outside the stipulated range is applied, or if no voltage is applied.

2. Input-output port (P0,P1,P2,P3,P4,P5,P6,P9,PA)

Ports 0 through 6, 9, and A in the ML63512/514 are I/O ports whose direction (input or output) can be selected in bit units. When input mode is selected, either a pull-up resistor input or a high-impedance input can be selected. When output mode is selected, either CMOS output or n-channel open-drain output can be selected.

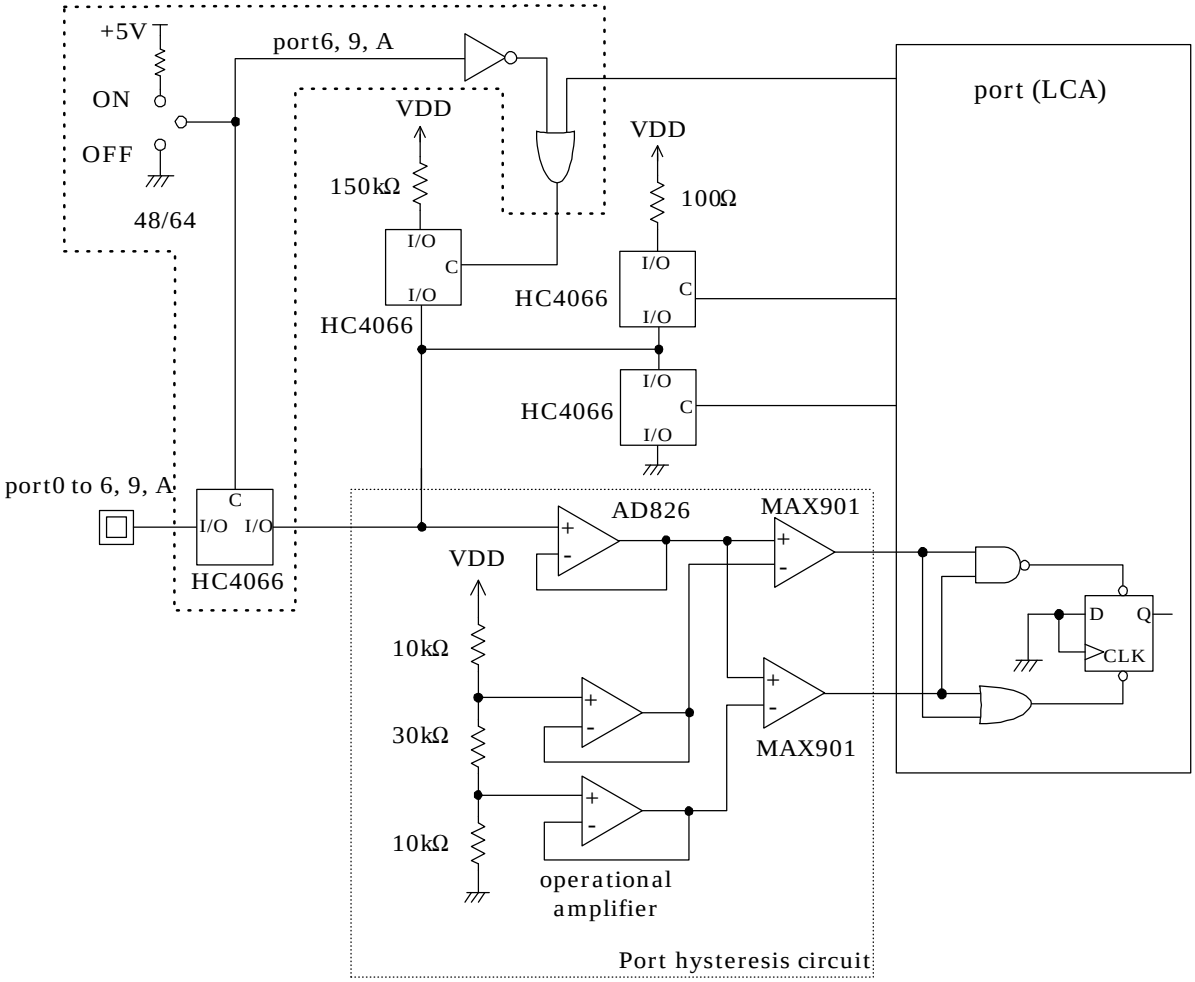


Figure 4-2 Input-output port

3. Input port (P7)

Port 7 in the ML63512/514 is an input port for which either pull-up resistor input or high-impedance input can be selected in bit units. Comparator and level detector input pin functions are also allocated to this port as secondary functions.

• Comparator port

The P7.0 and P7.1 pins in the ML63512/514 can be used as input ports or comparators. The Dr.63514 implements the input port block with discrete components and the comparator block with an ML63514 itself. However, the discrete components (circuits) in the input block influence the characteristics of the comparator. Therefore, be sure to disconnect the port circuit connection with the CMP switch when using the comparator. This will achieve comparator characteristics identical to those of the ML63512/514.

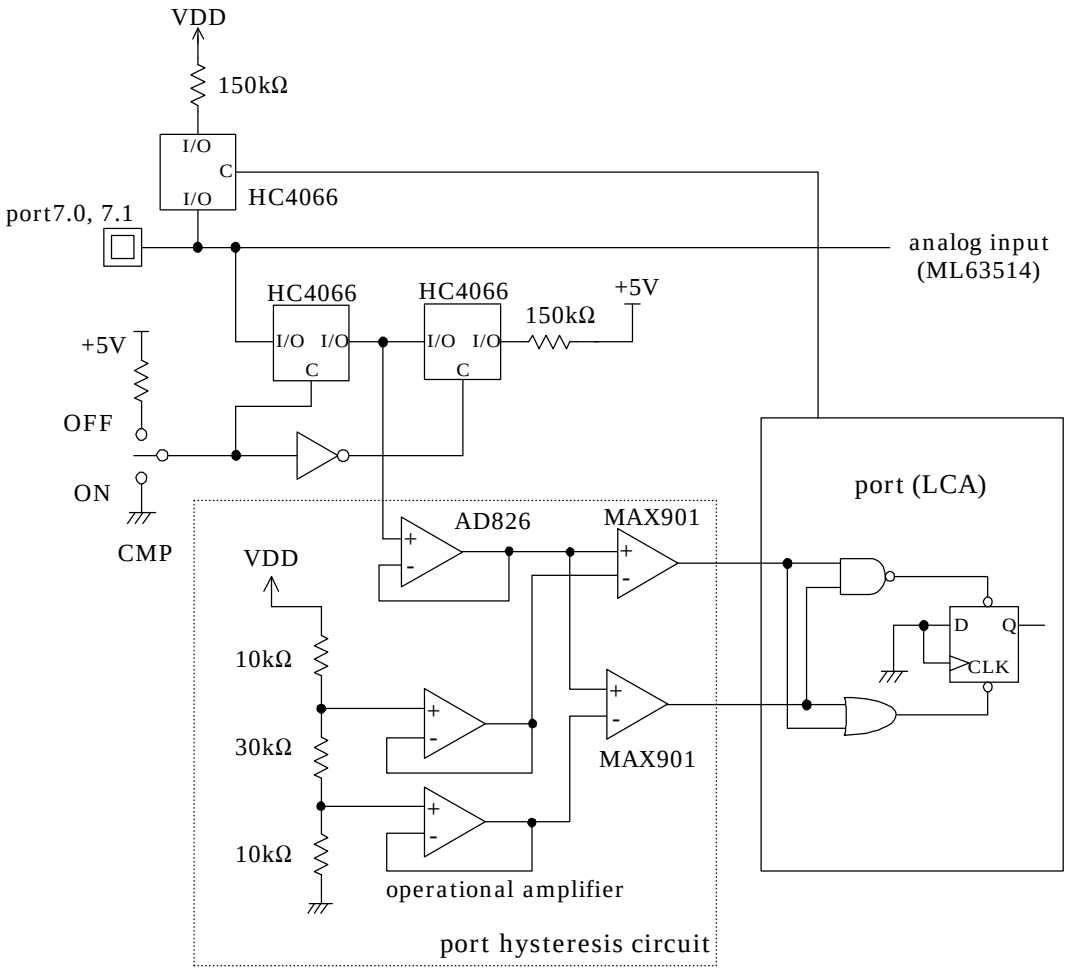


Figure 4-3 P7.0, P7.1

■ **Note 2** ■

Since the P7.0 and P7.1 pin input port circuits are not connected when the comparator usage is set (i.e. when the CMP switch is on), they cannot be used as input ports in this mode.

■ **Note 3** ■

With the input port usage setting (i.e. when the CMP switch is off), although comparator can be used, the Dr.63514 cannot provide the same characteristics as the ML63512/514.

• Level detector port

Pins P7.2 and P7.3 in the ML63512/514 can be used as either input ports or as level detector inputs. The Dr.63514 in-circuit emulator implements the input port block with discrete components and the level detector block with an ML63514. However, the discrete components (circuits) in the input block influence the characteristics of the level detector. Therefore, the port circuit is automatically disconnected when the level detector is operating. This will achieves level detector characteristics identical to those of the ML63512/514.

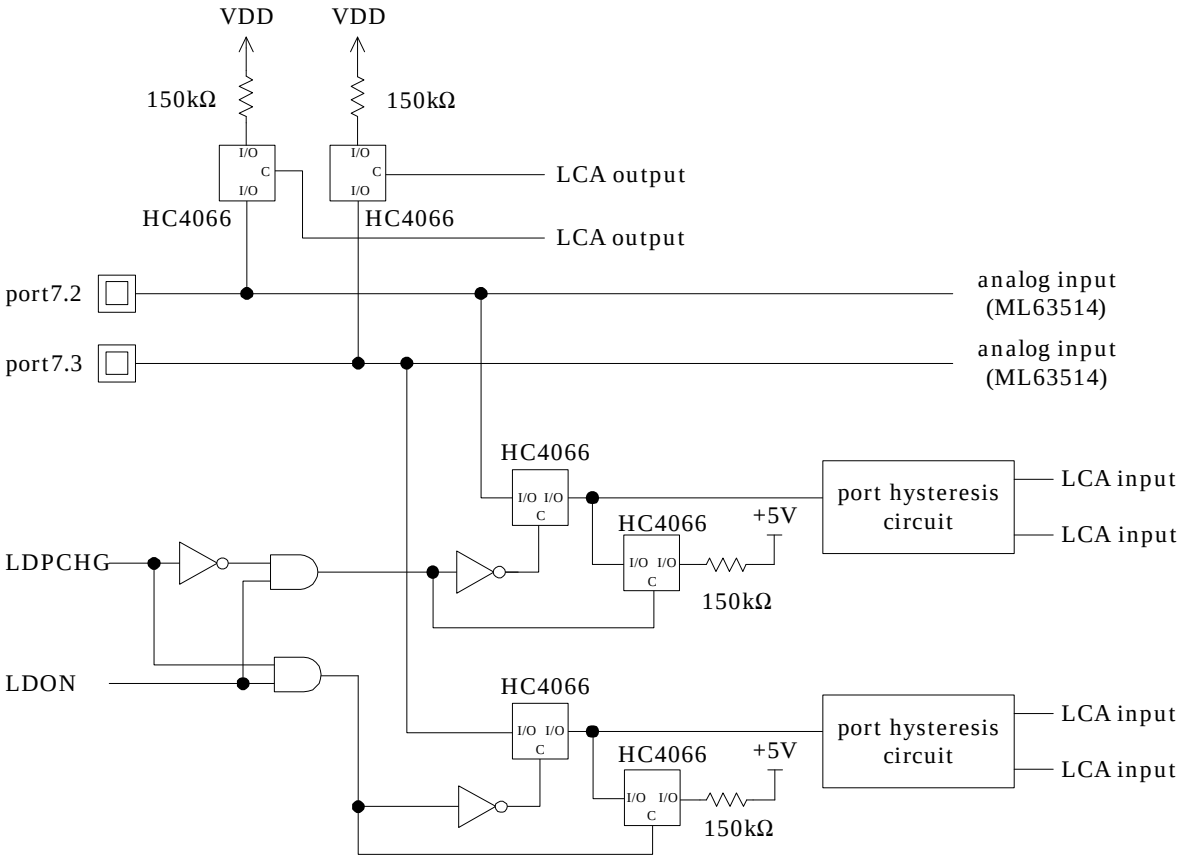


Figure 4-4 P7.2, 7.3

■ **Note 4** ■

Since the input port circuits are disconnected for ports used as level detectors during level detector operation, they cannot be used as input ports.

• Comparator/level detector circuit

In the Dr.63514 in-circuit emulator, the comparator and level detector blocks are implemented using an ML63514.

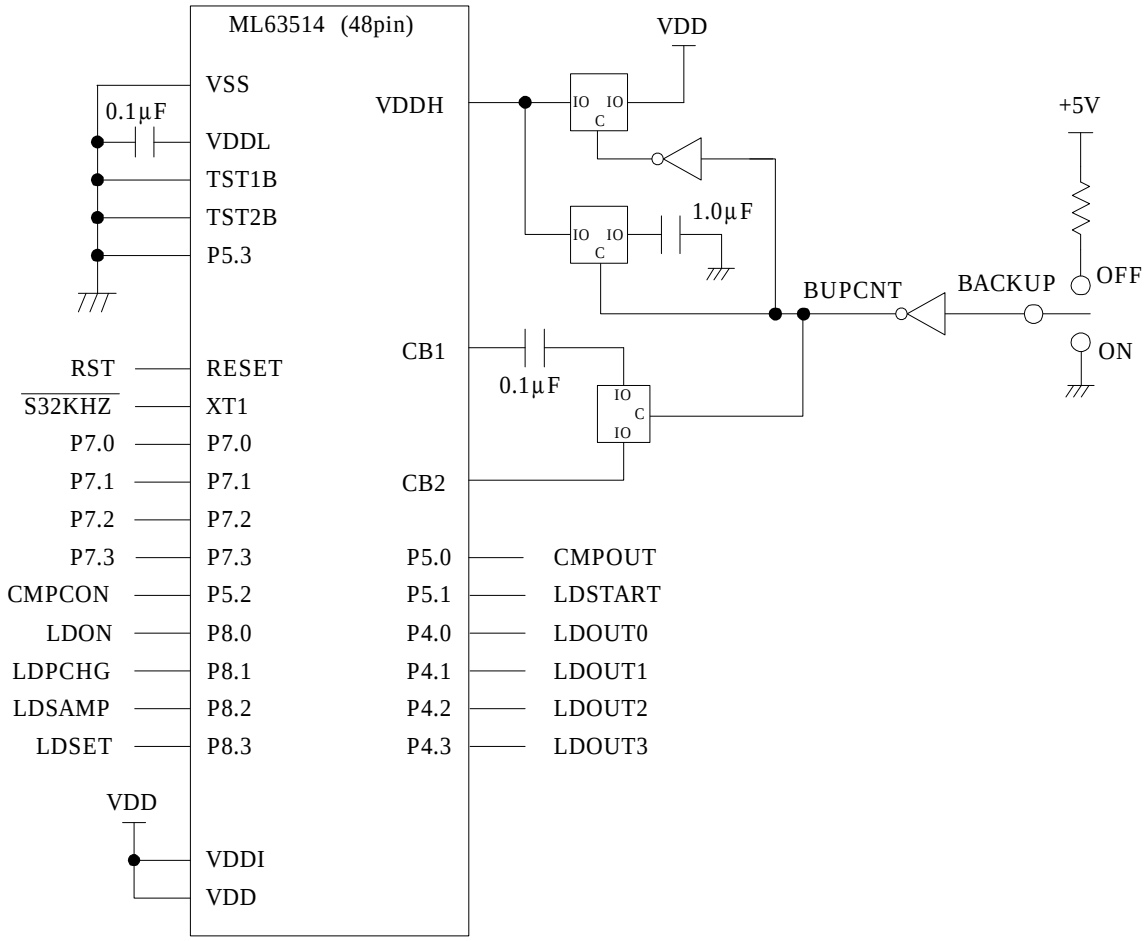


Figure 4-5 ML63514 test mode circuit

■ **Note 5** ■

This BACKUP switch is completely unrelated to the backup function of the ML63512/514 being evaluated. The ML63512/514 backup function is implemented using only the BUPCON register in the Dr.63514 in-circuit emulator.

Caution

The BACKUP switch must be set to the operating voltage used. If the switch is set to the wrong position, the ML63514 used in the Dr.63514 in-circuit emulator may be destroyed.

4. Output PORT

Port 8 in the ML63512/514 is an n-channel open-drain port that can drive LEDs.

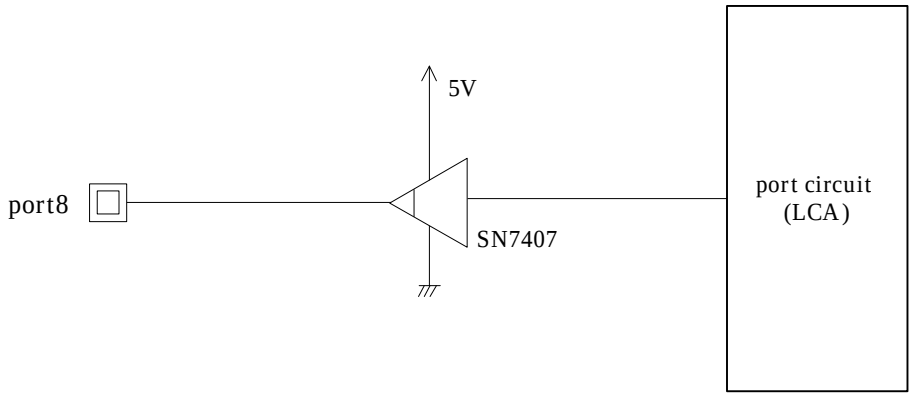


Figure 4-6 Output port

5. MD terminal

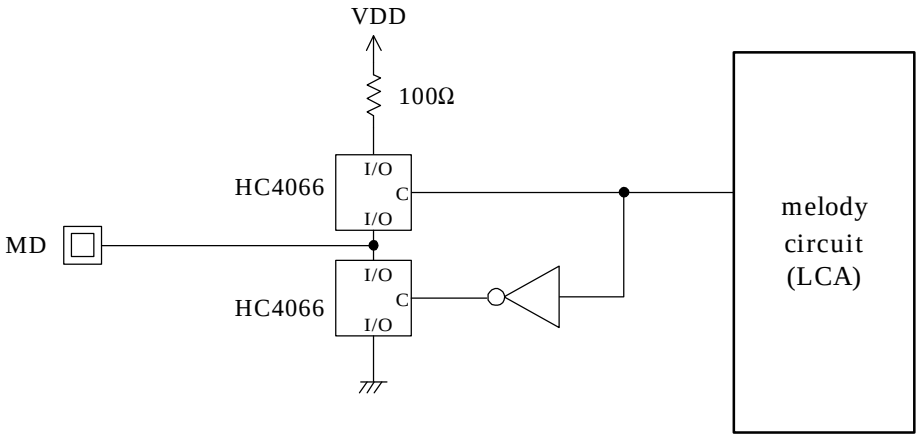


Figure 4-7 MD terminal

6. RESETB terminal

Reset inputs to the RESETB pin can be enabled or disabled with commands. This pin can be set to use either an internal pull-up resistor or and external pull-up resistor with the mask option function. Mask option data settings in the Dr.63514 in-circuit emulator are implemented by defining the mask option data allocation area at location 0FE2H (1FE2H) in code memory in the same manner as in the ML63512/514.

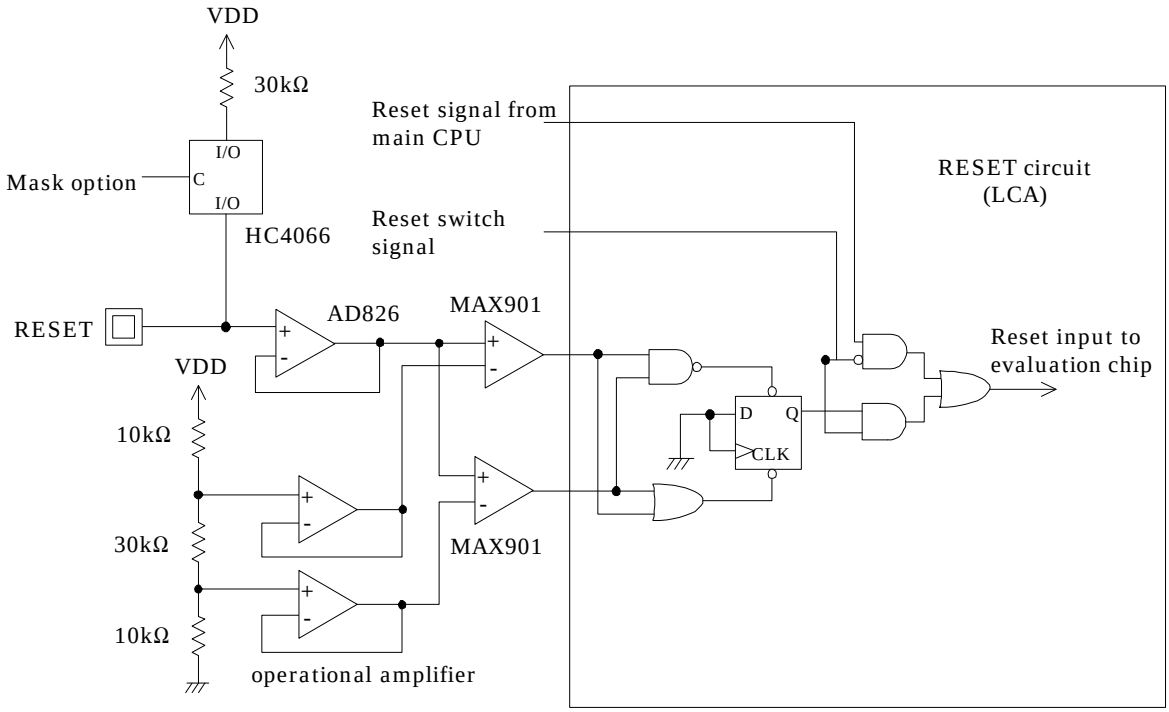


Figure 4-8 RESETB terminal

Also note that the input circuit discussed above have the following hysteresis characteristics.

Table 4-1 User interface circuit hysteresis characteristics

	ML63512/514		Dr.63514	
	VIL	VIH	VIL	VIH
VDD=1.5V	0.3V	1.2V	0.3V	0.7V
VDD=3.0V	0.6V	2.4V	0.5V	1.6V
VDD=5.0V	1.0V	4.0V	0.8V	2.6V

Table 4-2 Hysteresis value (rated value)

	ML63512/514(Typ.)	Dr.63514(Typ.)
VDD=1.5V	0.1V	0.40V
VDD=3.0V	0.5V	1.10V
VDD=5.0V	1.0V	1.80V

(5) Clock circuit

These clocks can be selected from the following.

Low-speed (XT) — Internal (32.768kHz) / External (user cable USR.XT pin)

High-speed (OSC) — Internal (2MHz) / External (user cable USR.OSC pin)

A built-in voltage conversion circuit converts the external clock input to the internal “H” level from VDD, the positive power supply voltage (0.9 to 5.5V).

These selected clocks are output to user cable for monitoring. These pins have output signal levels of 5V.

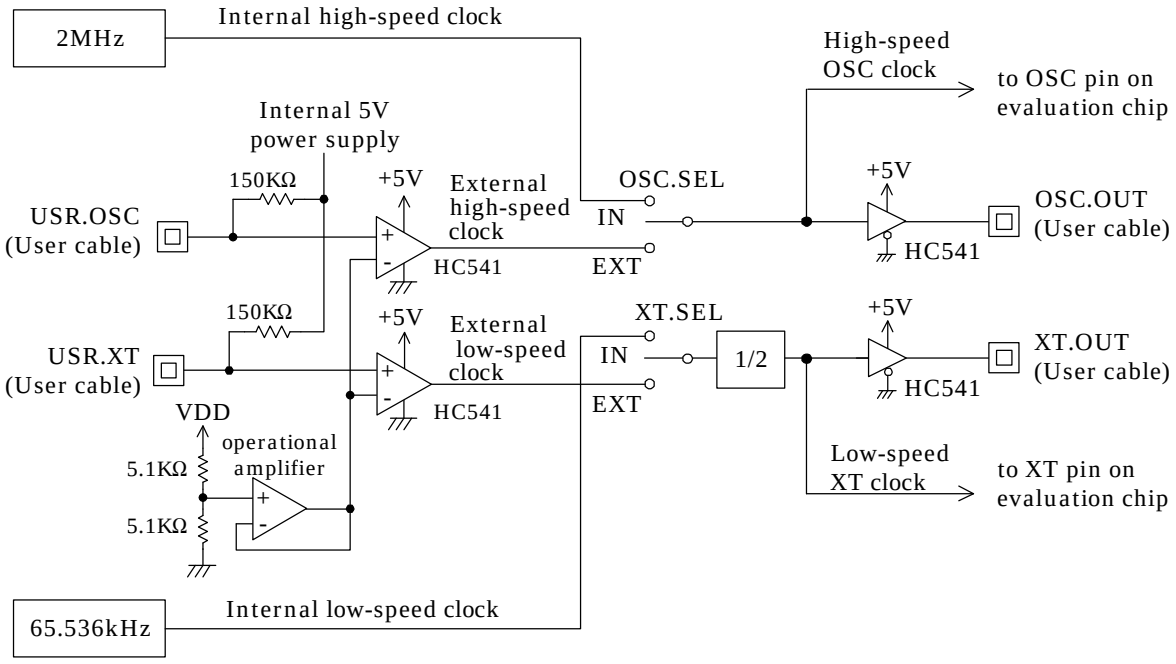


Figure 4-9 clock control part

2. Initialization

The following table summarizes the results of the initializations when the power is first applied and when the reset switch is pressed.

Item	Powering up	Reset
Evaluation chip	Same as for production versions of ML63512/514.	←
Break conditions	Breakpoint break, call stack overflow break, register stack overflow break	←
Breakpoint bits	All “0”	Unchanged
Break status	Dummy “No break status”	←
Trace memory	Blank	Unchanged
Trace condition	All addresses traced	←
Trace trigger	Free-running trace	←
Trace enable bits	All “0”	Unchanged
Trace pointer	Zero	Unchanged
Cycle counter	Zero	←
Cycle counter trigger	Free-running trace	←
User reset	Input disabled	←
Sync out bits	All “0”	Unchanged
IE bits	All “0”	Unchanged
Memory expansion	Expansion off	←

3. Operation Timing

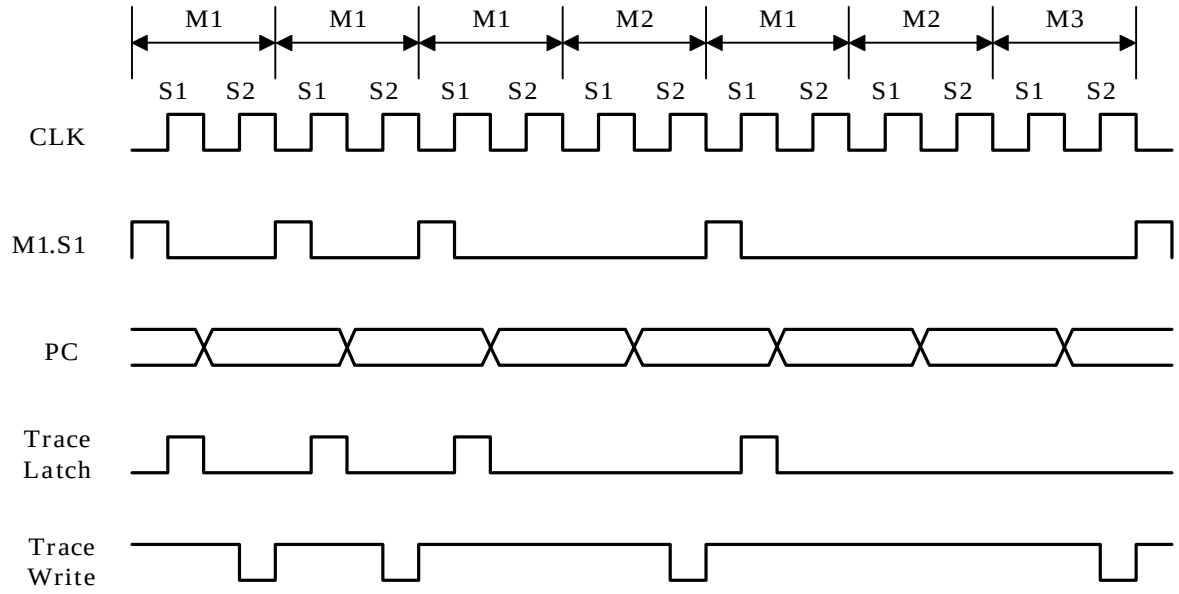


Figure 4-10 Trace Timing Chart

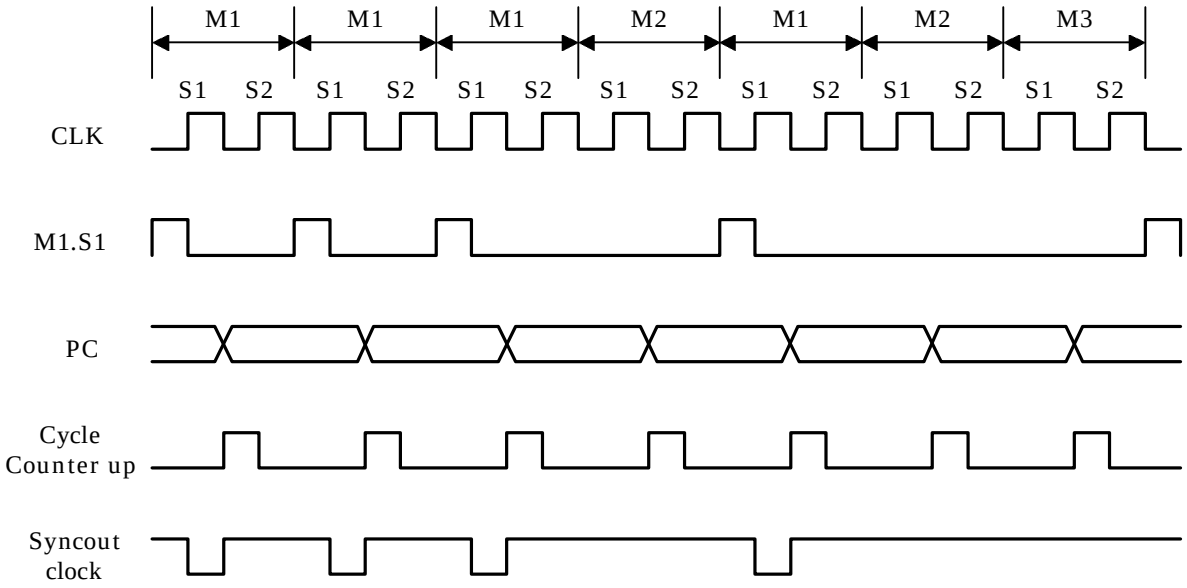


Figure 4-11 Cycle Counter/Sync Out Timing Chart

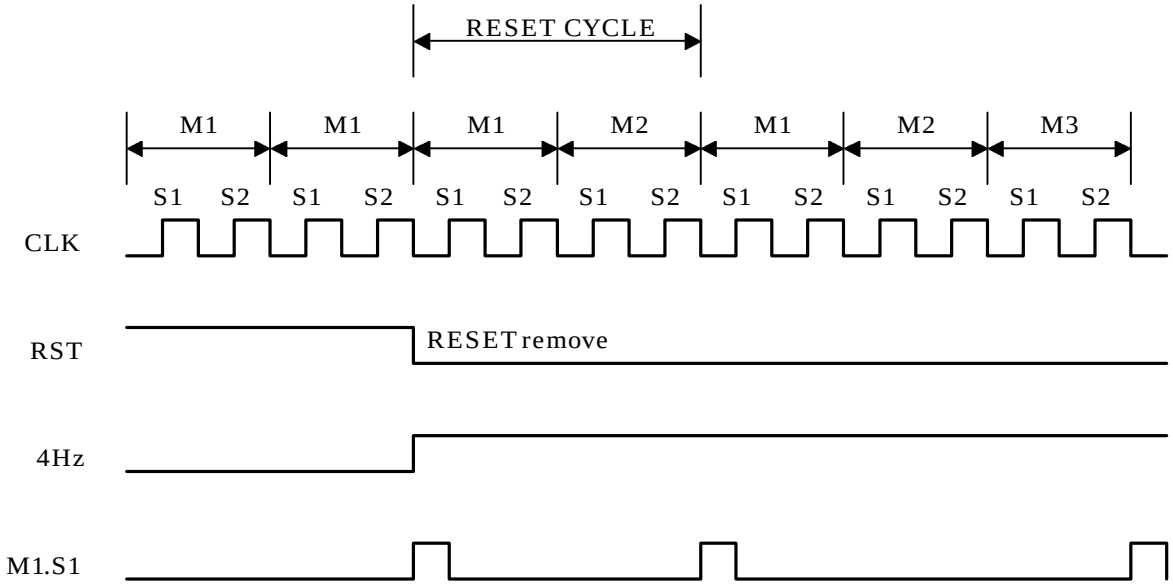


Figure 4-12 Reset Timing Chart

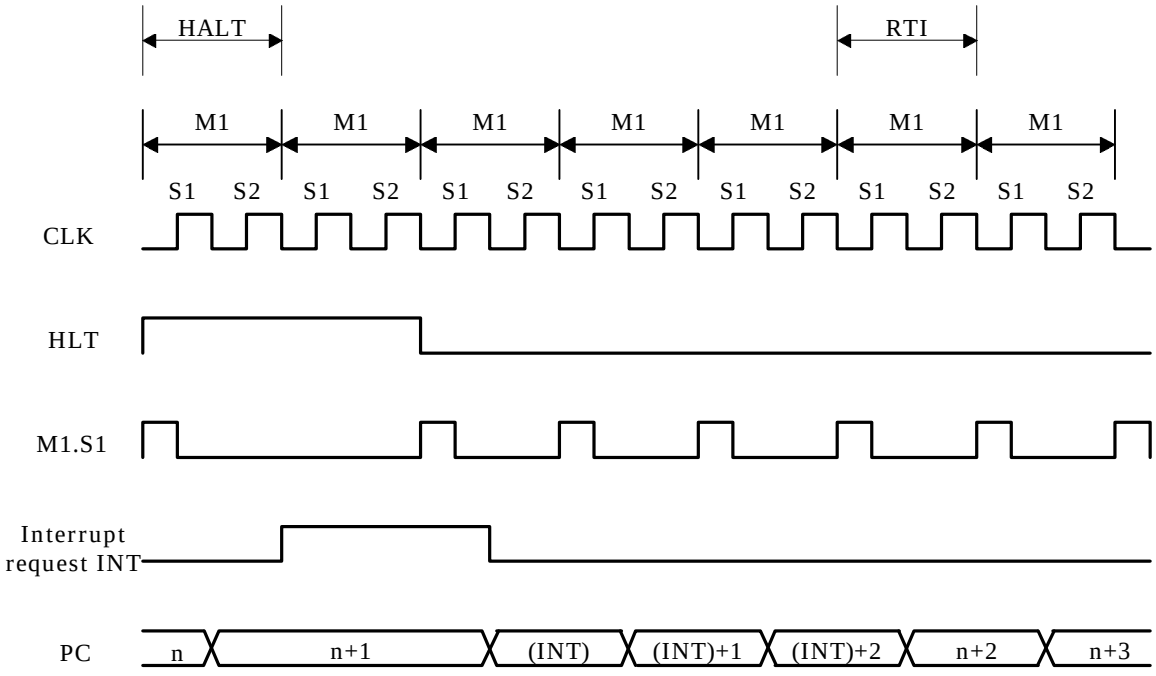
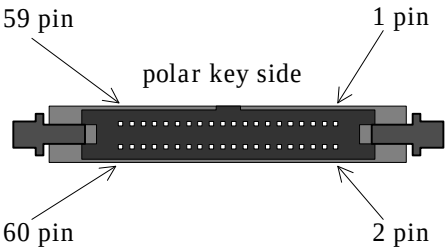


Figure 4-13 Halt Mode Timing Chart

Appendices

1. User Cable Connector Layout

The figure shows the structure of the Dr.63514 in-circuit emulator's user connector (a 60-pin connector). This user connector is connected to the user cable and used to connect to the user's application circuit.



The 60-pin connector shown in the figure at the left is the user connector. Pin 1 is at the upper right.

Figure A-1 USR connector layout

2. User cable layout

The figure below shows the structure of the user cable, which is an accessory for the Dr.63514 in-circuit emulator. This user cable is used attached to the Dr.63514 in-circuit emulator to connect to the user application circuit.

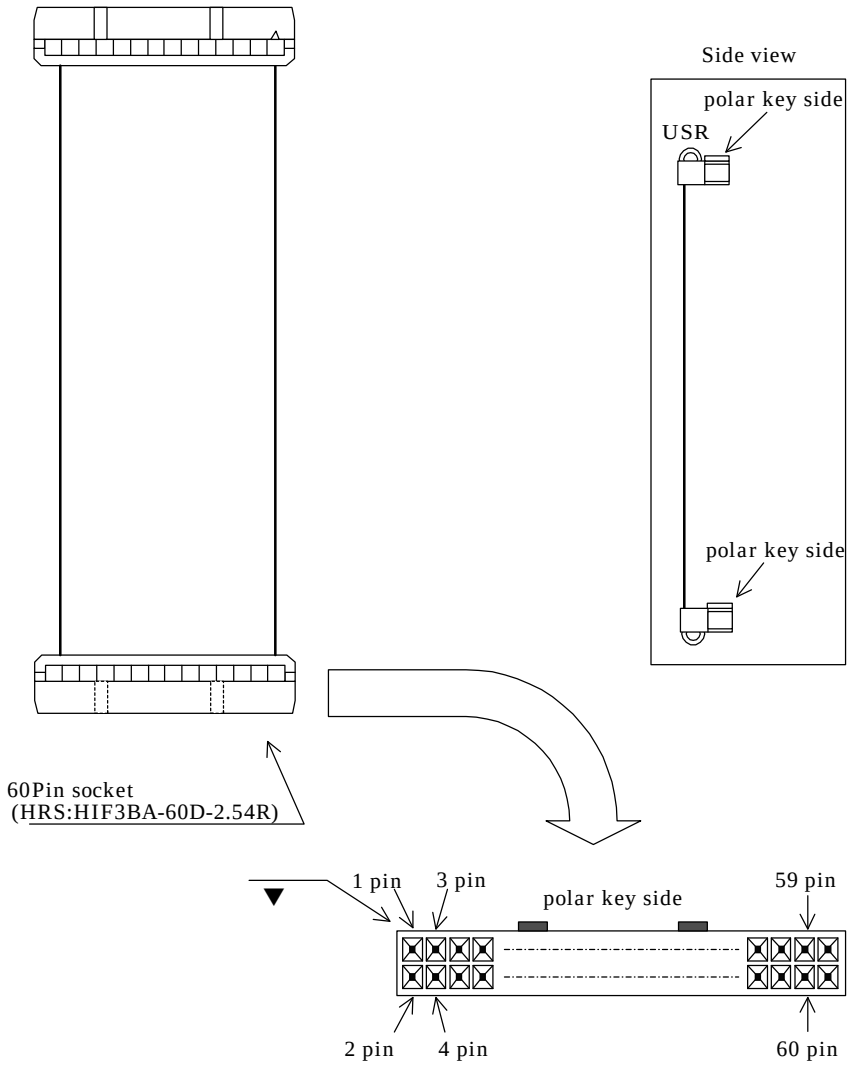


Figure A-2 User cable layout

■ **Note 1** ■

The HIF3BA-60D-2.54R (manufactured by Hirose Electronics, Ltd.) is used as the user application circuit connector socket for the user cable in the Dr.63514 in-circuit emulator. Therefore, either the HIF3BA-60PA-2.54DS (right angle pin header type) or the HIF3BA-60PA-2.54DSA (straight pin header type) from the same manufacturer must be used as the connector to connect to the user cable in the user application circuit.

3. User cable pin arrange

Of the pins in the ML63512/514, the TEST1B, TEST2B, XTO, XT1, OSC0, OSC1, OSCM, CB1, CB2, VDDL, VDDH, and VDDI pins are not implemented by the Dr.63514 in-circuit emulator.

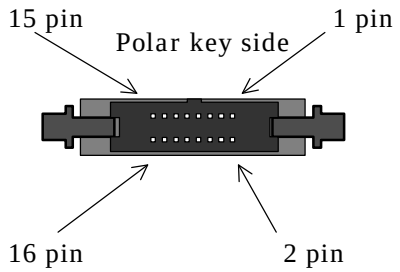
Table A-1 User Cable Pin List

UCN pin number	Signal name	Input-Output	Function	TQFP-48 pin number	TQFP-64 pin number
1	USR.VDD	I	positive power supply voltage	23	29
2					
3	P0.0/INT0	I/O	4-bit input-output port	37	51
4	P0.1/INT1			38	52
5	P0.2/INT2			49	53
6	P0.3/INT3			40	54
7	P1.0/TIM0CAP/TIM0OVF	I/O	4-bit input-output port	41	55
8	P1.1/TIM1CAP/TIM1OVF			42	56
9	P1.2/T0CK			43	57
10	P1.3/T1CK			44	58
11	P2.0/TBCCLK	I/O	4-bit input-output port	45	59
12	P2.1/HSCLK			46	60
13	P2.2			47	61
14	P2.3			48	62
15	P3.0/RXD	I/O	4-bit input-output port	1	3
16	P3.1/TXC			2	4
17	P3.2/RXC			3	5
18	P3.3/TXD			4	6
19	P4.0	I/O	4-bit input-output port	5	7
20	P4.1			6	8
21	P4.2			7	9
22	P4.3			8	10
23	P5.0	I/O	4-bit input-output port	9	11
24	P5.1			10	12
25	P5.2			11	13
26	P5.3			12	14
27	P6.0	I/O	4-bit input-output port	-	15
28	P6.1			-	16
29	P6.2			-	17
30	P6.3			-	18

UCN pin number	Signal name	Input-Output	Function	TQFP-48 pin number	TQFP-64 pin number
31	P7.0/CMPIN	I	4-bit input port	13	19
32	P7.1/CMPREF			14	20
33	P7.2/LDIN0			15	21
34	P7.3/LDIN1			16	22
35	P8.0	O	4-bit output port	17	23
36	P8.1			18	24
37	P8.2			19	25
38	P8.3			20	26
39	MD	O	melody output pin	36	46
40	RESETB	I	reset input pin	35	45
41	N.C	-		-	-
42	USR.XT	I	external low-speed clock input pin	-	-
43	N.C	-		-	-
44	USR.OSC	I	external high-speed clock input pin	-	-
45	N.C	-		-	-
46	HALT	O	monitor pin	-	-
47	N.C	-		-	-
48	XT.CLK	O	monitor pin	-	-
49	N.C	-		-	-
50	OSC.CLK	O	monitor pin	-	-
51	P9.0	I/O	4-bit input-output pin	-	47
52	P9.1			-	48
53	P9.2			-	49
54	P9.3			-	50
55	PA.0	I/O	4-bit input-output pin	-	63
56	PA.1			-	64
57	PA.2			-	1
58	PA.3			-	2
59	VSS(GND)	I	negative power supply voltage	22	28
60					

4. Probe Cable Connectors and Pin Layout

The probe connector on left side of the emulator is for the probe cable.



The 16-pin connector shown in the figure at the left is the probe connector. Pin 1 is at the upper right.

Figure A-3 Probe connector layout

Table A-2 Probe connector pin list

Pin number	Probe color	Name	Pin number	Probe color	Name
1	Black	PROBE0	9	Yellow	SYNC.OUT
2	-	VSS	10	-	VSS
3	Brown	PROBE1	11	Green	EXT.BRK
4	-	VSS	12	-	VSS
5	Red	PROBE2	13	Blue	VSS
6	-	VSS	14	-	VSS
7	Orange	PROBE3	15	Purple	VSS
8	-	VSS	16	-	VSS

■ **Note 1** ■

(1) PROBE0 to PROBE3 are for tracing external signals.

(2) SYNC.OUT produces a pulse each time that the emulator executes the instruction at an address with its sync out bit set to “1.”

(3) EXT.BRK is an external break signal.

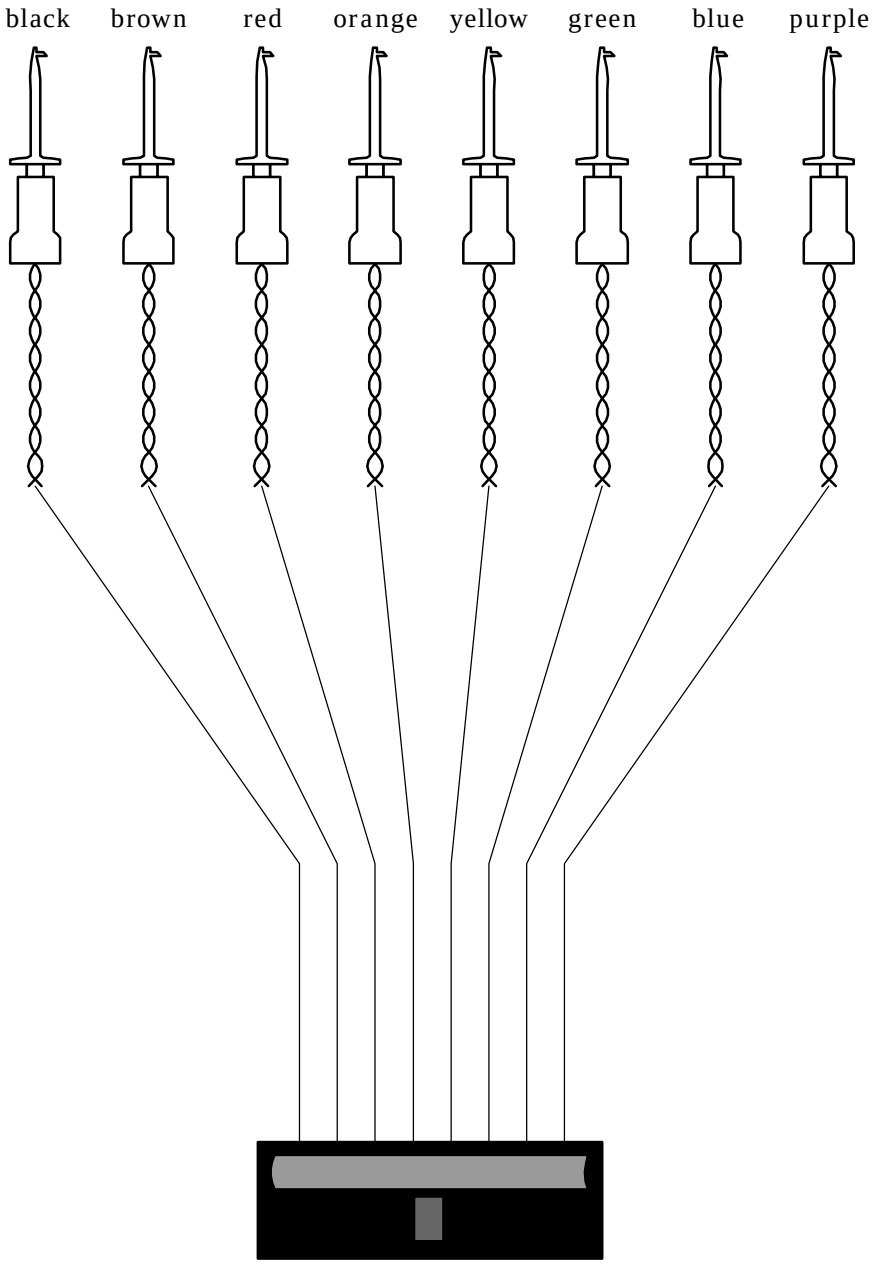
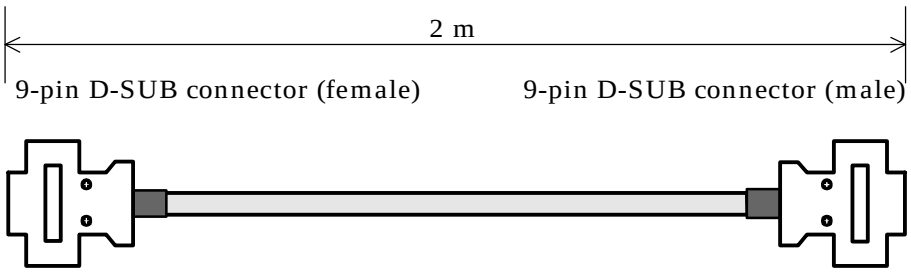


Figure A-4 Probe cable layout

5. RS232C Cable Wiring Diagrams

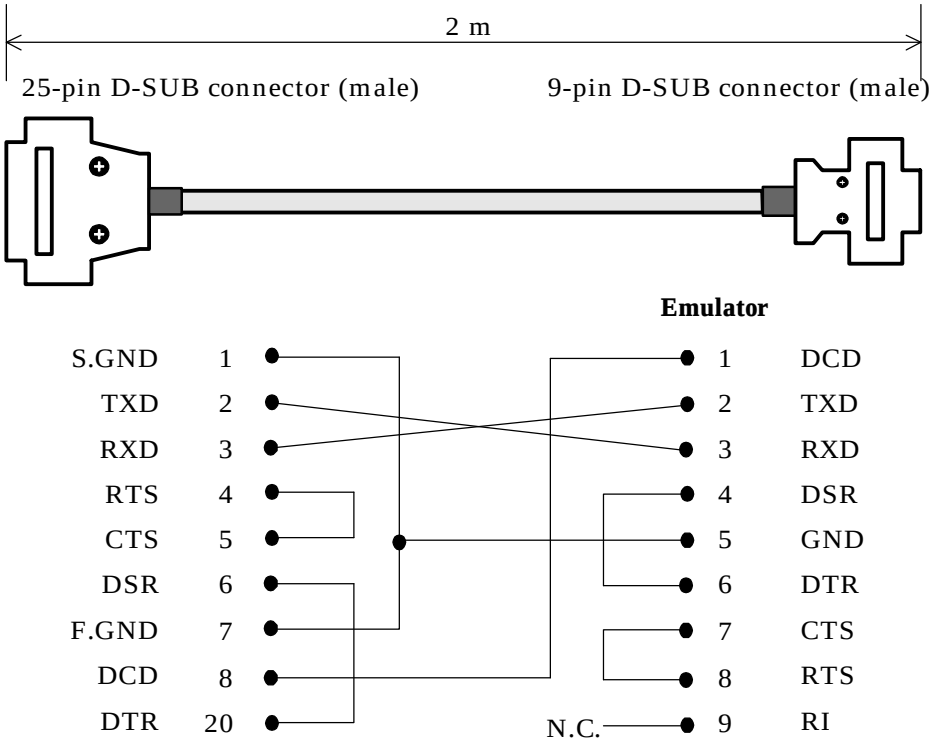
TCS-DRIBM (9-9 pin cable)



Emulator

DCD	1	●	1	DCD
RXD	2	●	2	TXD
TXD	3	●	3	RXD
DTR	4	●	4	DSR
GND	5	●	5	GND
DSR	6	●	6	DTR
RTS	7	●	7	CTS
CTS	8	●	8	RTS
RI	9	●	9	RI

TCS-DRPC (25-9 pin cable)



■ **Note 1** ■

All pins other than those listed above are not connected.

6. RS232C Interface Circuit

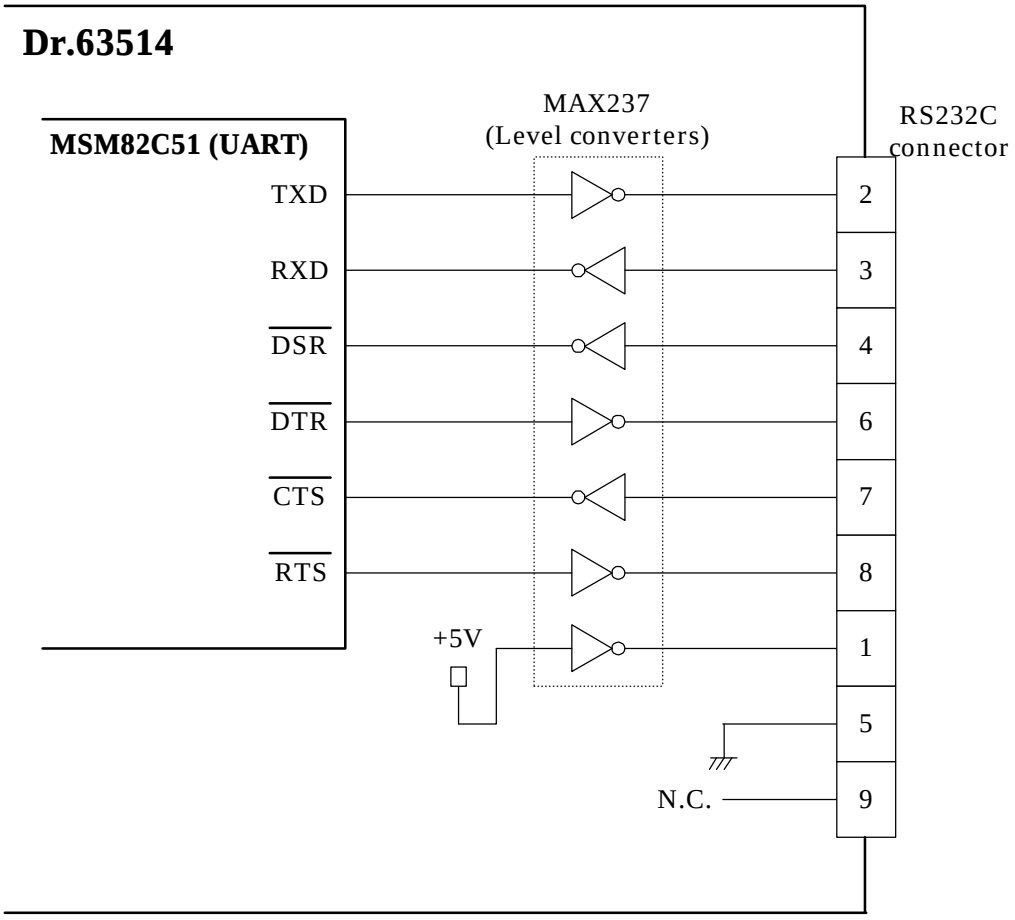


Figure A-5 RS232C interface circuit

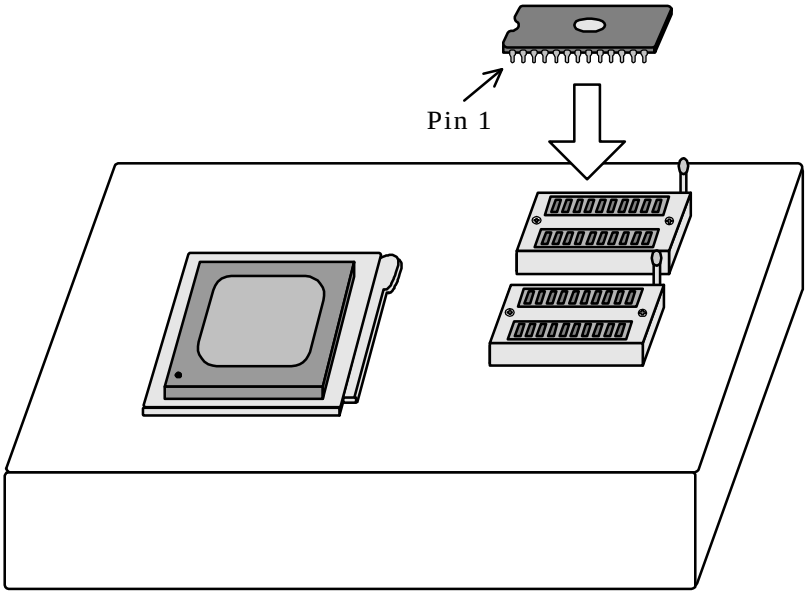
7. Installing EPROMs

There are two EPROM sockets on top of the emulator. Evaluation involves executing the user application program directly from EPROMs in them. Emulation offers commands for transferring EPROM contents to code memory.

Install an EPROM in its socket with the following procedure.

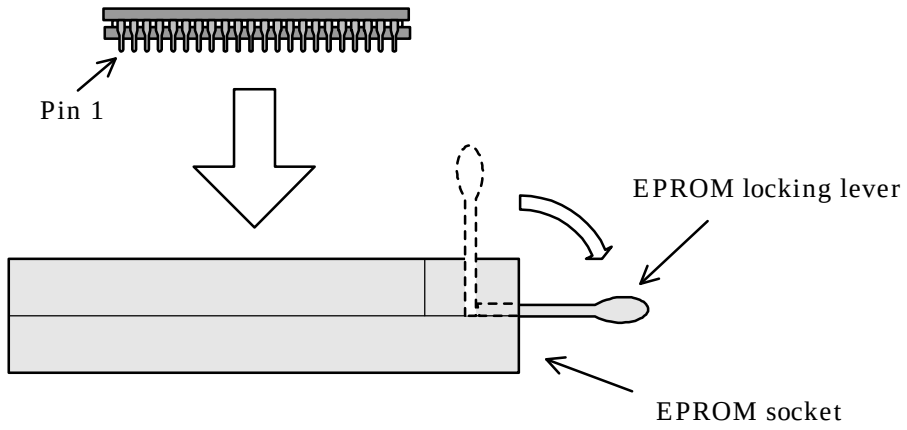
(1) Turn off the power to the emulator.

(2) Flip the lever beside the socket to its vertical position to unlock the socket.



(3) Fit the EPROM containing half of the user application program into the socket.

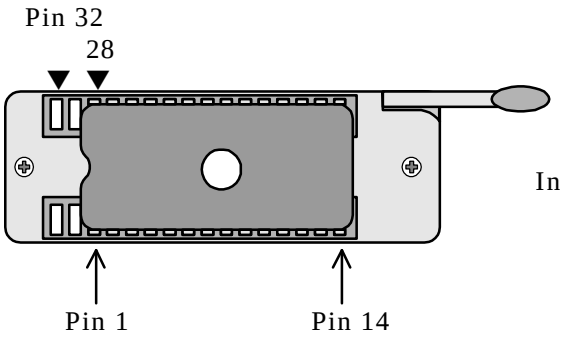
(4) Flip the locking lever to the side to lock the EPROM in place.



The sockets accept the following EPROM types. Note how the position of pin 1 differs for each type.

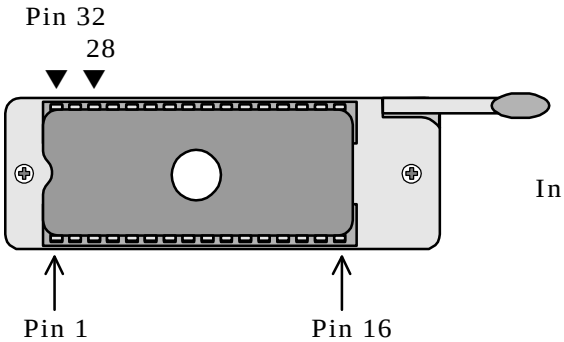
- MSM27512 and compatible devices (64K 8 bits; 28 pins)
- MSM27101 and compatible devices (128K 8 bits; 32 pins)

(1) MSM27512



Install with pin 14 all the way to the right.

(2) MSM27101



Install with pin 16 all the way to the right.

■ **Note 1** ■

Always be sure that the power is off before removing or installing an EPROM.

■ **Note 2** ■

A user application program always requires two EPROMs: one each in the EPROM.HIGH and EPROM.LOW sockets. See Chapter 2 Section 2.2 for such details as programming ranges.

8. If Emulator Doesn't Start

Symptom	Possible cause	Procedure
The ERROR LED lights.	The emulator is not operating properly.	Restart the emulator.
	The evaluation chip is not operating properly.	Check the IN/EXT setting of XT.SEL switch. Then restart the emulator.
	The serial link to the development host is not operating properly.	Make sure that the baud rate and other serial interface parameters of the development host match those of the emulator. Make sure that the cable specifications match those of the development host's serial port. Check the cable connections.
All LEDs other than the HALT display LED light.	The emulator's main control CPU has detected a system bus error due to noise, etc.	Restart the emulator.
The LEDs do not light. The debugger stalls after displaying its starting message.	The emulator is inoperative.	Make sure that the power supply cable is connected. Then restart the emulator.
In evaluation mode, the RUN LED turns off and real-time emulation is performed.	Because an N area break occurred.	Press the emulator reset switch.

If the Dr.63514 in-circuit emulator does not operate correctly even if above measures are taken, the Dr.63514 in-circuit emulator itself may have failed. Contact your Oki Electric Industry sales outlet or your Oki Electric Industry representative immediately.