Instruction Manual

Tektronix

TMS555
MPC565 Microcontroller Software Support

071-1015-00

www.tektronix.com

Tektronix, Inc., P.O. Box 500, Beaverton, OR 97077

SOFTWARE WARRANTY

Tektronix warrants that the media on which this software product is furnished and the encoding of the programs on the media will be free from defects in materials and workmanship for a period of three (3) months from the date of shipment. If a medium or encoding proves defective during the warranty period, Tektronix will provide a replacement in exchange for the defective medium. Except as to the media on which this software product is furnished, this software product is provided “as is” without warranty of any kind, either express or implied. Tektronix does not warrant that the functions contained in this software product will meet Customer’s requirements or that the operation of the programs will be uninterrupted or error-free.

In order to obtain service under this warranty, Customer must notify Tektronix of the defect before the expiration of the warranty period. If Tektronix is unable to provide a replacement that is free from defects in materials and workmanship within a reasonable time thereafter, Customer may terminate the license for this software product and return this software product and any associated materials for credit or refund.

THIS WARRANTY IS GIVEN BY TEKTRONIX IN LIEU OF ANY OTHER WARRANTIES, EXPRESS OR IMPLIED. TEKTRONIX AND ITS VENDORS DISCLAIM ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. TEKTRONIX’ RESPONSIBILITY TO REPLACE DEFECTIVE MEDIA OR REFUND CUSTOMER’S PAYMENT IS THE SOLE AND EXCLUSIVE REMEDY PROVIDED TO THE CUSTOMER FOR BREACH OF THIS WARRANTY. TEKTRONIX AND ITS VENDORS WILL NOT BE LIABLE FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES IRRESPECTIVE OF WHETHER TEKTRONIX OR THE VENDOR HAS ADVANCE NOTICE OF THE POSSIBILITY OF SUCH DAMAGES.

Table of Contents

Getting Started

Operating Basics

Reference

Specifications

Replaceable Parts List

Index

List of Figures

List of Tables

Preface

This instruction manual contains information specific to the TMS555 MPC565 microcontroller support package and is part of a set of information on how to operate this product on compatible Tektronix logic analyzers.

If you are familiar with operating microcontroller support packages on the logic analyzer for which the TMS555 MPC565 support was purchased, you will probably only need this instruction manual to set up and run the support.

If you are not familiar with operating microcontroller support packages, you will need to supplement this instruction manual with information on basic operations to set up and run the support.

Information on basic operations of microcontroller support packages is included with each product. Each logic analyzer includes basic information that describes how to perform tasks common to support packages on that platform. This information can be in the form of online help, an installation manual, or a user manual.

This manual provides detailed information on the following topics:

- Connecting the logic analyzer to the target system
- Setting up the logic analyzer to acquire data from the target system
- Acquiring and viewing disassembled data

Manual Conventions

This manual uses the following conventions:

- The term “disassembler” refers to the software that disassembles bus cycles into instruction mnemonics and cycle types.
- MPC565 also refers to MPC555 unless specifically stated.
- The phrase “information on basic operations” refers to logic analyzer online help, an installation manual, or a user manual covering the basic operations of the microcontroller support.
- The term “logic analyzer” refers to the Tektronix logic analyzer for which this product was purchased.

Contacting Tektronix

Phone	1-800-833-9200*
Address	Tektronix, Inc. Department or name (if known) 14200 SW Karl Braun Drive P.O. Box 500 Beaverton, OR 97077 USA
Web site	www.tektronix.com
Sales support	1-800-833-9200, select option 1*
Service support	1-800-833-9200, select option 2*
Technical support	Email: techsupport@tektronix.com 1-800-833-9200, select option 3* 1-503-627-2400 6:00 a.m. - 5:00 p.m. Pacific time

*** This phone number is toll free in North America. After office hours, please leave a voice mail message.**
Outside North America, contact a Tektronix sales office or distributor; see the Tektronix web site for a list of offices.

Getting Started

Getting Started

This section contains information on the TMS555 MPC565 microcontroller support, and information on connecting your logic analyzer to your target system.

Support Package Description

The TMS555 microcontroller support package displays disassembled data from systems based on the PowerPC MPC565 and MPC555 microcontrollers.

The support package is also expected to work for MPC561/562, MPC563/564, MPC566, and MPC556. The package does not support Code Compression.

Refer to information on basic operations to determine how many modules and probes your logic analyzer needs to meet the minimum channel requirements for the TMS555 microcontroller support.

To use this support efficiently, you need the items listed in the information on basic operations as well as the MPC555 User Manual, Motorola, Revision October 2000, and the MPC565 Reference Manual Revision, October 2000.

Logic Analyzer Software Compatibility

The label on the microcontroller support floppy disk states which version of logic analyzer software this support is compatible with.

Logic Analyzer Configuration

The TMS555 support requires a minimum of one 102-channel, 100 MHz acquisition module.

Requirements and Restrictions

Review the electrical specifications in the *Specifications* section in this manual as they pertain to your target system, as well as the following descriptions of other MPC565 support requirements and restrictions.

Hardware Reset. If a hardware reset occurs in your MPC565 system during an acquisition, the application disassembler might acquire invalid samples.

System Clock Rate. The support can acquire data from the MPC565 microcontroller operating at speeds of up to 56 MHz[1]. The MPC565 microcontroller support has been tested to 40 MHz for the nonburst mode and 20MHz for the burst mode.

Nonintrusive Acquisition. Acquiring microcontroller bus cycles is nonintrusive to the target system. That is, the TMS555 MPC565 microcontroller does not intercept, modify, or present back signals to the target system.

Channel Groups. Channel groups required for clocking and disassembly are the addr Group, Data Group, Control Group, Chip_Select Group, Byte_Enable Group, and ITR Group. The Misc group is not required for clocking and disassembly.

The sample that was available for testing showed behavior inconsistent from those described in the MPC565 device reference manual. The following restrictions arise from these observations.

Refetching in Burst Mode. The processor refetches instructions while operating in burst mode, even for noncontrol flow instructions (sequential instructions). This mostly happens with multiple Read/Write instructions and floating point instructions, but apparently is not related to the type of instructions only. Since the disassembler has no knowledge or any indication of these refetches that are made for normal sequential instructions, the instructions are parsed and displayed multiple times.

No Flush Information on the VF and VFLS Signals. The VF and VFLS lines do not give the Flush Information as described in the device manual. This does not affect the disassembly. Device behavior as described in the manual is untested; the disassembly in Memory Image mode may be wrong if the behavior is different.

Unexpected Messages on the VF Pins. If Data Show cycles are enabled, then the messages “Indirect branch taken” and “VSYNC asserted” are displayed for the VF signals even when there are no corresponding instructions. In Memory Image mode, disassembly may be wrong because of these unexpected messages. This problem does not occur if Data Show Cycles are disabled. If this problem occurs, you cannot enable data show cycle while using Memory Image mode.

[1] Specification at time of printing. Contact your Tektronix sales representative for current information on the fastest devices supported.

Timing Display Format

The support has a Timing Display Format file. It sets up the display to show the following waveforms:

CLKOUT
addr
Data
RD/WR~
BURST~
TS~
BDIP~
TA~
BI~/STS~
AT2
BR~
BG~
BB~
TEA~
OE~
RETRY~
Byte_Enable

***NOTE**. The addr, Data and Byte_Enable are displayed in bus form.*

Functionality Not Supported

Interrupt Signals. Not all of the interrupt signals are acquired by the TMS555 support software. Such signals are identified by the TMS555 support software which displays the address for the interrupt service.

Alternate Bus Master. The disassembly does not process alternate bus master transactions.

Code Compression. The package does not support code compression feature of MPC556, MPC566, MPC562, and MPC564.

Functionality Supported But Not Tested

The following features are supported, but are not tested:

- 8 and 16 bit port sizes
- Little Endian Mode
- Alternate Bus master
- Address Offset feature in Memory Image Mode

This support package may work for MPC561/562, and MPC563/564 microcontrollers, though it has not been tested.

Limitations Of The Support

The support has the following limitations when consecutive/ multireads and/or consecutive branches are encountered. These limitations can be overcome by using the Marking Options provided.

- Consecutive reads and multireads may be identified as fetches if AT[2] (read/write indicator) is not enabled.
- The branch target embedded within the reads may be identified as READ. This is expected only for Normal nonburst mode.
- Two branches within the instruction queue length may not be identified properly. That is, the instructions are parsed, but the indication, whether taken or not taken, may not be shown.
- The support identifies as not taken a conditional/indirect branch address that is reached sequentially. For example, consider that there is a branch from external to internal memory. Now the instructions are executed in the internal memory. Consider that there is another branch in the internal memory that branches back to the next sequential instruction of the external memory. The support then identifies the branch of external memory as not taken.

Connecting the Logic Analyzer to a Target System

You can use the channel probes, clock probes, and leadsets with a commercial test clip (or adapter) to make the connections between the logic analyzer and your target system.

To connect the probes to MPC565 signals in the target system using a test clip, follow the steps:

1. Power off your target system. You do not need to power off the logic analyzer.

CAUTION. *To prevent static damage, handle the microprocessor, probes, and the logic analyzer module in a static-free environment. Static discharge can damage these components.*

Always wear a grounding wrist strap, heel strap, or similar device while handling the microprocessor.

2. To discharge your stored static electricity, touch the ground connector located on the back of the logic analyzer. If you are using a test clip, touch any of the ground pins on the clip to discharge stored electricity from the test clip.

CAUTION. *To prevent permanent damage to the pins on the microprocessor, place the target system on a horizontal surface before connecting the test clip.*

3. Place the target system on a horizontal, static-free surface.
4. Use Tables 3–13 through 3–15 starting on page 3–7 to connect the channel probes to MPC565 signal pins on the test clip or in the target system.
5. Use leadsets to connect at least one ground lead from each channel and the ground lead from each clock probe to the ground pins on your test clip.

Labeling P6434 Probes

The TMS555 MPC565 software support package relies on the channel mapping and labeling scheme for the P6434 Probes. Apply labels using the instructions described in the P6434 Probe Instructions manual.

Operating Basics

Setting Up the Support

This section provides information on how to set up the support and covers the following topics:

- Clocking options
- Timing diagram

The information in this section is specific to the operations and functions of the TMS555 MPC565 support on any Tektronix logic analyzer for which the support can be purchased. Information on basic operations describes general tasks and functions.

Before you acquire and display disassembled data, you need to load the support and specify the setups for clocking and triggering as described in the information on basic operations. The support provides default values for each of these setups, but you can change the values as needed.

Installing the Support Software

***NOTE**. Before you install any software, you should verify that the microcontroller support software is compatible with the logic analyzer software.*

To install the TMS555 MPC565 software on your Tektronix logic analyzer, follow these steps:

1. Insert the floppy disk in the disk drive.
2. Click the Windows Start button, point to Settings, and click Control Panel.
3. In the Control Panel window, double-click Add/Remove Programs.
4. Follow the instructions on the screen for installing the software from the floppy disk.

To remove or uninstall software, follow the above instructions and select Uninstall. You need to close all windows before you uninstall any software.

Channel Group Definitions

The software automatically defines channel groups for the support. The channel groups for the MPC565 support are addr, Data, Control, Byte_Enable, Chip_Select, ITR and Misc. If you want to know which signal is in which group, refer to the channel assignment tables beginning on page 3–2.

Clocking

Acquisition Setup

The TMS555 MPC565 affects the logic analyzer setup menus (and submenus) by modifying existing fields and adding micro-specific fields.

On the logic analyzer, the TMS555 MPC565 adds the selection “MPC565” to the Load Support Package dialog box, under the File pulldown menu. Once “TMS555 MPC565 support” has been loaded, the “Custom” clocking mode selection in the logic analyzer module Setup menu is also enabled.

Clocking Options

The TMS555 support offers a microcontroller-specific clocking mode for the MPC565 microcontroller. This clocking mode is the default selection whenever you load the MPC565 support.

Disassembly will not be correct when using the Internal or External clocking modes. Information on basic operations describes how to use these clock selections for general purpose analysis.

Custom Clocking

A special clocking program is loaded to the module every time you load the MPC565 support. This special clocking is called Custom.

In this support with custom clocking, the module logs in all the signals at every rising edge of the clock signal (CLKOUT). The module then sends all the logged-in signals to the trigger machine and to the acquisition memory of the module for storage. That is, in custom clocking, the clocking state machine (CSM) generates one master sample at every rising edge of the clock.

When Custom is selected, the Custom Clocking Options menu adds the subtitle “MPC565 Microcontroller Clocking Support”, and displays the clocking option —External Bus Interface. This is the only custom clocking option available for this support.

Bus Timing Diagram. All the signals are acquired and mastered at every rising edge of the signal CLKOUT. A basic timing diagram for burst mode is given in Figure 2-1.

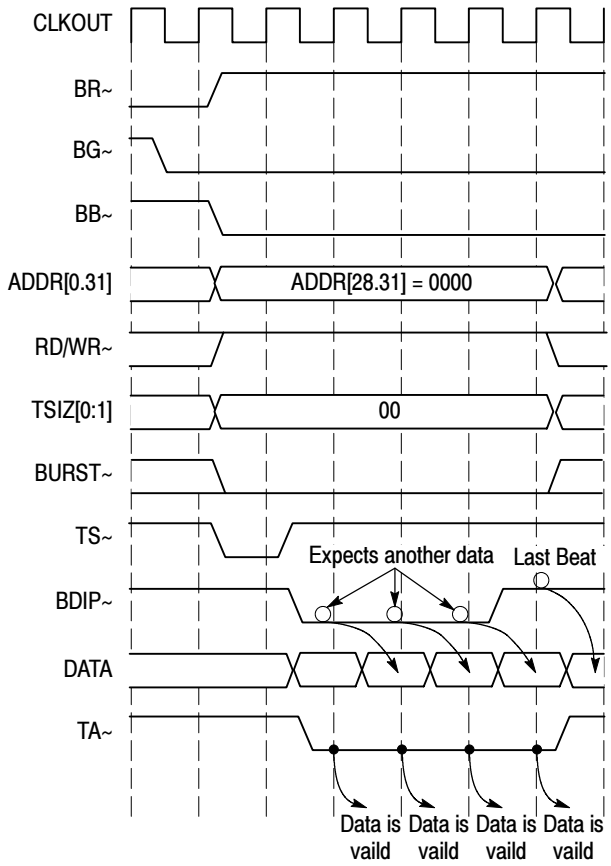


Figure 2-1: Bus timing diagram

Acquiring and Viewing Disassembled Data

This section describes how to acquire data and view it disassembled. The following information covers these topics and tasks:

- Acquiring data
- Viewing disassembled data in various display formats
- Viewing cycle-type labels
- Changing the way data is displayed
- Changing disassembled cycles with the mark cycles function

Acquiring Data

Once you load the MPC565 support by default, the custom clocking option is selected. Specify the trigger, if any, and you are ready to acquire and disassemble data.

If you have any problems acquiring data, refer to information on basic operations in your online help or *Appendix A: Error Messages and Disassembly Problems* in the user manual.

Viewing Disassembled Data

You can view disassembled data in four display formats: Hardware, Software, Control Flow, and Subroutine. The information on basic operations describes how to select the disassembly display formats.

NOTE*. You must set the selections in the Disassembly property page (the Disassembly Format Definition overlay) correctly for your acquired data to be disassembled correctly. Refer to* Changing How Data is Displayed *on page 2-9.*

The default display format shows the addr, Data, Mnemonics, Control, Chip_Select, and Byte_Enable channel group values for each sample of acquired data.

If a channel group is not visible, you can add the required column by pressing Ctrl + L and selecting the group of interest.

The disassembler displays special characters and strings in the instruction mnemonics to indicate significant events. Table 2–1 shows these special characters and strings, and describes what they represent.

Table 2–1: Description of special characters in the Listing display

Character or string displayed	Description
>>	The instruction was manually marked.
0x	Indicates an immediate hexadecimal value.
>	Indicates insufficient room on the screen to show all available data in any column.
ILLEGAL INSTRUCTION	Illegal instruction
INSUFFICIENT DATA	Indicates there is insufficient data available for complete disassembly of the instruction.

Hardware Display Format

In Hardware display format, the disassembler displays certain cycle-type labels in parentheses.

In Hardware display format, all valid opcode fetch bus cycles will be disassembled and displayed. Noninstruction bus cycles will be displayed with the appropriate Cycle Type label. There is no attempt to link operand reads and writes with the instructions, that cause them. This is the default format for disassembly.

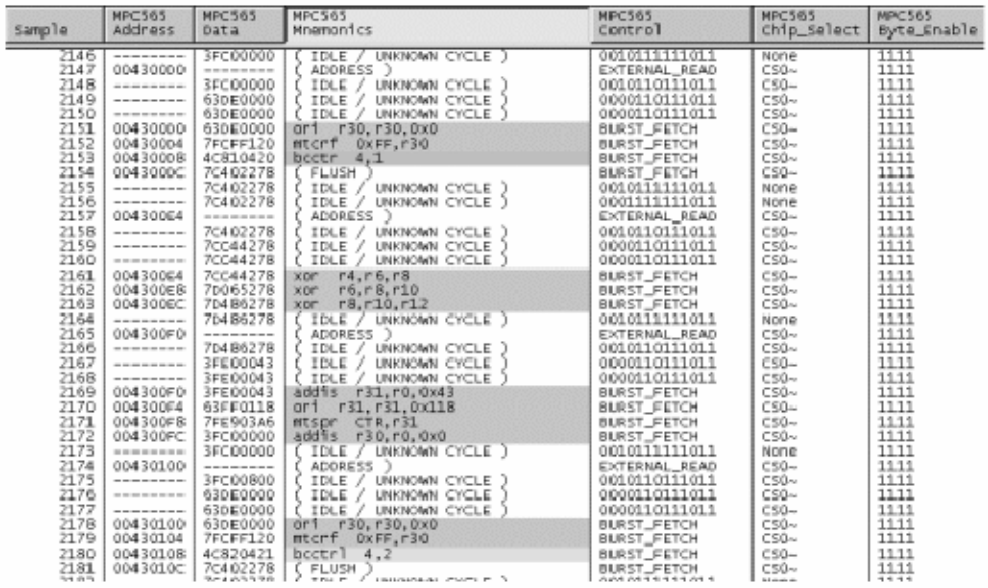
Table 2–2: General cycle type label definitions

Cycle type	Definition
(READ)	Read cycle
(WRITE)	Write cycle
(ADDRESS)	Address
(FLUSH)	This cycle is fetched but not executed
(EXTENSION)	This cycle is an extension to a preceding instruction opcode
(TRANSFER ERROR)	Bus transfer error
(ALTERNATE MASTER CYCLE)	Alternate master cycle
(READ DATA SHOW CYCLE - ADDRESS)	Read Data Show Cycle - Address
(READ DATA SHOW CYCLE - DATA)	Read Data Show Cycle - Data
(WRITE DATA SHOW CYCLE - ADDRESS)	Write Data Show Cycle - Address
(WRITE DATA SHOW CYCLE - DATA)	Write Data Show Cycle - Data
(BURST READ DATA SHOW CYCLE - ADDRESS)	Burst Data Show Cycle - Address

Table 2-2: General cycle type label definitions (cont.)

Cycle type	Definition
(BURST READ DATA SHOW CYCLE - DATA)	Burst Data Show Cycle - Data
(RETRY CYCLE)	Retry cycle
(IDLE/UNKNOWN CYCLE)	Idle/unexpected/unrecognized combination of control signals

Figure 2-2 shows an example of the Hardware Display format.



Sample	MPC565 Address	MPC565 Data	MPC565 Mnemonics	MPC565 Control	MPC565 Chip_Select	MPC565 Byte_Enable
2146	---------	3FC00000	(IDLE / UNKNOWN CYCLE)	00101111111011	None	1111
2147	00430000	---------	(ADDRESS)	EXTERNAL_READ	CS0~	1111
2148	---------	3FC00000	(IDLE / UNKNOWN CYCLE)	00101101111011	CS0~	1111
2149	---------	630E0000	(IDLE / UNKNOWN CYCLE)	00000110111011	CS0~	1111
2150	---------	630E0000	(IDLE / UNKNOWN CYCLE)	00000110111011	CS0~	1111
2151	00430000	630E0000	ori r30,r30,0x0	BURST_FETCH	CS0~	1111
2152	00430004	7FCFF120	mtcrf 0xFF,r30	BURST_FETCH	CS0~	1111
2153	00430008	4C810420	bcctr 4,1	BURST_FETCH	CS0~	1111
2154	0043000C	7C402278	(FLUSH)	BURST_FETCH	CS0~	1111
2155	---------	7C402278	(IDLE / UNKNOWN CYCLE)	00101111111011	None	1111
2156	---------	7C402278	(IDLE / UNKNOWN CYCLE)	00011111111011	None	1111
2157	00430004	---------	(ADDRESS)	EXTERNAL_READ	CS0~	1111
2158	---------	7C402278	(IDLE / UNKNOWN CYCLE)	00101101111011	CS0~	1111
2159	---------	7CC44278	(IDLE / UNKNOWN CYCLE)	00000110111011	CS0~	1111
2160	---------	7CC44278	(IDLE / UNKNOWN CYCLE)	00000110111011	CS0~	1111
2161	00430004	7CC44278	xor r4,r6,r8	BURST_FETCH	CS0~	1111
2162	00430008	7D065278	xor r6,r8,r10	BURST_FETCH	CS0~	1111
2163	0043000C	7D4B6278	xor r8,r10,r12	BURST_FETCH	CS0~	1111
2164	---------	7D4B6278	(IDLE / UNKNOWN CYCLE)	00101111111011	None	1111
2165	00430010	---------	(ADDRESS)	EXTERNAL_READ	CS0~	1111
2166	---------	7D4B6278	(IDLE / UNKNOWN CYCLE)	00101101111011	CS0~	1111
2167	---------	3FE00043	(IDLE / UNKNOWN CYCLE)	00000110111011	CS0~	1111
2168	---------	3FE00043	(IDLE / UNKNOWN CYCLE)	00000110111011	CS0~	1111
2169	004300F0	3FE00043	addis r31,r0,0x43	BURST_FETCH	CS0~	1111
2170	004300F4	63FF0118	ori r31,r31,0x118	BURST_FETCH	CS0~	1111
2171	004300F8	7FE903A6	mtspr CTR,r31	BURST_FETCH	CS0~	1111
2172	004300FC	3FC00000	addis r30,r0,0x0	BURST_FETCH	CS0~	1111
2173	========	3FC00000	(IDLE / UNKNOWN CYCLE)	00101111111011	None	1111
2174	00430100	---------	(ADDRESS)	EXTERNAL_READ	CS0~	1111
2175	---------	3FC00800	(IDLE / UNKNOWN CYCLE)	00101101111011	CS0~	1111
2176	---------	63DE0000	(IDLE / UNKNOWN CYCLE)	00000110111011	CS0~	1111
2177	---------	63DE0000	(IDLE / UNKNOWN CYCLE)	00000110111011	CS0~	1111
2178	00430100	63DE0000	ori r30,r30,0x0	BURST_FETCH	CS0~	1111
2179	00430104	7FCFF120	mtcrf 0xFF,r30	BURST_FETCH	CS0~	1111
2180	00430108	4C820421	bcctrl 4,2	BURST_FETCH	CS0~	1111
2181	0043010C	7C402278	(FLUSH)	BURST_FETCH	CS0~	1111

Figure 2-2: Example of the Hardware Display format

Software Display Format

The Software display format shows only the first opcode fetch of executed instructions. Flushed cycles and extensions are not shown, even though they are part of the executed instruction. Data reads and writes are not displayed.

Any “special” cycles that are described as displayed in Control Flow or Subroutine display formats are also displayed here.

Control Flow Display Format

The Control Flow display format shows only the first opcode fetch of instructions that cause a branch in the addressing.

The support always displays unconditional branches acquired. The support will not identify whether a branch is taken if a conditional branch branches to an address that is reached sequentially.

These instructions generate an unconditional change in the control flow in the MPC565 microcontroller:

b
ba
mtmsr
mtspr (using this instruction with some special purpose registers will effect the machine in a manner similar to indirect branches.)

These instructions might generate a conditional change in the control flow in the MPC565 microcontroller:

bc
bca
bcctr
bclr

This format also displays any “special” cycles displayed in the Subroutine display format.

Subroutine Display Format

The Subroutine display format shows only the first opcode fetch of subroutine call and return instructions. The subroutine display shows conditional subroutine calls if they are considered to be taken.

Thses instructions generate an unconditional subroutine call or a return in the MPC565 microcontroller:

sc
rfi
isync
bl
bla

These instructions might generate a conditional subroutine call or a return in the MPC565 microcontroller:

tw
twi
bcl
bcla
bcctrl
bclrl

The disassembler displays some instructions that cause traps or interrupts, as well as exception vector reads. Vector reads may be interpreted as exceptions taken.

Changing How Data is Displayed

There are common fields and features that allow you to further modify displayed data to suit your needs. You can make common and optional display selections in the Disassembly property page (the Disassembly Format Definition overlay).

You can make selections unique to the MPC565 support to do the following tasks:

- Change how data is displayed across all display formats
- Change the interpretation of disassembled cycles
- Display exception cycles

Optional Display Selections

You can make optional selections for disassembled data. In addition to the common selections (described in the information on basic operations), you can change the displayed data as shown in Table 2–3:

Table 2–3: Logic analyzer disassembly display options

Description	Option	
Show:	Hardware Software Control Flow Subroutine	(Default)
Highlight:	Software Control Flow Subroutine None	(Default)
Disassemble Across Gaps:	Yes No	 (Default)

Micro-Specific Fields

The following micro-specific fields are available in the Disassembly options page.

Memory Image Status. You can choose one of the two available options to select the Memory Image status. When you choose the Enabled option, you cannot edit or modify the S-Record (Image File currently in use). You must choose the Disabled option to edit or modify the S-record. This field is applicable only for Memory Image mode.

Memory Image Status: Enabled (default)
Disabled

Disassemble Based On. You can choose one of the two options to select the basis for disassembly. If you choose the option Fetch Stream, normal disassembly occurs. If you select the Memory Image option, disassembly is based on the image file. Select one of the two available options.

Disassemble Based On: Fetch Stream (default)
Memory Image

Image File Path. You must enter the complete path to the S-record file in the property for Image file path. Use the Browse button for this. By default, this field is blank. The file path must be entered if the "Memory Image" option is chosen. This field is applicable only for Memory Image mode.

Address Offset in Hex. This is the address offset (in hexadecimal) from the starting address (as indicated by the S-record) where the user program is loaded in memory. By default this is 0x00000000.

For example, the linker output and the corresponding S-record file have a starting address of 0x0, but you want to load these at a different address. You can load the starting address at 0x50 and then specify the 0x50 offset as 0xFFFFFFB0 in the Address Offset in Hex field.

- When the S-record address is less than the Processor_Address, then the Address_Offset must be negative.
- When the S-record address is greater than the Processor_Address, then the Address_Offset must be positive.

So the correspondence intended is:

Processor_Address + Address_Offset == S_Record_Address

This field is applicable only for Memory Image mode.

Arbitrer. You must select one of the available options based on your system configuration. Alternate master cycles are identified depending upon this selection. This field is applicable only for Fetch Stream mode.

Arbitrer: Single Processor (default)
Internal
External

Suppress Sequences. You can suppress or display idle cycles in the Hardware Display format by selecting one of the two available options. This field is applicable only for Fetch Stream mode.

Suppress Sequences: No (default)
Yes

AT2 Activated. You can activate AT[2] by selecting one of the two available options. If IRQ[4]/AT[2]/SGPIOC[4] pin is programmed as AT[2], then Read/Fetch indication is done using AT[2] or else Read/Fetch identification is done by heuristics. This field is applicable only for Fetch Stream mode.

AT2 Activated: Yes (default)
No

RETRY~ Activated. If you select Yes and the IRQ[3]/KR/RETRY/SGPIOC[3] is programmed as RETRY, then the Retry cycles are displayed. This field is applicable only for Fetch Stream mode.

RETRY~ Activated: Yes (default)
No

WE~ / BE~ /AT used as. Select the appropriate option, depending on whether you are using WE[0:3]/BE[0:3]/AT[0:3] signals as write enables or byte enables. Invalid data will be dashed out for Write cycles, or for both Read as well as Write cycles, depending on the selected option. This field is applicable only for Fetch Stream mode.

WE~/BE~/AT used as: WE[0:3]~(default)
BE[0:3]~

BI~/STS~ used as. If you choose the STS~ option, then the show cycles are identified and displayed in Fetch stream mode. This field is applicable only for Fetch Stream mode. Set this by selecting one of the available options.

BI~/STS~ used as: BI~(default)
STS~

Port Size. The pakcage supports three Port sizes. This field is applicable only for Fetch Stream mode. Select one of the three available options depending upon the port size.

Port Size: 32-bit (default)
16-bit
8-bit

Endian Mode. Select one of the two available options for Endian mode.

Endian Mode: Big Endian (default)
Little Endian

Burst Length. Based on the burst length setting on your system, the burst length for MPC561/562 and MPC563/564 can be 4 or 8. For MPC565/566 and MPC555/556 the default value of burst length is 4. This field is applicable only for Fetch Stream mode. Select the burst length by selecting one of the following options.

Burst Length: 4 (default)
8

Vector Table Base Address. Type in the Vector Table Base Address in the field. The default address is 00000000.

Chip Select CS0. Enter the Base Address to get the 32-bit full address display in the listing. The default address is 00000000.

Chip Select CS1. Enter the Base Address to get the 32-bit full address display in the listing. The default address is 00000000.

Chip Select CS2. Enter the Base Address to get the 32-bit full address display in the listing. The default address is 00000000.

Chip Select CS3. Enter the Base Address to get the 32-bit full address display in the listing. The default address is 00000000.

Marking Cycles

The disassembler has a Mark Opcode function that allows you to change the interpretation of a cycle type. Using this function, you can select a cycle and change it.

Logic Analyzer. Use the Mark Opcode options to place Marks. The Mark Opcode button is always available. If the marked sample is not an Address cycle or Data cycle of the potential bus master, the disassembler replaces the Mark Opcode selections by a note indicating that “An Opcode Mark cannot be placed at the selected data sample.”

When you mark a cycle, the character “>>” is displayed immediately to the left of the Mnemonics column. You can unmark cycles by using the “Undo Mark” selection, which removes the character “>>”.

The list of selections varies depending on the selection in the Bus Processor Select field in the Disassembly property page (Disassembly Format Definition overlay).

Table 2–4 describes the various combinations of mark selections.

Table 2–4: Mark selections and definitions

Mark selection	Definition
Opcode	Mark cycle as an instruction opcode
Read	Mark cycle as a Read cycle
Flush	Mark cycle as a flushed cycle
Undo Mark	Remove all marks from the current sample

***NOTE**. If the Read/Fetch indicator (AT[2]) is enabled, then only Flush and Undo Mark marking selections are available.*

Information on basic operations contains more details on marking cycles.

Displaying Exception Labels

The disassembler can display MPC565 exception labels. The exception table must reside in external memory for interrupt and exception cycles to be visible to the disassembler.

You can enter the table prefix in the Exception Prefix field. The Exception Prefix field provides the disassembler with the offset address; enter the Vector Base Address.

These fields are located in the Disassembly property page (Disassembly Format Definition overlay).

Table 2–5 lists the MPC565 interrupt and exception labels.

Table 2–5: Interrupt and exception labels

Offset	Displayed interrupt or exception name
0x00000	(RESERVED)
0x00100	(SYSTEM RESET)
0x00200	(MACHINE CHECK)
0x00300	(RESERVED)
0x00400	(RESERVED)
0x00500	(EXTERNAL INTERRUPT)
0x00600	(ALIGNMENT)

Table 2-5: Interrupt and exception labels (cont.)

Offset	Displayed interrupt or exception name
0x00700	(PROGRAM)
0x00800	(FLOATING-POINT UNAVAILABLE)
0x00900	(DECREMENTER)
0x00A00	(RESERVED)
0x00B00	(RESERVED)
0x00C00	(SYSTEM CALL)
0x00D00	(TRACE)
0x00E00	(FLOATING POINT ASSIST)
0x01000	(IMPLEMENTATION DEPENDENT SOFTWARE EMULATION)
0x01100	(RESERVED)
0x01200	(RESERVED)
0x01300	(IMPLEMENTATION DEPENDENT INSTRUCTION PROTECTION ERROR)
0x01C00	(IMPLEMENTATION DEPENDENT DATA BREAKPOINT)
0x01D00	(IMPLEMENTATION DEPENDENT INSTRUCTION BREAKPOINT)
0x01E00	(IMPLEMENTATION DEPENDENT MASKABLE EXTERNAL BREAKPOINT)
0x01F00	(NON-MASKABLE EXTERNAL BREAKPOINT)

Viewing an Example of Disassembled Data

A demonstration system file (or demonstration reference memory) is provided on your software disk to show an example of how your MPC565 microcontroller bus cycles and instruction mnemonics look when they are disassembled. Viewing the system file is not a requirement for preparing the module for use and you can view it without connecting the logic analyzer to your target system.

Information on basic operations describes how to view the file.

Instruction Trace Reconstruction (ITR)

The logic analyzer acquires data, which appears on the external bus of the microprocessor. When code is being executed from the internal memory, there is no external bus activity. This severely limits the information that a logic analyzer can display. To address this problem, some indirect methods are used to logically

track the program flow, even though instruction fetches are happening from internal memory. A brief explanation follows with examples showing ways that you can use the ITR method with this support.

It is possible to reconstruct the program execution. That is, the portions of the program, which get executed form the internal memory, can be read from the Image file and shown on the display. This can occur if an Image File of the program that is being executed is available in S-record format, and if the processor provides information about the control flow instructions being executed and if that information can be acquired by the logic analyzer.

Memory Image S-Record

The memory image is a hexadecimal form of the program being executed by the processor. The memory image is the output of the Compiler/Assembler and Linker. Linker output is normally available in one of the industry standard formats like Intel Hex format, S-record format, or a proprietary format used by the software development system. This support requires the external image file to be in the Motorola S-record format. Usually, tools are available to convert proprietary output formats into Motorola S-record. You can use GNU tools to convert a source file into an S-record file (Image file).

Image Reader

The Motorola MPC565 microcontroller provides VF and VFLS signals. Instruction queue status pins (VF) show the type of the last fetched instruction or how many instructions were flushed from the queue. Because of an exception during the clock, the history buffer flushes status pins (VFLS) to show how many instructions are flushed from the buffer. You can do a program trace using the information available on these pins along with the Show cycles when they are enabled only for indirect branches.

The TMS555 package supports only the S-record format. Therefore the support requires that the Image File be available in Motorola S-record format. At least one taken indirect branch instruction is expected in 700 instructions for the support to show proper disassembly.

Viewing Internal Memory Activity

This procedure (for converting a source file into an S-record file) uses GNU tools. If you do not have this software, you must find an alternative. Contact your Tektronix sales representative if you need support.

This section on viewing the Internal Memory activity on the Tektronix logic analyzer consists of a three-step procedure:

- Retrieving Control Flow information
- Generating an S-record file (Image file)
- Configuring the Logic Analyzer

Retrieving Control Flow Information. Follow this procedure to retrieve information about the Control Flow from the processor.

1. Enable the VF[0:3] and VFLS[0:2] signals by setting the VF and VFLS fields in MIOS1TPCR register.
2. Enable the Show Cycles for Indirect Branches by setting the ISCT_SER field to the appropriate value in the ICTRL register.
3. Program the pin BI~/STS~ as STS~ by setting the appropriate value in DBGC field of SIUMCR register.

NOTE. *For more information about the registers and the values to be set in the respective fields, refer to the device vendor manual.*

Generating an S-Record File. The source code must be converted into an S-record format. For example, the following steps produce an S-record file from a source file using GNU Compiler for PowerPC.

NOTE. *The file naming conventions followed by the GNU Compiler are:*

- A source file has an extension '.s'
- An object file has an extension '.o'
- An elf file, for example the output of the linker, has the extension '.elf'
- The Motorola S-records have an extension '.src'

At the command prompt, do the following steps.

1. Create the object file (.o) using the following command:

   as -o objectfile.o source.s

2. Create the elf file and the S-record format file, using the linker command:

   ld objectfile.o --oformat srec -o srecord.src

NOTE. *If you are using the GNU Compiler for PowerPC, refer to the respective documentation for further details about the commands.*

Configuring the Logic Analyzer. Follow these steps to configure your logic analyzer.

1. Load the support package.
2. Modify the properties in the property page of the logic analyzer as shown in Figure 2–3.

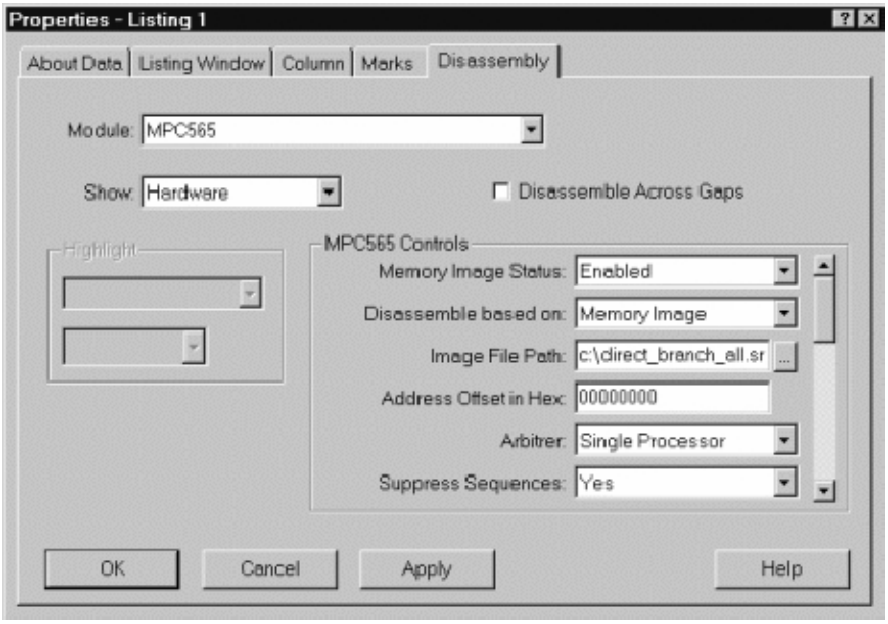
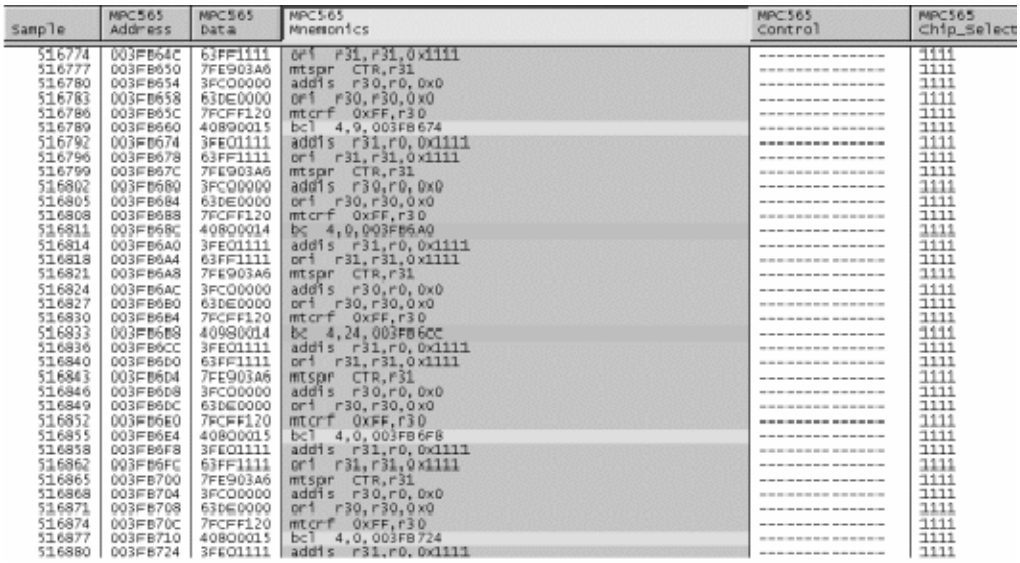


Figure 2–3: Example of Properties Page display

3. Enter the complete path to the S-Record file/Image file in the property Image file path. You can do this either manually or by using the menu button to the right of the property for Image file path which opens a “Browse” window.
4. Once the settings are done, select OK/Apply to view the data on the display. To revert to the original Fetch Stream data, change the value of the property “Disassemble based on” to “Fetch Stream”.

Figure 2–4 displays a sample screen for ITR display.



Sample	MPC565 Address	MPC565 Data	MPC565 Mnemonics	MPC565 Control	MPC565 Chip_Select
516774	003FB64C	63FF1111	ori r31,r31,0x1111	----------	1111
516777	003FB650	7FE903A6	mtspr CTR,r31	----------	1111
516780	003FB654	3FC00000	addis r30,r0,0x0	----------	1111
516783	003FB658	63DE0000	ori r30,r30,0x0	----------	1111
516786	003FB65C	7FCFF120	mtcrf 0xFF,r30	----------	1111
516789	003FB660	40890015	bcl 4,9,003FB674	----------	1111
516792	003FB674	3FE01111	addis r31,r0,0x1111	----------	1111
516796	003FB678	63FF1111	ori r31,r31,0x1111	----------	1111
516799	003FB67C	7FE903A6	mtspr CTR,r31	----------	1111
516802	003FB680	3FC00000	addis r30,r0,0x0	----------	1111
516805	003FB684	63DE0000	ori r30,r30,0x0	----------	1111
516808	003FB688	7FCFF120	mtcrf 0xFF,r30	----------	1111
516811	003FB68C	40820014	bc 4,0,003FB6A0	----------	1111
516814	003FB6A0	3FE01111	addis r31,r0,0x1111	----------	1111
516818	003FB6A4	63FF1111	ori r31,r31,0x1111	----------	1111
516821	003FB6A8	7FE903A6	mtspr CTR,r31	----------	1111
516824	003FB6AC	3FC00000	addis r30,r0,0x0	----------	1111
516827	003FB6B0	63DE0000	ori r30,r30,0x0	----------	1111
516830	003FB6B4	7FCFF120	mtcrf 0xFF,r30	----------	1111
516833	003FB6B8	40980014	bc 4,24,003FB6CC	----------	1111
516836	003FB6CC	3FE01111	addis r31,r0,0x1111	----------	1111
516840	003FB6D0	63FF1111	ori r31,r31,0x1111	----------	1111
516843	003FB6D4	7FE903A6	mtspr CTR,r31	----------	1111
516846	003FB6D8	3FC00000	addis r30,r0,0x0	----------	1111
516849	003FB6DC	63DE0000	ori r30,r30,0x0	----------	1111
516852	003FB6E0	7FCFF120	mtcrf 0xFF,r30	----------	1111
516855	003FB6E4	40820015	bcl 4,0,003FB6F8	----------	1111
516858	003FB6F8	3FE01111	addis r31,r0,0x1111	----------	1111
516862	003FB6FC	63FF1111	ori r31,r31,0x1111	----------	1111
516865	003FB700	7FE903A6	mtspr CTR,r31	----------	1111
516868	003FB704	3FC00000	addis r30,r0,0x0	----------	1111
516871	003FB708	63DE0000	ori r30,r30,0x0	----------	1111
516874	003FB70C	7FCFF120	mtcrf 0xFF,r30	----------	1111
516877	003FB710	40820015	bcl 4,0,003FB724	----------	1111
516880	003FB724	3FE01111	addis r31,r0,0x1111	----------	1111

Figure 2–4: Example of ITR display

Error Messages Specific to ITR. The following are the error messages relevant to the ITR support.

1. *** S-Record: File path too long ***
2. *** S-Record: Not a valid file ***
3. *** S-Record: File open failed (bad path?) ***
4. *** S-Record: Non-hexadecimal digit ***
5. *** S-Record: File operation failure(s) ***
6. *** S-Record: No or incomplete associated image bytes ***
7. *** S-Record: Null character in file ***
8. *** S-Record: Line too long ***
9. *** S-Record: Start of line is bad ***
10. *** S-Record: Length field is too small ***
11. *** S-Record: Non-digit type character ***
12. *** S-Record: Address space wrapping not supported ***
13. *** S-Record: Internal problem, mixed endian layouts not supported ***

14. *** S-Record: Unable to allocate sufficient memory ***

15. *** S-Record: Internal problem, too many bytes requested at once ***

16. *** S-Record: Internal problem, region vs. content mismatch ***

17. *** S-Record: Internal problem, invalid cache entry accessed ***

18. *** S-Record: Internal problem, bad start region ***

19. *** Memory Image Disabled ***

***NOTE**. The error message 19 is displayed when the option Disabled is selected for the Memory Image Status field.*

Reference

Reference: Symbol and Channel Assignment Tables

This section lists the symbol tables and channel assignment tables for disassembly and timing.

Symbol Tables

The TMS555 support supplies two symbol-table files. The MPC565_Ctrl file replaces specific Control group values with symbolic values and the MPC565_CS file replaces the Chip_Select group with symbolic values. Symbol files can be applied to a group when the radix Symbolic is chosen.

Symbol tables are generally not for use in timing or MPC565_T support disassembly.

Tables 3–1 through 3–2 show the definitions for name, bit pattern, and meaning of the group symbols in the files MPC565_Ctrl and MPC565_CS.

Table 3-1: MPC565_Ctrl group symbol table definitions

Symbol	Control group value				Description
	AT2	BG~ BB~ BR~ TEA~	RD/WR~ OE~ BDIP~ TA~	TS~ BURST~ STS~ RETRY~	
BURST_FETCH	X	X X X 1	1 X X 0	X 0 1 X	Burst fetch
FETCH	0	X X X 1	1 X 1 0	1 1 X 1	Opcode fetch
READ	X	X X X 1	1 X 1 0	1 1 X 1	Read
WRITE	X	X X X 1	0 X X 0	1 1 X 1	Write
SHOW_CYCLE	X	X X X 1	X X X 1	1 X 0 1	Show cycle
EXTERNAL_READ	X	X X X 1	1 X X 1	0 X 0 1	External read
EXTERNAL_WRITE	X	X X X 1	0 X X 1	0 X 0 1	External write
TRANSFER_START	X	X X X 1	X X X 1	0 X X 1	Transfer start
TRANSFER_ERROR	X	X X X 0	X X X 1	1 X X 1	Transfer error
RETRY_CYCLE	X	X X X 1	X X X 1	1 X X 0	Retry cycle
----------	1	1 1 1 1	1 1 1 1	1 1 1 1	Invalid

Table 3-2: MPC565_CS group symbol table definitions

Symbol	Chip_Select group value CS0~ CS1~ CS2~ CS3~	Description
CS0~	0 1 1 1	Chip Select 0
CS1~	1 0 1 1	Chip Select 1
CS2~	1 1 0 1	Chip Select 2
CS3~	1 1 1 0	Chip Select 3
None	1 1 1 1	None

***NOTE**. The symbols that are not covered in Tables 3-1 and 3-2 will be shown in binary.*

Information on basic operations describes how to use symbolic values for triggering and for displaying other channel groups symbolically, such as for the addr channel group.

Channel Assignment Tables

Channel assignments shown in Table 3-3 through Table 3-10 use the following conventions:

- All signals are required by the support unless indicated otherwise.
- Channels are shown starting with the most significant bit (MSB) descending to the least significant bit (LSB).
- Channel group assignments are for all modules unless otherwise noted.
- A tilde (~) following a signal name indicates an active low signal.

Table 3-3 shows the probe section and channel assignments for the logic analyzer addr group and the microcontroller signal to which each channel connects. By default, this channel group is displayed in hexadecimal.

Table 3-3: addr group channel assignments

Section:channel	MPC565 signal name
A2:7-0	A8-A15
A1:7-0	A16-A23
A0:7-0	A24-A31

Table 3–4 shows the probe section and channel assignments for the Data group and the microcontroller signal to which each channel connects. By default, this channel group is displayed in hexadecimal.

Table 3-4: Data group channel assignments

Section:channel	MPC565 signal name
D3:7-0	D0 - D7
D2:7-0	D8 - D15
D1:7-0	D16 - D23
D0:7-0	D24 - D31

Table 3–5 shows the probe section and channel assignments for the Control group and the microcontroller signal to which each channel connects. By default, this channel group is displayed as symbols.

Table 3-5: Control group channel assignments

Section:channel	MPC565 signal name
C1:0	AT2
C2:7	BG~
C3:1	BB~
C3:2	BR~
QUAL:1	TEA~
Clock:3	RD/WR~
C1:5	OE~
C1:6	BDIP~
Clock:2	TA~
Clock:0	TS~
C1:7	BURST~
QUAL:0	BI~/STS~
C0:2	RETRY~

Table 3–6 shows the probe section and channel assignments for the Byte_Enable group and the microcontroller signal to which each channel connects. By default, this channel group is displayed as binary.

Table 3-6: Byte_Enable group channel assignments

Section:channel	MPC565 signal name
C1:4	WE0~
C1:3	WE1~
C1:2	WE2~
C1:1	WE3~

Table 3–7 shows the probe section and channel assignments for the logic analyzer Chip_Select group and the microcontroller signal to which each channel connects. By default, this channel group is displayed in symbols.

Table 3-7: Chip_Select group channel assignments

Section:channel	MPC565 signal name
A3:3	CS0~
A3:2	CS1~
A3:1	CS2~
A3:0	CS3~

Table 3–8 shows the probe section and channel assignments for the logic analyzer ITR group and the microcontroller signal to which each channel connects. By default, this channel group is not visible.

Table 3-8: ITR group channel assignments

Section:channel	MPC565 signal name
C0:4	VFLS1
C0:5	VFLS0
C2:2	VF0
C2:1	VF1
C2:0	VF2

Table 3–9 shows the probe section and channel assignments for the logic analyzer Misc group and the microcontroller signal to which each channel connects. By default, this channel group is not visible.

Table 3-9: Misc group channel assignments

Section:channel	MPC565 signal name
C0:3	PTR
C3:7	CR~
C3:6	KR
C3:4	IWP0
C3:3	IWP1
C3:0	LWP0
C2:6	DSCK
C2:5	DSDO
C2:4	DSDI
C3:5	RSTCONF~
C0:1	SRESET~
C0:0	HRESET~
A3:7	PORESET~
C0:7	TSIZ0
C0:6	TSIZ1

Table 3–10 shows the probe section, clock channel assignments and the MPC565 signal to which each channel connects.

Table 3-10: Clock channel assignments

Logic analyzer section & probe	AMP mictor & pin numbers	MPC565 signal name
Clock:3	C5	RD/WR~
Clock:2	D6	TA~
Clock:1	A6	CLKOUT
Clock:0	A5	TS~

Table 3–11 shows the probe section, qualifier assignments and the MPC565 signal to which each channel connects.

Table 3-11: Qualifier channel assignments

Logic analyzer section & probe	AMP mictor & pin numbers	MPC565 signal name
QUAL:0	D5	BI~/STS~
QUAL:1	C6	TEA~

***NOTE**. The signals listed in table 3–10 and 3–11 are not used as qualifiers or clocks except for the signal CLKOUT.*

CPU To Mictor Connections

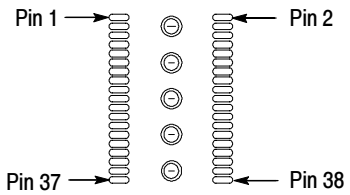
This section contains information about Mictor connections.

For design purposes, you may need to make connections between the CPU and the Mictor pins of the P6434 Mass Termination Probe. Refer to the *P6434 Mass Termination Probe* manual, Tektronix part number 070-9793-XX, for more information on mechanical specifications.

***NOTE**. To preserve signal quality in the target system, you should connect a 180 Ω resistor in series between each ball pad of the CPU and each pin of the Mictor connector. The resistor must be within 1/2 inch of the ball pad of the CPU.*

The recommended pin assignment is the AMP pin assignment, because the AMP circuit board layout model and other commercial CAD packages use the AMP numbering scheme. See Table 3–12.

Table 3-12: Recommended pin assignments for a Mictor connector (component side)

Type of pin assignment	Comments
Recommended  **AMP Pin Assignment**	Recommended. This pin assignment is the industry standard and is what we recommend that you use.

Tables 3–13 through 3–15 show the mictor pin connections for the logic analyzer and the AMP mictors.

Table 3-13: Mictor connections for Mictor A pins

AMP Mictor pin number	Logic analyzer channel name	MPC565 signal name
A5	Clock:0	TS~
A6	Clock:1	CLKOUT
A7	A3:7	PORESET~

Table 3-13: Mictor connections for Mictor A pins (Cont.)

AMP Mictor pin number	Logic analyzer channel name	MPC565 signal name
A15	A3:3	CS0~
A17	A3:2	CS1~
A19	A3:1	CS2~
A21	A3:0	CS3~
A23	A2:7	A8 (MSB)
A25	A2:6	A9
A27	A2:5	A10
A29	A2:4	A11
A31	A2:3	A12
A33	A2:2	A13
A35	A2:1	A14
A37	A2:0	A15
A8	A1:7	A16
A10	A1:6	A17
A12	A1:5	A18
A14	A1:4	A19
A16	A1:3	A20
A18	A1:2	A21
A20	A1:1	A22
A22	A1:0	A23
A24	A0:7	A24
A26	A0:6	A25
A28	A0:5	A26
A30	A0:4	A27
A32	A0:3	A28
A34	A0:2	A29
A36	A0:1	A30
A38	A0:0	A31 (LSB)

Table 3-14: Mictor connections for Mictor D pins

AMP Mictor pin number	Logic analyzer channel name	MPC565 signal name
D5	QUAL:0	BI~/STS~
D6	Clock:2	TA~
D7	D3:7	D0 (MSB)
D9	D3:6	D1
D11	D3:5	D2
D13	D3:4	D3
D15	D3:3	D4
D17	D3:2	D5
D19	D3:1	D6
D21	D3:0	D7
D23	D2:7	D8
D25	D2:6	D9
D27	D2:5	D10
D29	D2:4	D11
D31	D2:3	D12
D33	D2:2	D13
D35	D2:1	D14
D37	D2:0	D15
D8	D1:7	D16
D10	D1:6	D17
D12	D1:5	D18
D14	D1:4	D19
D16	D1:3	D20
D18	D1:2	D21
D20	D1:1	D22
D22	D1:0	D23
D24	D0:7	D24
D26	D0:6	D25
D28	D0:5	D26
D30	D0:4	D27
D32	D0:3	D28
D34	D0:2	D29

Table 3-14: Mictor connections for Mictor D pins (Cont.)

AMP Mictor pin number	Logic analyzer channel name	MPC565 signal name
D36	D0:1	D30
D38	D0:0	D31 (LSB)

Table 3-15: Mictor connections for Mictor C pins

AMP Mictor pin number	Logic analyzer channel name	MPC565 signal name
C5	Clock:3	RD/WR~
C6	QUAL:1	TEA~
C7	C3:7	CR~
C9	C3:6	KR
C11	C3:5	RSTCONF~
C13	C3:4	IWP0
C15	C3:3	IWP1
C17	C3:2	BR~
C19	C3:1	BB~
C21	C3:0	LWP0
C23	C2:7	BG~
C25	C2:6	DSCK
C27	C2:5	DSDO
C29	C2:4	DSDI
C33	C2:2	VF0
C35	C2:1	VF1
C37	C2:0	VF2
C8	C1:7	BURST~
C10	C1:6	BDIP~
C12	C1:5	OE~
C14	C1:4	WE0~
C16	C1:3	WE1~
C18	C1:2	WE2~
C20	C1:1	WE3~
C22	C1:0	AT2

Table 3-15: Mictor connections for Mictor C pins (Cont.)

AMP Mictor pin number	Logic analyzer channel name	MPC565 signal name
C24	C0:7	TSIZ0
C26	C0:6	TSIZ1
C28	C0:5	VFLS0
C30	C0:4	VFLS1
C32	C0:3	PTR
C34	C0:2	RETRY~
C36	C0:1	SRESET~
C38	C0:0	HRESET~

Specifications

Specifications

This section contains the specifications for the support.

Specification Tables

Table 4–1 lists the electrical requirements the system under test must produce for the support to acquire correct data.

Table 4-1: Electrical specifications

Characteristics	Requirements
Target system clock rate	
MPC555 specified clock rate	Maximum 40 MHz
MPC555 tested clock rate	Maximum 20 MHz
MPC565 specified clock rate	Maximum 56 MHz
MPC565 tested clock rate	Maximum 40 MHz
Minimum setup time required	2.5 ns
Minimum hold time required	0 ns

The TMS555 microcontroller support has been tested at 20 MHz for the burst mode and 40 MHz for the nonburst mode.

Replaceable Parts List

Replaceable Parts Lists

This section contains a list of the replaceable components and modules for the TMS555 MPC565 support. Use this list to identify and order replacement parts.

Parts Ordering Information

Replacement parts are available through your local Tektronix field office or representative.

Changes to Tektronix products are sometimes made to accommodate improved components as they become available and to give you the benefit of the latest improvements. Therefore, when ordering parts, it is important to include the following information in your order:

- Part number
- Instrument type or model number
- Instrument serial number
- Instrument modification number, if applicable

If you order a part that has been replaced with a different or improved part, your local Tektronix field office or representative will contact you concerning any change in part number.

Using the Replaceable Parts List

The tabular information in the Replaceable Parts List is arranged for quick retrieval. Understanding the structure and features of the list will help you find all of the information you need for ordering replacement parts. The following table describes the content of each column in the parts list.

Parts list column descriptions

Column	Column name	Description
1	Figure & index number	Items in this section are referenced by figure and index numbers to the exploded view illustrations that follow.
2	Tektronix part number	Use this part number when ordering replacement parts from Tektronix.
3 and 4	Serial number	Column three indicates the serial number at which the part was first effective. Column four indicates the serial number at which the part was discontinued. No entry indicates the part is good for all serial numbers.
5	Qty	This indicates the quantity of parts used.
6	Name & description	An item name is separated from the description by a colon (:). Because of space limitations, an item name may sometimes appear as incomplete. Use the U.S. Federal Catalog handbook H6-1 for further item name identification.
7	Mfr. code	This indicates the code of the actual manufacturer of the part.
8	Mfr. part number	This indicates the actual manufacturer's or vendor's part number.

Abbreviations

Abbreviations conform to American National Standard ANSI Y1.1–1972.

Chassis Parts

Chassis-mounted parts and cable assemblies are located at the end of the Replaceable Parts List.

Mfr. Code to Manufacturer Cross Index

The table titled Manufacturers Cross Index shows codes, names, and addresses of manufacturers or vendors of components listed in the parts list.

Manufacturers cross index

Mfr. code	Manufacturer	Address	City, state, zip code
80009	TEKTRONIX, INC.	P.O. BOX 500	BEAVERTON, OR, 97077-0001

Replaceable parts list

Fig. & index number	Tektronix part number	Serial no. effective	Serial no. discont'd	Qty	Name & description	Mfr. code	Mfr. part number
					STANDARD ACCESSORIES		
	071-1015-00			1	MANUAL,TECH INSTRUCTIONS,MPC565;TMS555	80009	071-1015-00

Index

Index

A

B

C

D

Q

R

S

T

U

V

W