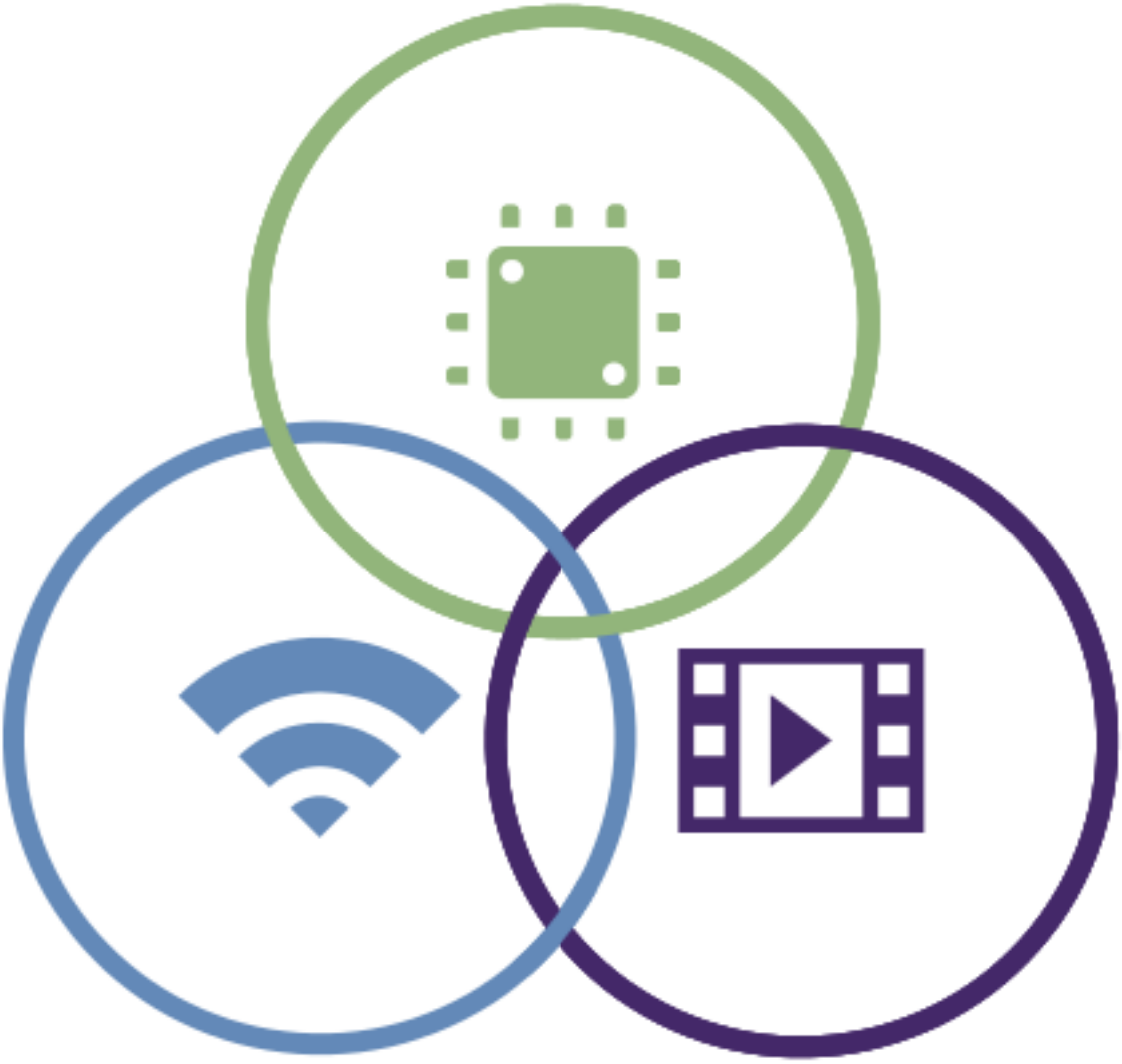
MIPS Debug

Probe Connection and Specifications

Revision: 1.4
(03/04/2017)
Public



Contents

List of Figures

List of Tables

1 Introduction

This document provides connection information and specifications for MIPS family of probes.

Licensing

Licenses for open source components can be found on the probe's in-built webserver at: http://img-sp00xxx/license where xxx is the serial number of the probe as printed on side and base, eg: http://img-sp00155/license.

Terminology

The following terms are used in this document:

SysProbe

SysProbe is the name of the master module that accommodates a sub-assembly PCB that provides a specific set of debug features. The name of the sub-assembly, such as SP55E-HD, can be read on the end panel of the SysProbe.

'probe' vs 'debug adapter'

Throughout this document the terms 'probe' and 'debug adapter are used interchangeably.

2 MIPS Probe specifications

This section describes physical and electrical specifications of MIPS probes, including the target connectors and cables required to connect to targets.

List of MIPS probes

A summary description of each probe is given below. Physical and electrical characteristics of the connections and probe requirements are later in the section.

SP55E description

The SP55E probe provides high-speed debug functions via a 14-way JTAG connection to target. Connection to host can be via Ethernet or USB.

SP55ET description

The SP55ET probe provides high-speed debug functions via a 14-way connection to target. In addition it has iFlowtrace and iMon ports. The iFlowtrace port takes trace data from the target's trace port, buffering the data in the probe before it is retrieved by the debugger on your host PC. The iMon port samples voltage drop across series resistors, enabling monitoring of changes in current at specific times. This data can be buffered on the probe before retrieval on the host.

Connection to host can be via Ethernet or USB.

SP55E-HD description

The SP55E-HD probe provides high-speed debug functions via a 10-way, 0.05/1.25mm-pitch connection to target.

This probe supports JTAG and the cJTAG IEEE1194.7 2 wire debug protocol.

From firmware version 2.2.0.0 it supports a subset of the cJTAG standard to allow interoperability with the cJTAG to JTAG adapter supplied with M-class cores.

Connection to host can be via Ethernet or USB.

SP55ET-HD description

The SP55ET-HD probe provides high-speed debug functions via a 20-way, 0.05/1.25mm-pitch connection to target. It also has iFlowtrace and iMon ports. The iFlowtrace port takes trace data from the target's trace port, buffering the data in the probe before it is retrieved by the debugger on your host PC. The iMon port samples voltage drop across series resistors, enabling monitoring of changes in current at specific times. This data can be buffered on the probe before retrieval on the host.

This probe supports EJTAG and the cJTAG IEEE1194.7 2 wire debug protocol.

From firmware version 2.2.0.0 it supports a subset of the cJTAG standard to allow interoperability with the cJTAG to JTAG adapter supplied with M-class cores.

Connection to host can be via Ethernet or USB.

SP58ET description

The probe is a specialist probe providing debug and tracing functionality with 16GB of DDR3 memory, a 256GB SSD and intel i5 host CPU. High-speed serial interfaces and 8GB of dedicated DDR3 memory mean it can sustain 2.5GB per second of trace data capture. The probe runs a version of XUBUNTU 14.

Connection to target is via 38-pin MICTOR for parallel trace or 22-pin Nexus for high speed serial trace and JTAG + sideband signalling.

The probe can interface with Codescape Debugger on a host computer, or Codescape Debugger can be run on the probe directly. The probe is shipped with Codescape Console pre-installed on the probe for low-level interaction directly with the target.

Connection to host is via Ethernet only.

SP55E interface specifications

This topic describes the general specifications of the SP55E interfaces.

Table 1: SP55E JTAG Interface

Feature	SP55E
Supported JTAG IR width	1-2048
Supported JTAG DR width	1-2048
Target must support BYPASS scan	Yes
Additional register delays in scan-chain between TAP and core	0-2047 before TAP 0-2047 after TAP
TCK maximum frequency	31.25MHz currently
TCK edge on which JTAG outputs transition (TDO, TMS)	Configurable to be either
TCK edge on which JTAG input is registered (TDI)	Configurable to be either
Multiple TAPs on scan-chain	Yes
Multiple cores on single TAP	Yes
TRST required	No
Target TCK system	Must be simple clock buffer only; no PLL/DLL is permitted
Driver strength for target SoC TDO (for 31.25MHz TCK down 30cm shielded ribbon cable)	12mA. This output should be source-terminated for a 50R transmission-line

Table 2: SP55E Host Interface

Feature	SP55E
Protocol	1 Gb Ethernet
Protocol	IP over USB (direct to host)

SP55E probe family Target connector details

The diagram and table in this section show the pinouts for target connectors compatible with current models of SysProbe.

JTAG Target connector - SP55E connector details

The information in this topic describes the pinouts and physical characteristics of the JTAG connector on a target when it is used with an SP55E probe.

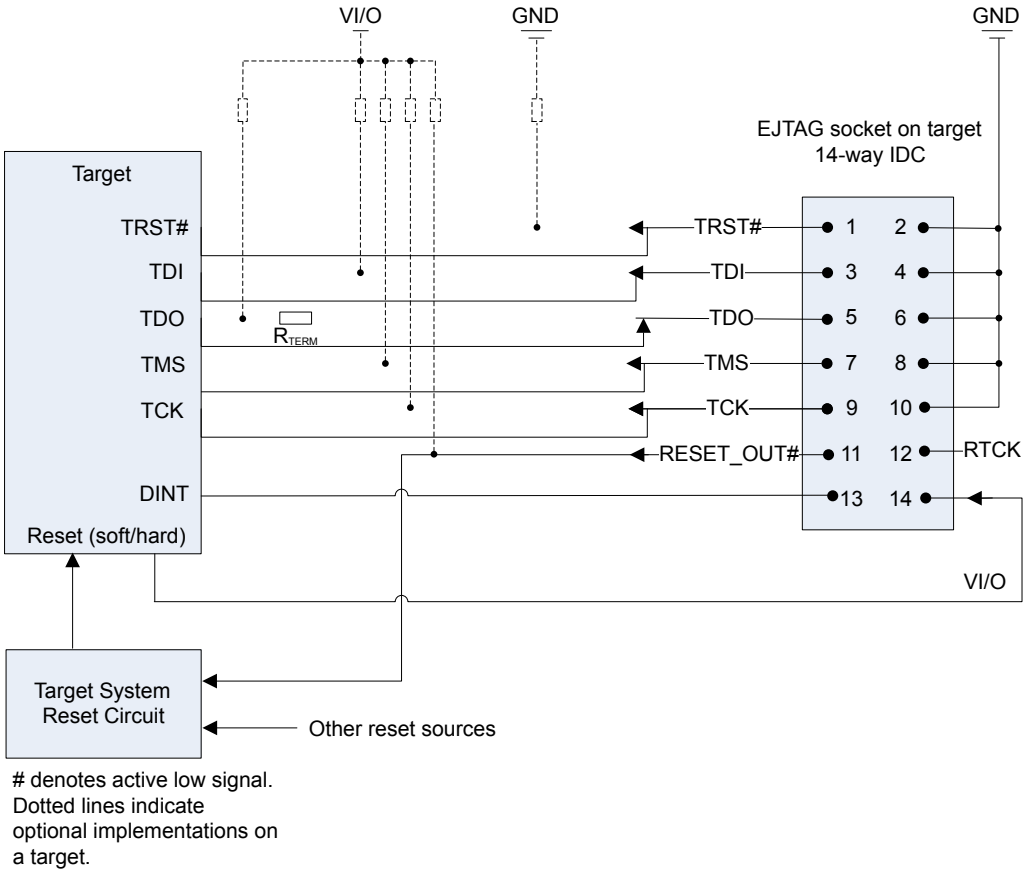


Figure 1: Target connector schematic

Note: If DINT is not implemented on the target, it is not necessary to provide a connection to the JTAG socket. It can be used to force an interrupt on the target from the probe. Rterm is optionally as part of the JTAG specification. For more information and details of design characteristics, request the JTAG specification from MIPS

Table 3: Target JTAG socket physical characteristics

Attribute	Value
Connector type	14-way IDC
Number of contacts	14
Number of rows	2
Gender	Male

Attribute	Value
Pitch	0.1"/2.54mm
Termination method	IDC

Table 4: Target JTAG Connection Pinouts

Pin	Signal	Direction	Pin	Signal	Direction
1	TRST# - Test Reset Input	Input	2	GND - Ground	GND
3	TDI – Test Data Input	Input	4	GND – Ground	GND
5	TDO – Test Data Output	Output	6	GND – Ground	GND
7	TMS – Test Mode Select Input	Input	8	GND – Ground	GND
9	TCK – Test Clock Input	Input	10	GND – Ground	GND
11	RESET# - System Reset	Input	12	RTCK – not implemented	
13	DINT default low.		14	VI/O – Target reference voltage.	Output

Note: # indicates an active low signal.
The table above displays the pinouts that need to be implemented on the target's JTAG connector, to support the JTAG protocol when used with an JTAG adapter board.

Pin 13 DINT

This pin can be driven low or high from Codescape Console using the command `config("assert dint",1)` where 1 is to drive DINT high. A '0' will drive DINT low. It defaults to low.

Pin 14 VI/O

This pin is an input circuit to the SP55E from the target. The normal requirement is 1.2mA at 1V2, rising to 3.3mA at 3V3.

SP55E-HD Target Connector Details

The information in this topic describes the pinouts and physical characteristics of the connector on a target when it is used with an SP55E-HD probe.

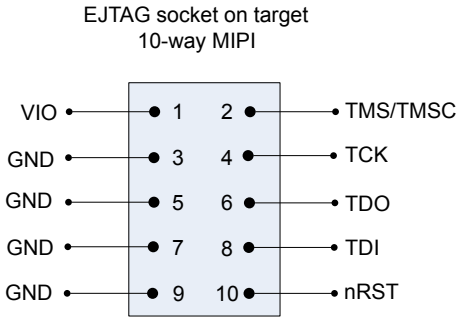


Figure 2: SP55E-HD target connector schematic

Table 5: Target SP55E-HD socket physical characteristics

Attribute	Value
Connector type	10-way MIPI-10-nRESET, ARM-10-JTAG
Number of contacts	10
Number of rows	2
Gender	Male
Pitch	0.05"/1.27mm
Termination method	MIPI

Table 6: Target SP55E-HD Connection Pinouts

Pin	Signal	Direction	Pin	Signal	Direction
1	VIO - Target reference voltage	Output	2	TMS – Test Mode Select Input or TMSC	Input (I/O for TMSC)
3	GND – Ground	GND	4	TCK – Test Clock Input	Input
5	GND – Ground	GND	6	TDO – Test Data Output	Output
7	GND – Ground	GND	8	TDI – Test Data Input	Input
9	GND – Ground	GND	10	nRST	Input

The table above displays the pinouts that need to be implemented on the target's JTAG connector, to support the JTAG protocol when used with an JTAG adapter board. The direction is relative to the target, so 'input' is an input to the target and an 'output' is an output from the target to the probe.

Notes:

- nRST is sometimes denoted as RST*
- MIPI defines pin 7 as a key with the pin removed from the target connector but grounded in the cable. The SP55E-HD implementation keeps the pin as a ground.
- Pins 2 and 4 have dual modes, switching to TMSC and TCKC respectively when processing cJTAG signals.
- RTCK (Return TCK) is not supported with this connector; if needed (for example, for HW emulators) the MIPS 14-pin 0.1" pitch connector is required.

- The 10 and 20-pin connectors do not include nTRST (tap reset, active low, sometimes labeled TRST*), an optional signal on MIPS EJTAG. Without nTRST, a target must provide its own means of resetting the TAP at power-up. In addition, with no nTRST, a board cannot include a pull-down resistor to the chip nTRST pin, otherwise the TAP will never come out of reset.

SP55ET-HD Target Connector Details

The information in this topic describes the pinouts and physical characteristics of the connector on a target when it is used with an SP55ET-HD probe.

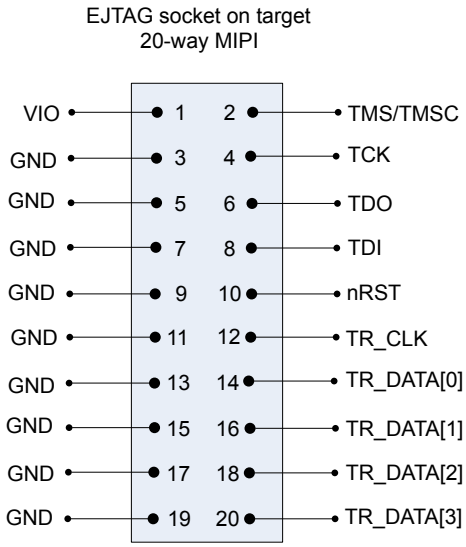


Figure 3: SP55E-HD target connector schematic

Table 7: Target SP55ET-HD socket physical characteristics

Attribute	Value
Connector type	20-way MIPI
Number of contacts	20
Number of rows	2
Gender	Male
Pitch	0.05"/1.27mm
Termination method	MIPI

Table 8: Target SP55E-HD Connection Pinouts

Pin	Signal	Direction	Pin	Signal	Direction
1	VIO - Target reference voltage	Output	2	TMS – Test Mode Select Input/TMSC	Input (I/O for TMSC)Output
3	GND – Ground	GND	4	TCK – Test Clock Input/ TCKC	Input
5	GND – Ground	GND	6	TDO – Test Data Output	Output

Pin	Signal	Direction	Pin	Signal	Direction
7	GND – Ground	GND	8	TDI – Test Data Input	Input
9	GND – Ground	GND	10	nRST	Input
11	GND – Ground	GND	12	TR_CLK	Output
13	GND – Ground	GND	14	TR_DATA[0]	Output
15	GND – Ground	GND	16	TR_DATA[1]	Output
17	GND – Ground	GND	18	TR_DATA[2]	Output
19	GND – Ground	GND	20	TR_DATA[3]	Output

The table above displays the pinouts that need to be implemented on the target's JTAG connector, to support the JTAG protocol when used with an JTAG adapter board. The direction is relative to the target, so 'input' is an input to the target and an 'output' is an output from the target to the probe.

Notes:

- nRST is sometimes denoted as RST*
- Pins 2 and 4 have dual modes, switching to TMSC and TCKC respectively when processing cJTAG signals.
- In cJTAG mode, pins 6 and 8 (TDO and TDI) are not used.
- The 10 and 20-pin connectors do not include nTRST (tap reset, active low, sometimes labeled TRST*), an optional signal on MIPS EJTAG. Without nTRST, a target must provide its own means of resetting the TAP at power-up. In addition, with no nTRST, a board cannot include a pull-down resistor to the chip nTRST pin, otherwise the TAP will never come out of reset.

SP55ET trace connection to target

This section describes the connections for the SP55ET iFlowtrace model of the SysProbe.

The SP55ET board offers three target connectors:

JTAG

This offers the same JTAG connection available from the standard SP55E.

iFlowtrace

The SP55ET iFlowtrace port takes trace data from the target's trace port. Trace data is timestamped and buffered locally in the SP55ET RAM. The collection of trace data is only started in the core when signalled appropriately.

iFlowtrace uses a 4-bit data port and trace clock signal.

Current Sense (iMon)

The Current Sense port samples voltage drop across series resistors, enabling monitoring of changes in current at specific times.

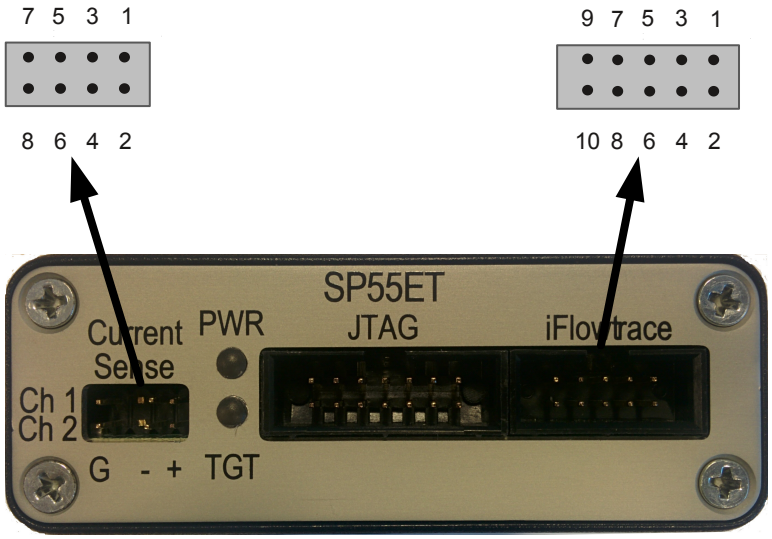


Figure 4: SP55ET Pin orientations

iMon connector

This topic describes the iMon connections on the SP55ET and SP58ET probes.

iMon Current Sensing specification

This topic describes the current-sensing function (iMon).

The current sense function (iMon) provides a technique for correlating code execution with current flow on your target.

Two resistors (channels) can be monitored at the same time. Time-stamping in the probe then enables a comparison to be drawn between current used on each channel and the code being executed at that time.

Voltages

Maximum voltage is 12V.

Maximum voltage is 12V.

+-5%

Board requirements

Each voltage line to be monitored must have a resistor placed in series, with the resistor connected in parallel to the pins on the target's iMon port as per the pinout in *Table 9: iMon Current Sense connector pinout* on page 23.

iMon cable

A suitable cable is supplied with the probe.

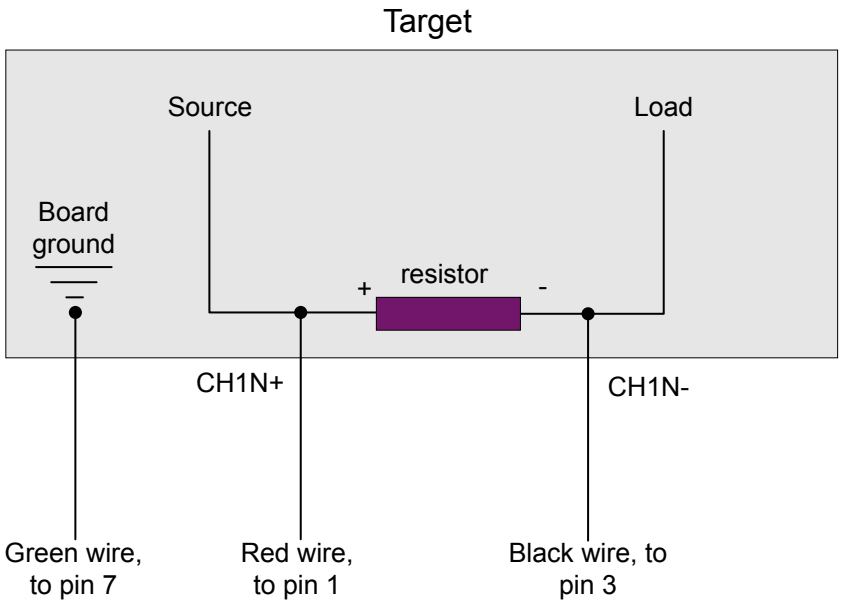


Figure 5: iMon Current Sensing target resistor diagram

iMon Current Sense pinout

This topic gives the pinout of the iMon connector.

Table 9: iMon Current Sense connector pinout

IDC pin number	Signal name at debug adapter	Signal description
1	CH1 IN+	Connected to *positive* side of the channel one series resistor.
2	CH2 IN+	Connected to *positive* side of the channel two series resistor.
3	CH1 IN-	Connected to *negative* side of the channel one series resistor
4	CH2 IN-	Connected to *negative* side of the channel two series resistor
5	NC	Not connected
6	NC	Not connected
7	GND	Connected to (on target board) to board digital ground.
8	GND	Connected to (on target board) to board digital ground.

iFlowtrace connector

This topic describes the pinout of the iFlowtrace connector.

iFlowtrace socket pinout

IDC pin number	Signal name at debug adapter	Signal description
1	CLK	Trace clock signal.
2	GND	Ground
3	DATA0	iFlowData bit 0.
4	GND	Ground
5	DATA1	iFlowData bit 1
6	GND	Ground
7	DATA2	iFlowData bit 2
8	GND	Ground
9	DATA3	iFlowData bit 3
10	GND	Ground

SP58ET Specifications and Connectors

SP58ET Target Connectors

This topic gives information on the connectors used by the SP58ET to connect to targets.

The SP58ET has the following probe-target connectors:

PDTrace

Provides a 38-pin MICTOR connector for debugging via JTAG and PD Trace data collection.

Nexus

Provides a 22-pin Nexus connector for high speed serial trace, debug via JTAG + sideband signalling.

Current Monitor

This connector must be used in conjunction with either the PDTrace or Nexus connector when tracing data from a core with the PDTrace block. It enables you to correlating code execution with current flow on your target. See *iMon connector* on page 22 for specifications.

SP58ET PDTrace Connector Details

The information here describes the pinouts and physical characteristics of the PDTrace MICTOR connector on a target when it is used with an SP58ET probe.

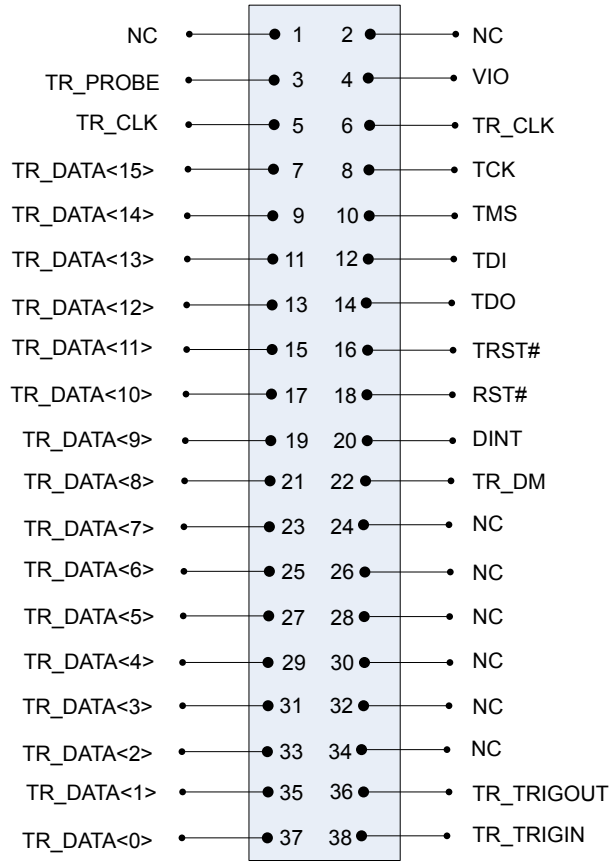


Figure 6: SP58ET PDTrace (MICTOR) target connector schematic

Note:

- Signal ground is provided by the central earthing strip, which consists of 5 lugs. These should all be connected directly to the target board ground plane to attain the proper impedence for the high-speed trace signals.
- VIO is the I/O voltage for both PDTrace and JTAG.

Table 10: PDTrace target socket physical characteristics

Attribute	Value
Connector type	38-way MICTOR
Number of contacts	38
Number of rows	2
Gender	Female
Pitch	0.025"/0.635mm

Table 11: Target SP58ET PDTrace Connection Pinouts

Pin	Signal	Direction	Pin	Signal	Direction
1	Not connected	NC	12	JTAG Data Input	Input
2	Not connected	NC	14	JTAG Data Output	Output

Pin	Signal	Direction	Pin	Signal	Direction
3	TR_Probe, enables probe interface block when driven to ground.	Input	16	JTAG Reset	Input
4	I/O Voltage for trace data and JTAG	Output	18	Board-level Reset, driven low by probe to force target into reset state.	Input
5	Trace port clock	Output	20	Debug interrupt	Input
6	Trace port clock	Output	22	Debug Mode	Output
7 - 37	Trace Port Data, on odd pins	Input	24-34	NC	
8	JTAG Clock	Input	36	Trigger output from TCB	Output
10	JTAG Mode Select	Input	38	Trigger input to TCB	Input

The table above displays the pinouts that need to be implemented on the target's MICTOR connector when connecting to the PDTrace port on the SP58ET. The direction is relative to the target, so 'input' is an input to the target and an 'output' is an output from the target to the probe.

SP58ET Nexus Connector Details

The information here describes the pinouts and physical characteristics of the Nexus connector on a target when it is used with an SP58ET probe.

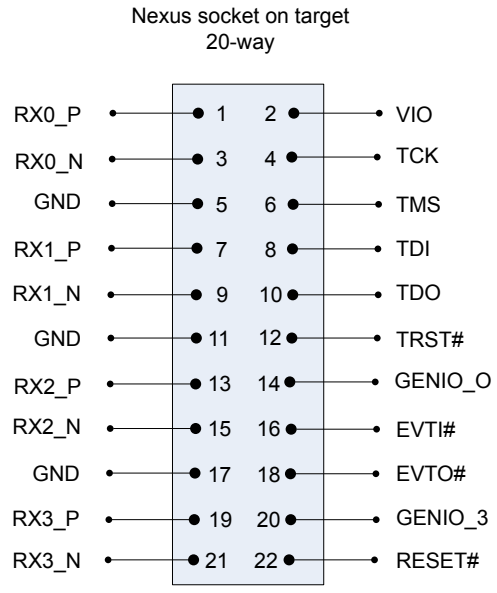


Figure 7: SP58ET Nexus target connector schematic

Table 12: Nexus target socket physical characteristics

Attribute	Value
Connector type	Nexus 5001 2x11
Number of contacts	22

Attribute	Value
Number of rows	2
Gender	Female

The SP58ET is supplied with a Samtec cable compatible with Samtec connectors such as the ASP-137969-01.

Table 13: SP58ET Nexus Target Connection Pinouts

Pin	Signal	Direction	Pin	Signal	Direction
1	RX0_P Differential trace data, lane 0, P signal.	Output	2	VIO - Target reference voltage	Output
3	RX0_N Differential trace data, lane 0, N signal	Output	4	TCK – Test Clock Input	Input
5	Ground	GND	6	TMS – Test Mode Select Input	Input
7	RX1_P Differential trace data, lane 1, P signal	Output	8	TDI – Test Data Input	Input
9	RX1_N Differential trace data, lane 1, N signal	Output	10	TDO – Test Data Output	Output
11	Ground	GND	12	TRST# tap reset	Input
13	RX2_P Differential trace data, lane 2, P signal	Output	14	GENIO_0, GPIO signal	
15	RX2_N Differential trace data, lane 2, N signal	Output	16	EVTI#, Probe event signal	Input
17	Ground	GND	18	EVTO#, Target event signal	Output
19	RX3_P Differential trace data, lane 3, P signal	Output	20	GENIO, GPIO signal	I/O
21	RX3_N Differential trace data, lane 3, N signal	Output	22	RESET#, Board-level Reset, driven low by probe to force target into reset state.	Input

The table above displays the pinouts that need to be implemented on the target’s Nexus connector when connecting to the Nexus port on the SP58ET. The direction is relative to the target, so 'input' is an input to the target and an 'output' is an output from the target to the probe.

SysProbe JTAG characteristics

This topic describes the signal characteristics for the JTAG input and output from the SysProbe probes.

The SysProbe JTAG outputs are driven by DDR output registers, (before going through voltage translation), yielding very low skew between TCK edges and TMS/TDO edges.

The TCK period must be long enough, after considering all skews, for all setup and hold time requirements to be met. For systems with high signal skew, the TCK period should be made longer.

The standard JTAG configuration is:

- SysProbe TMS/TDO to change on the falling-edge of TCK, ready for the target to register these signals on the following TCK rising-edge.
- SysProbe TDI to be registered on the rising-edge of TCK, after the target has produced edges on the falling-edge of TCK.

This configuration provides a half TCK-period for setup-times, and a half TCK-period for hold-times, and potentially providing margin for signal-skew between the probe and the target.

In many cases, more setup-time is required than hold-time, and so if a greater proportion of the TCK period is used for setup-time, a shorter TCK period may be used.

Non-standard JTAG output configuration

The TCK edge on which the TMS/TDO outputs transition is programmable.

SysProbe can be configured to output TMS/TDO 8ns after the rising-edge of TCK, still providing some hold-time and margin for skew, but potentially meeting the target setup-time requirement with a shorter TCK period.

This feature is disabled by default.

Non-standard JTAG input configuration

The TCK edge on which the TDI input is registered is programmable.

As the target produces TDI edges on TCK in response to SysProbe generating edges, a delay is guaranteed between the SysProbe producing TCK and receiving TDI events as follows:

1. SysProbe generates TCK falling-edge
2. TCK edge propagates through voltage-translation buffer and cable
3. Target produces edge on TDI
4. TDI edge propagates through cable and voltage-translation buffer

This delay guarantees significant TDI hold-time, and so it’s acceptable for SysProbe also to register TDI on the falling-edge. This configuration can potentially meet the SysProbe setup-time requirement with a shorter TCK period.

This feature is enabled by default.

JTAG signal timing

Timing diagrams are given for two cases:

1. JTAG outputs change on falling-edge of TCK (the JTAG standard)
2. JTAG outputs change on rising-edge of TCK

In each case, TCK is running at 31.25MHz.

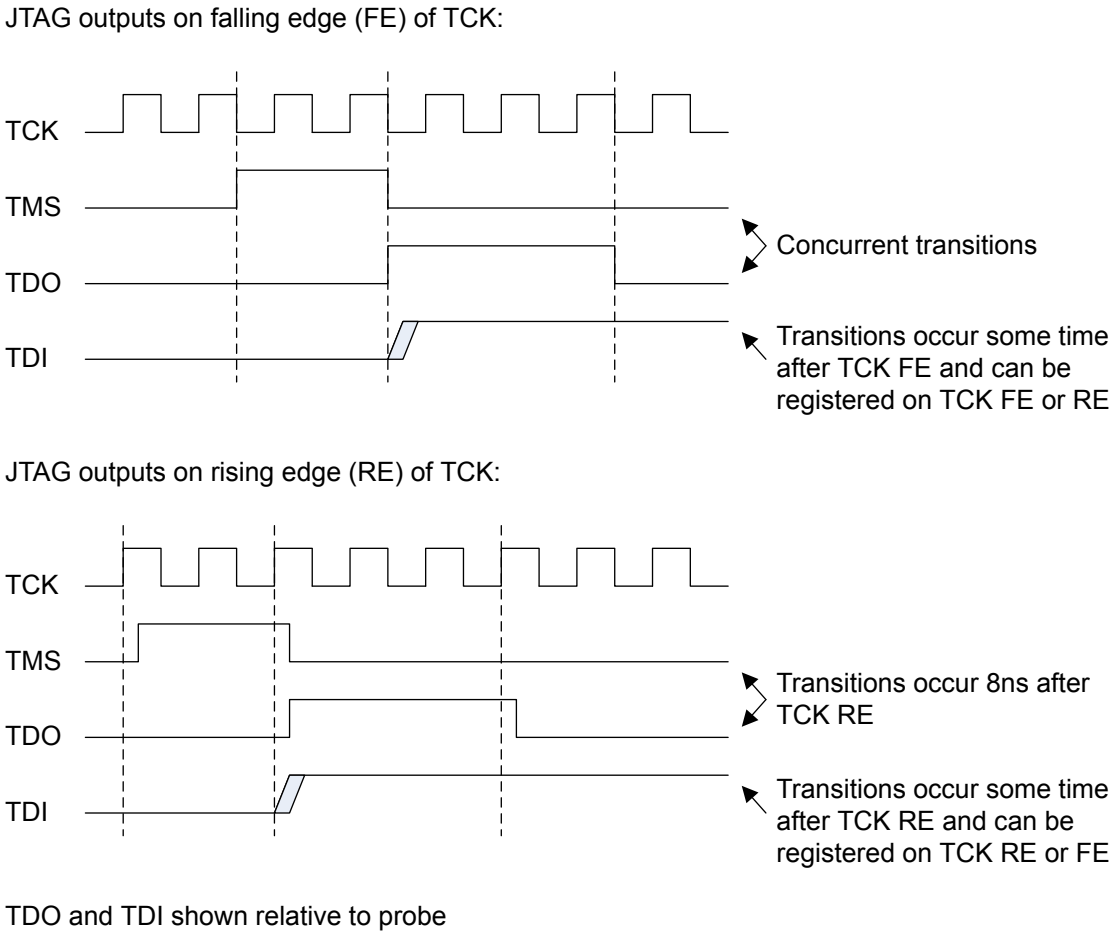


Figure 8: JTAG signal timing

JTAG DC Characteristics

This topic describes the DC characteristics of the JTAG connector on MIPS probes.

Table 14: MIPS JTAG DC Characteristics

Description	Condition	Symbol	Min	Max	Units
Input low voltage	VIO = 0.8V	VIL	-	0.24	V
	VIO=1.1V to 1.95V		-	0.35xVIO	V
	VIO=2.3V to 2.7V		-	0.7	V
	VIO=3.0V to 3.3V		-	0.8	V
Input high voltage	VIO = 0.8V	VIH	0.56	0.8	V
	VIO=1.1V to 1.95V		0.65xVIO	VIO	V

Description	Condition	Symbol	Min	Max	Units
	VIO=2.3V to 2.7V		1.6	VIO	V
	VIO=3.0V to 3.3V		2	VIO	V
Target I/O voltage		VIO	0.8	3.3	V
Input current		II	-	+/-5	uA
Input current, VIO	VIO=0.8V-3.3V	IVIO	-	3.4	uA
Output low voltage	IO=100uA	VOL	-	0.1	V
	VIO=0.8V-3.3V				
	IO=3mA		-	0.25	V
	VIO=1.1V				
	IO=6mA		-	0.35	V
	VIO=1.4V				
	IO=8mA		-	0.45	V
	VIO=1.65V				
	IO=9mA		-	0.55	V
	VIO=2.3V				
	IO=12mA		-	0.7	V
	VIO=3.0V				
Output high voltage	IO= -100uA	VOH	VIO-0.1	-	V
	VIO=0.8V-3.3V				
	IO= -3mA		0.85	-	V
	VIO=1.1V				
	IO= -6mA		1.05	-	V
	VIO=1.4V				
	IO= -8mA		1.2	-	V
	VIO=1.65V				
	IO= -9mA		1.75	-	V
	VIO=2.3V				
	IO= -12mA		2.3	-	V
	VIO=3.0V				

Notes for making your own JTAG cable

EMC Compliance

To fully comply with EC directive 2004/108/EC concerning emissions and immunity you must connect the flying ground lead (if fitted) on the supplied cable to a suitable secure earth point on your target system.

The JTAG debug output connector on the debug adapter is a 14-way IDC.

If you are making your own cable to connect to a custom installation, note that the JTAG interface has a theoretical maximum length of 300mm based on a 20MHz clock speed. As a guideline we recommend that the cable does not exceed 250mm in length and the on-board track length to the processor does not exceed 50mm where possible. You may be able to achieve longer cable lengths at slower clock speeds but performance is not guaranteed. All cables must be screened and earthed.

Board and cable impedance matching

Even though the JTAG signals are relatively slow some thought is needed when routing these signals for the target system PCB design.

The 14-way cable has a characteristic impedance of 65R, ideally the PCB traces for the JTAG signals should match this. If the impedances cannot be matched then the traces should be kept short, sub 5cm. All the traces need to kept to similar lengths (within 1cm difference) to avoid skew.

Special care is needed on the TDO line (data out of the target SoC), the pad drive strength needs to be strong enough to deal with the relatively high capacitance of the cable + traces, but not too high so that it will generate very fast slew-rate edges. We recommend a value between 4mA and 12mA. Most output drivers will have a relatively low impedance, this needs matching to the PCB traces and cable with a source termination resistor (RTERM on schematic), typical values will be in the range 15R - 33R.

SysProbe PDtrace characteristics

MIPS offers a range of probes supporting interaction with the PDtrace block in cores. This section describes the signal timing and DC characteristics for probes using MICTOR connectors for PDtrace

The PDtrace capture device is a Xilinx Kintex7 (speed-grade 1).

PDtrace is a DDR system, where TR_DATA is sampled on RE (rising edge) and FE (falling edge) of TR_CLK. The TR_CLK period is specified RE-to-RE.

PDtrace signal timing

Table 15: PDtrace signal timing requirements

Signal	Period RE-RE	Jitter tolerance	Duty cycle	Skew
TR_CLK	100ns max 3.333ns min	< 20% of clock input period or 1ns max	Allowable 35% Target 50%	1ns max to TR_DATA Target 0ns

Note:

- Maximum TR_CLK rate 300MHz.
- The period is rising-edge to rising-edge.
- Longer TR_CLK periods may be supported in future (if there is demand), by adding a low-frequency mode.

PDtrace DC Characteristics

TR_CLK

High-impedance LVCMOS input, driven by the target. DC-voltage determined by target VIO (see *Table 14: MIPS JTAG DC Characteristics*

TR_DATA

High-impedance LVCMOS input, driven by the target. DC-voltage determined by target VIO. Trace data is sampled twice from TR_DATA in each TR_CLK clock period.

TR_PROBE

Static activation CMOS signal which indicates the presence of the probe. DC characteristics are the same as the JTAG port. DC-voltage determined by target VIO.

TR_DM

Driven by target. Indicates that the target is in debug mode.

TR_TRIG_IN

This is an asynchronous signal with no timing requirements.

Cable and target connection characteristics

Target drivers must be source terminated.

Output impedence of cable drivers should be 50Ohms to minimise reflections and reduce chance of ISI. That applies to TR_CLK, TR_DATA, TR_DM and TR_TRIG_OUT.

3 Using and Configuring MIPS probes

This section describes how to connect MIPS probes to targets and how to reflash their firmware.

SP55E Overview

The SP55E is a high-speed debug adapter that uses JTAG and TCP/IP protocol to connect a host debugging PC to a target. Connection between debugging PC and an SP55E can be via Ethernet or a direct USB cable. Both connections use TCP/IP protocol. Although USB and Ethernet can be connected at the same time, only one of them will be used. When an SP55E is connected to a PC via USB, it presents as a client-mode network adapter.

SP55 Power requirements

The SP55E is supplied with a 12V, 15W DC power supply (centre pin positive). The board can be run on a supply voltage from 5v to 12V. Wattage requirements will depend on the board activity.

SP55E Connectors

This topic lists the characteristics of the standard connectors on the SP55E probe.

USB

Connector type	microUSB
Protocol	Client-mode, utilizing a network adapter for TCP/IP.

RJ45/Ethernet

Connector type	RJ45
Protocol	Ethernet TCP/IP

JTAG

Connector type	14-way IDC, 2-row, 2.54mm pitch
Protocol	JTAG compatible with targets complying with MIPS JTAG architecture and the MIPS OCI architecture. Note that an adapter is required to use the SP55E with Meta targets.

SP55E External LEDs

This topic describes the LEDs found on all SP55E models.

PWR LED

Shows a steady green light when the SP55E is powered up.

TGT LED

Shows a steady green light when the target is powered up and connected to the SP55E by the ribbon cable.

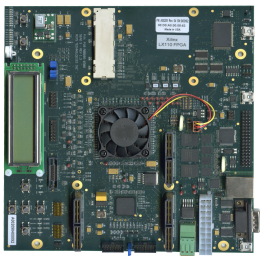
RJ45 Ethernet socket

The RJ45 socket has two built-in LEDs. If your SP55E is powered up and connected to a target, the LEDs give the following indications:

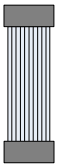
Steady green LNK	Ethernet link OK
Flashing green ACT	Ethernet activity

SP55E Connections

This topic describes connections between the SP55E, the target and the host.

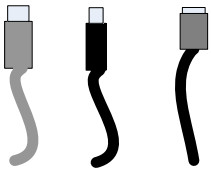


MIPS target board



JTAG cable:
Both the target and Sysprobe must be powered off when connecting or disconnecting the JTAG cable.

USB/RJ45 cable:
The USB and Ethernet cable can be connected at the same time but only one will establish TCP/IP connection. It is advisable to only connect one at a time.



Micro USB

RJ45 Ethernet

12V DC from AC-DC adapter, centre pin positive.

Figure 9: SP55E connection to target

The SP55E can connect to the debugger host PC either by USB (connected directly to the host PC) or via Ethernet.

SP55E establishing host-probe connection

This topic describes the connections between host and probe.

Several members of the SysProbe family can connect to the debugger host PC either by USB (connected directly to the host PC) or via Ethernet (see *List of MIPS probes* on page 14). The instructions here apply to any probes with that dual-capability.

SP55E RJ45/Ethernet connection to host

When an Ethernet cable is connected to the SP55E, the probe tries to connect to DHCP and request an IP address. Default DNS name of the SP55E is img-sp***** where ***** is the last five digits of the SP55E's serial number.

USB connection to host

USB IP address

The USB connection will present itself as a network adapter when connected to a Host PC running Windows or Linux. On connection, the Host PC will request an address from the SP55E. The SP55E will serve an IP address from the range 169.254.100.0 to 169.254.254.254. This will be the IP address of the USB port as seen from the host.

The address will be static, derived from the serial number of the SP55E.

The number is derived by the divisor and modulus of the last 4 digits of the serial number when divided by 100. For example an SP55E with a serial number of 02DALS32000378 would obtain an IP address of 169.254.3.78.

Troubleshooting USB-host connections

Ping the address

If Codescape cannot detect the SP55E, try pinging the IP derived from the probe's serial number.

Connection on Linux

Using older versions of some Linux distributions connectivity to the SP55E through USB isn't automatically setup and visible as an active network interface. In this case, the ping test will fail. Check that the probe has been discovered using lsusb or checking dmesg. For example:

```
Host$ lsusb
        Bus 002 Device 011: ID 0525:a4a2 Netchip Technology, Inc. Linux-USB Ethernet/
RNDIS Gadget
```

Look for the interface in ifconfig –a. For example:

```
Host$ ifconfig -a
        usb0      Link encap:Ethernet  HWaddr 76:1A:33:9D:CF:C9
        inet6 addr: fe80::741a:33ff:fe9d:cfc9/64 Scope:Link
        UP BROADCAST RUNNING MULTICAST  MTU:1494  Metric:1
        RX packets:12 errors:0 dropped:0 overruns:0 frame:0
        TX packets:4 errors:0 dropped:0 overruns:0 carrier:0
        collisions:0 txqueuelen:1000
        RX bytes:3269 (3.1 KiB)  TX bytes:332 (332.0 b)
```

If no IP address is assigned, the DHCP daemon may not be setup for this interface automatically. This can be configured using the network manager/network config that is provided by your Linux distribution. Either you can set the IP address statically or start avahi-autoipd daemon on the interface. For example:

```
Host$ sudo avahi-autoipd -D usb0
```

With the avahi autoipd daemon running on the network interface, ifconfig should show the interface with an IP address. For example:

```
Host$ ifconfig
        usb0      Link encap:Ethernet  HWaddr 76:1A:33:9D:CF:C9
        inet6 addr: fe80::741a:33ff:fe9d:cfc9/64 Scope:Link
        UP BROADCAST RUNNING MULTICAST  MTU:1494  Metric:1
        RX packets:12 errors:0 dropped:0 overruns:0 frame:0
        TX packets:10 errors:0 dropped:0 overruns:0 carrier:0
        collisions:0 txqueuelen:1000
        RX bytes:3269 (3.1 KiB)  TX bytes:696 (696.0 b)
        usb0:avahi Link encap:Ethernet  HWaddr 76:1A:33:9D:CF:C9
        inet addr:169.254.10.129  Bcast:169.254.255.255  Mask:255.255.0.0
        UP BROADCAST RUNNING MULTICAST  MTU:1494  Metric:1
```

Note: If setting the IP address manually, it is important to ensure the scope is set to link-local.

Connection on Windows

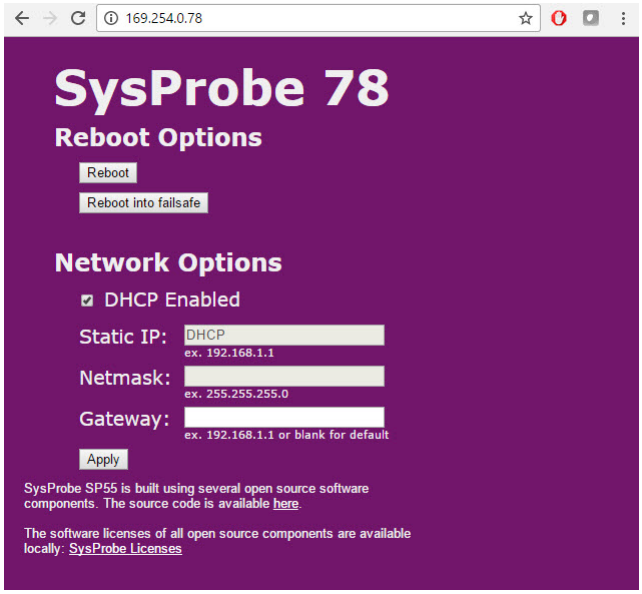
On Windows the probe will be identified as a RNDIS device. If connection has not worked, try updating the Windows RNDIS driver. You can also try adding a new 'Remote RNDIS Compatible Device' from your Devices window.

Setting a Static IP for SP55E probes

Probes such as the SP55E range can have a static IP address set for their Ethernet connection by using the USB-host IP link and a web page interface. This procedure is explained below.

Determine the static IP address you want to use and check that the USB IP connection to your host is working (see *USB connection to host* on page 39). You will need the USB IP connection address.

1. Open a web browser and enter the USB connection IP address.
   This should open a page like this:



2. Uncheck the DHCP box.
3. Enter the static IP address in the Static IP field.
4. Enter the appropriate Netmask, for most networks 255.255.255.0 will suffice.
5. Enter a Gateway if required for your network.
6. Click Apply.
   The Static IP address will now be set. To unset it, use the USB address again to connect via a web browser, check the DHCP box and click Apply.

SP58 host-probe connection

SP58ET External LEDs, Switches and other connectors

SP58ET Front Panel LEDs

This topic describes the external LEDs found on the SP58ET probe.

PDTrace LEDs

PWR

Lit when power is detected on the VIO pin to the PDTrace port.

JTAG

Lit when activity is detected on the JTAG pins for the PDTrace port.

TRACE

Lit when activity is detected on the trace pins for the PDTrace port.

Nexus LEDs

PWR

Lit when power is detected on the VIO pin to the Nexus port.

JTAG

Lit when activity is detected on the JTAG pins for the Nexus port.

TRACE

Lit when activity is detected on the trace pins for the Nexus port.

General LEDs

PWR

Shows a steady green light when the probe is powered up.

RDY

Lit when the probe is booted and ready for use.

SP58ET External Switches and Buttons

ON/OFF (rear panel)

Turns power on/off to the probe. In the off position the probe is completely powered off.

RST (front panel)

Pressing this switching in will reboot the probe. Note that the the RDY LED will be lit once the probe has rebooted.

SP58ET Rear Panel Connectors

This topic describes the connectors found on the rear panel of the SP58ET.

HDMI

For connection to a monitor when working with the on-board OS.

USB3 ports

The four USB3 ports can be used to connect various peripherals such as keyboard, mouse, memory sticks or external hard drives. External hard drives and memory sticks should automount.

FPGA USB-JTAG

This is a diagnostic port that enables connection to the on-board FPGA JTAG. It can be used to configure the FPGA and program the configuration flash. It is not intended for general use and is a diagnostic tool only in the event of probe failure.

Connecting or logging in to the SP58ET

The SP58ET offers several connection methods:

- Direct login (plug in monitor to HDMI port, keyboard and mouse to USB3 ports).
- Connect from Codescape Debugger.
- Remote connection via SSH, RDP or VNC.

Direct login will give access to the pre-installed functions such as Codescape Console. Other software can be downloaded and installed on the probe.

SP58ET Default username and password

A default account is pre-created on each SP58ET. This can be used when accessing the probe via remote connection or when using a keyboard and mouse and logging in directly.

Username	sysprobe
Password	sysprobe

For security purposes it is recommended that you alter the default username and password.

Remote Login to the SP58ET

The SP58ET supports remote login via three protocols; SSH, RDP or VNC.

Connecting using SSH

Connection should be made using the SSH '-Y' option to allow the use of GUI programs such as such as ‘xfc4-terminal’ (console), and Codescape Debugger.

If you are running Linux on your host, you can connect using natively installed SSH. If you are running Windows, you will need an SSH client such as Putty and an X-server (to use GUI apps) such as Xming.

Table 16: SSH Connection parameters

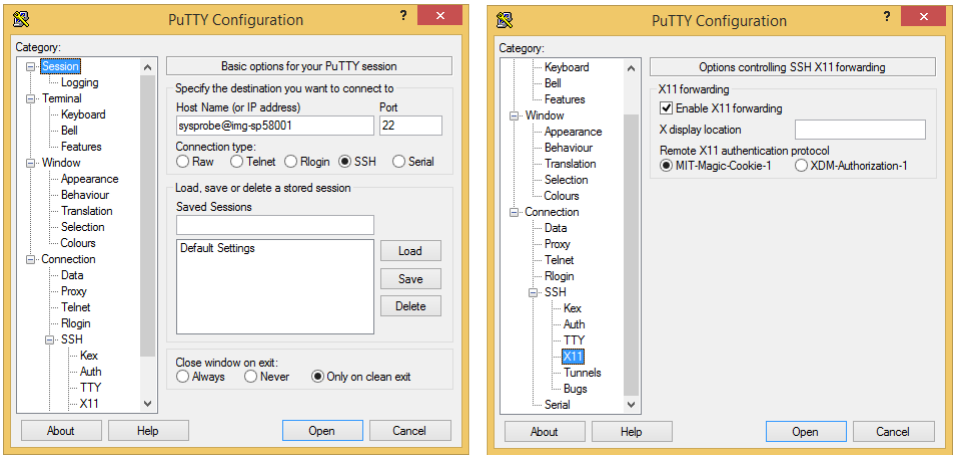
Parameter	Value
Username	`sysprobe`
Remote_host	`img-sp58***` where *** are the last 3 integers of your probe serial number.

Parameter	Value
Port	22

For example, on Linux:

```
ssh -Y sysprobe@img-sp58001
```

On Windows, using PuTTY



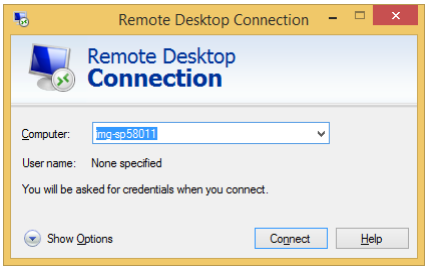
Connecting using RDP

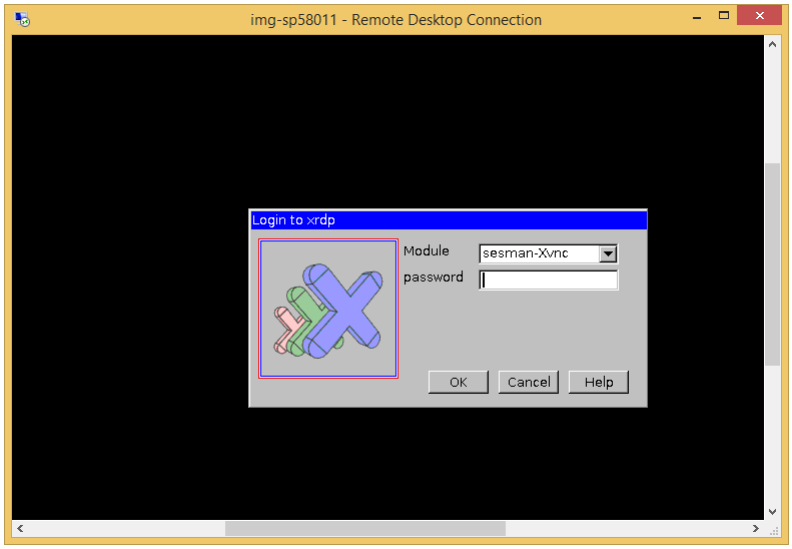
You can use any Remote Desktop client with RDP or VNC protocol to connect with the SP58ET. On Windows you can use the'Remote Desktop Connection' program and on Linux you can use KRDC or any similar program that uses the RDP or VNC protocol.

Table 17: RDP Connection parameters

Parameter	Value
Computer	`img-sp58***` where *** are the last 3 non-zero integers of your probe serial number.
Module Login	`sesman-Xvnc`
Password	Default, see *Connecting or logging in to the SP58ET*

For example, using RDC on Windows:





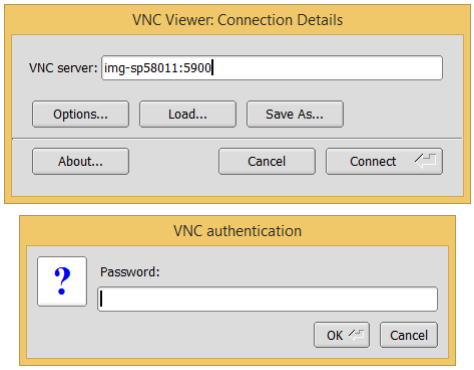
Connecting using VNC

Any standard VNC client can be used on Windows or Linux.

Table 18: VNC Connection parameters

Parameter	Value
VNC server	`img-sp58***` where *** are the last 3 non-zero integers of your probe serial number.
Port	5900
Password	Default, see *Connecting or logging in to the SP58ET*

For example, using TigerVNC:



SP58ET Troubleshooting

This topic describes how to diagnose and resolve any error conditions that are specific to the SP58ET.

FPGA upgrade needed. Please fit FPGA programming loopback cable

This message may appear in Codescape Console, Codescape Debugger or on the SP58ET if you are logged in to the probe. It indicates that that the FPGA needs reprogramming back to the factory defaults. Connecting a loopback cable between a USB port and the FPGA USB-JTAG port will force the FPGA to be re-programmed with the factory image.

Recovery process:

1. Connect USB cable between any of the USB3 ports and the port labelled 'FPGA USB-JTAG'.

   The probe will reboot. If you are connected via Codescape Console then it will not respond to commands, if connected via Codescape Debugger then it may disconnect.
2. Wait several minutes until the probe responds to commands in Codescape Console or reconnects in Codescape Debugger.
3. Remove the USB cable.

Connecting to a probe from Codescape Console

This topic explains how to connect to a probe with Codescape Console so that commands can be issued to interact with the target and probe.

Codescape Console is an interactive Python shell with built-in extensions for debugging (via a debug adapter) and control of a debug adapter. It can be used for reflashing, testing and for target bring-up scripts. It is installed when installing Codescape Debugger. In some circumstances it may be installed separately. For example it is installed independently on some probes.

The connection command is different when connecting to a probe if you are runnning Codescape Console on the probe.

To connect from a separate host

Your probe must be connected to your host. If you are going to issue target commands, it must also be connected to a target. See *SP55E Connections* on page 38.

1. Change directory to the Scripts directory below your Python location.
2. Start Codescape Console and connect to the probe using the command `CodescapeConsole sp####`

   Where #### are the last 4 non-zero digits from the serial number on the SysProbe (omit leading zeros if there are less than 4 non-zero digits).

For example:

```
C:\Python27\Scripts>CodescapeConsole sp138
                    Welcome to Codescape Console 8.3.0.30. Enter help()<enter>
 for help
                    <tab> completion has been enabled.
                    Identifier SysProbe 00138
                    Firmware   1.5.2.0
                    Location
                    Mode       uncommitted
                    TCK Rate   31250kHz
```

Running Codescape Console on a probe

The instructions below assumed you are logged in to the probe and these instructions are issued from a command line on the probe. Codescape Console should be on your user PATH. If you are going to issue target commands, the probe be connected to a target. See *SP55E Connections* on page 38.

Start Codescape Console and connect to the probe using the command `probe ('sp localhost')`

For example:

```
sysprobe@IMG-SP58001:~$ CodescapeConsole
Welcome to Codescape Console 8.5.1.8. Enter help()<enter> for help
<tab> completion has been enabled.
>>> probe('sp localhost')Identifier SysProbe 58001Firmware
            2.3.0.0Location
            127.0.0.1Mode
            autodetectedTCK Rate
            20833kHz
[s0c0v0]>>>
```

Checking and reflashing SP55E firmware

This topic describes how to check for available firmware and reflash the firmware on a SysProbe. The firmware loaded on a SysProbe can be reflashed via Codescape Console or Codescape Debugger.

Note: The current firmware version is displayed when you connect to a target with Codescape Console.

From Codescape Debugger you can right-click on the target pain and select 'Reflash Firmware'.

MIPS technical support will notify customers when updated firmware is available.

The updates are sent as a single .fsh file.

More information on Codescape Console can be found in the Codescape Online Help.

You must be connected to the debug adapter via Codescape Console to carry out the following instructions. See *Connecting to a probe from Codescape Console* on page 46

Checking available firmware

Codescape Console has a command, `firmwarelist()`, that checks for compatible firmware available online.

For example:

```
>>> firmwarelist()
            0: 1.2.3 - SP55e - http://codescape.mips.com/components/probes/firmware/
sp01020300.fsh
            1: 1.2.1 - SP55e - http://codescape.mips.com/components/probes/firmware/
sp01020100.fsh
            2: 1.1.0 - SP55e - http://codescape.mips.com/components/probes/firmware/
sp01010000.fsh
            3: 1.0.0 - SP55e - http://codescape.mips.com/components/probes/firmware/
sp01000000.fsh
```

Reflashing a SysProbe

This topic describes how to reflash the firmware on a SysProbe.

To reflash with the latest available flash image, use the `firmwareupgrade()` command with no parameter. For example::

```
>>> firmwareupgrade()
               100% - Waiting for probe to restart
               Identifier SysProbe 00155
               Firmware   1.2.3.0
               Location   192.168.154.55
               Mode       uncommitted
               TCK Rate   31250kHz
```

To reflash with a specific flash image, supply the url in the `firmwareupgrade()` command. For example:

```
>>> firmwareupgrade("http://codescape.mips.com/components/probes/firmware/
sp01020100.fsh")
```

Note: Reflashing is complete when the prompt is displayed again. Do not disconnect or power down the probe until the prompt appears.

Connecting to MIPS development boards

This section describes how to connect MIPS probes to specific MIPS development boards.

Malta + coreFPGA6

The Malta board is supplied pre-flashed with YAMON. Please refer the YAMON documentation supplied (also available at *http://wiki.prplfoundation.org/wiki/MIPS_documentation*). YAMON stdout and terminal interface by default uses UART0 as shown on the picture below. YAMON outputs at a UART speed of 38400 baud 8N1. When Linux is running this may be different depending on kernel config, if you connect a terminal to UART0 be aware that the terminal baud speed must match whatever baud rate is set by Linux.

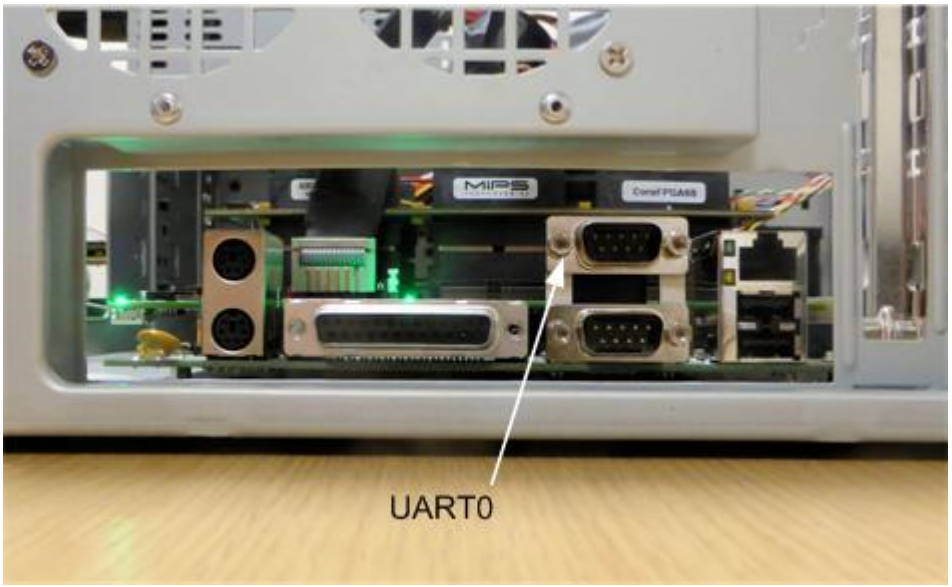


Figure 10: Location of UART0 on Malta

Make sure everything is powered off before connecting up.

The ribbon cable has a red line indicating the position of pin 1 and should be oriented as shown.

Figure 11: Connecting up a Malta + coreFPGA6 development board

SEAD3 Connection

This figure displays connection of the probe to a SEAD3 board.

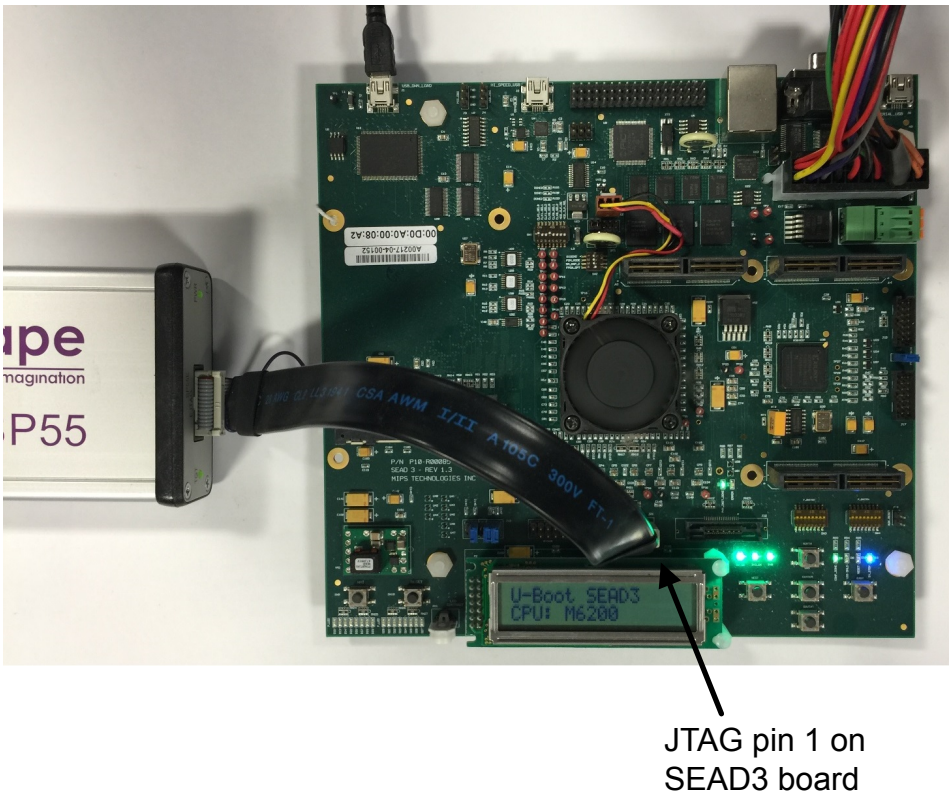


Figure 12: Connecting up a SEAD3 + coreFPGA6 development board

Diagnostics introduction

This section describes how to get diagnostics information on the state of the target and the host-probe connection to the target.

Diagnostics in Codescape

In Codescape there is a diagnostics report available from the Target Diagnostics window (Help menu > Diagnostics). Select 'Codescape Debugger > Comms Log' from the expandable tree on the left side.

In this report information is given about the debug adapter you are connected to and communications between the host PC, the debug adapter and the target.

If you are having difficulties connecting to the debug adapter or target you can use this information to diagnose the problem or you can copy and send the report using the 'Feature Request/Defect Reporting' option from the Help menu.

Listing probe transaction logs

A log of debug adapter transactions can be generated from Codescape Console using the `logfile` command. This can display error and transaction logs for the connected debug adapter.

Using the command without parameters prints a list of the available logs. For example:

```
>>> logfile()
   DA Info Log
   DA Error Log
   DA Verbose Log
   DA JTAG Log
```

Note: The logs available will depend upon the target and the type of probe (debug adapter) used.

Using a log name as a parameter prints the contents of that log. For example:

```
>>> logfile("DA Info Log")
   0.000:SoC  X:Generic : <info> : main                    : Initialising...
   0.000:SoC  X:Generic : <info> : setup                   :          Xilinx Configuration OK,
 VHDL version = 1.C
   0.000:SoC  X:Generic : <info> : board_init              : Board Revision 3
   0.103:SoC  X:Generic : <info> : main                    : Dash ID - 01EGNT33000401
   0.104:SoC  X:Generic : <info> : target_handler          : Dash Initialised, waiting for
 first host command to set operating mode......
   0.105:SoC  X:Generic : <info> : add_event               : add_event, add: id: 0, period: 1,
   30.823:SoC  X:Generic : <info> : jtag_scan               :   first command is JTAG Scan,
 entering Passive Mode !!
```