

# SR192A Self Test Routines

The purpose of this document is to describe the SR192A self test routines and their application.

## 1 RAM Test Algorithm

The SR192A Self Test Routines use a march pattern to test the RAMs described below.

1. Background fill memory range with hex 5s.
2. For each memory address in the range verify the background pattern (5) and write the compliment hex A.
3. For each memory address in the range verify the complimented pattern (A).

Two RAM test routines are used to specify address range to be tested, full and short. The full RAM test will access every memory cell in the address range. The short RAM test will only access the memory cells at each address boundary within the specified range. For example if the specified address range is hex 1000 to hex 8000 then the full RAM test would access each memory cell from 1000 to 8000, i.e., 1000, 1001, 1002, 1003, 1004, etc. The short RAM test would access 1000, 1001, 1002, 1004, 1008, etc.

## 2 Self Test Routines

The SR192A provides the following self test routines:

- Power On
- Word Serial Full RAM Test command
- Message Based “\*TST?” command
- VXI Plug&Play “**tasr192a\_self\_test**” function
- VXI Plug&Play “**tasr192a\_executeBit**” function

The following sections will describe each routine and how they would be applied in a user application.

### 2.1 Power On

The power on test is initiated by any of the following methods:

- Power is applied to the SR192A via the VXI backplane.
- Hard Reset (VXI Backplane RST pin low pulse).
- Soft Reset (Bit 0 of the SR192A VXI control register, A16 offset 4).

The power on test is executed in two steps. The CPU test and the Module test.

#### 2.1.1 CPU Test

The CPU Test verifies and initializes the CPU RAM and the VXI configuration registers.

If the CPU test passes, then the “Passed” bit in the SR192A VXI status register (A16 offset 4) is set high and the Module test is executed. If the CPU test fails, the “Passed” bit is set low and execution is halted.

#### 2.1.2 Module Test

The Module Test verifies each SR192A module by reading the module ID register. A module type specific tests is then performed. The results of the module test are set in the module status register. The following message based commands are used to query the module status register.

- “MODule:SElect <module>  
<module> = (TSA, TSB, DRA1, ..., DRA6, DRB1, ... DRB6, DAC)  
Select the specified module.
- “MODule:STATus?”  
Returns the selected modules status register. Upper eight bits are the module ID, lower eight bits are module type specific.

After selecting the module, send a system error query to verify that the module is installed.

- “SYStem:ERRor?”  
Should return (0, “No Error”)

The VXI Plug&Play function “**tasr192a\_queryModuleStatus**” can also be used to return the module status register.

#### **2.1.2.1 Timing Module**

- Re-verify the Module ID
- Full RAM test on the timing set memory
- Short RAM test on the last word memory.

If the timing module test passes then bit 0 of the module status register will be set to a 1.

#### **2.1.2.2 Dynamic I/O Module**

- Re-verify the Module ID
- Short RAM test on each of the five memories.

If the dynamic I/O module test passes then bit 0 of the module status register will be set to a 1.

#### **2.1.2.3 Static I/O Module**

- Re-verify the Module ID

If the static I/O module test passes then bit 0 of the module status register will be set to a 1.

#### **2.1.2.4 Accessory Module**

- Re-verify the Module ID
- Test for SR211 probe and run internal SR211 BIT and short RAM test on probe memory if installed.

If the accessory module test passes then bit 0 of the module status register will be set to a 1. If a SR211 probe pod is connected then bit 1 of the status register will set to a 1 and bit 2 indicated the SR211 BIT pass/fail result (1 = pass).

## **2.2 Word Serial Full RAM Test command**

---

The word serial full RAM test command performs the same function as the Power On, Module Test except that instead of using the short RAM test, it uses the full RAM test algorithm.

Perform the following steps to initiate the full RAM test:

1. Write a hex 1 to the SR192A VXI Data Low register, A16 offset 14.
2. Poll the SR192A VXI Response register, A16 offset 10 until bit 10 is set high.
3. Read the SR192A VXI Data Low Register, If the test passes a hex 8000 code will be returned, otherwise a 1 will be returned. A module that does not pass the full RAM test will set bit 0 of its module status to a zero. Each module take approximately 3 to 7 seconds for a full RAM test depending on the specific module.

## **2.3 Message Based “\*TST?” command**

---

The message based “\*TST?” command performs the same function as the Power On, Module Test.

The following table describes the contents of the returned result.

*TST? Results																															
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
																0	D R B 6	D R B 5	D R B 4	D R B 3	D R B 2	D R B 1	D R A 6	D R A 5	D R A 4	D R A 3	D R A 2	D R A 1	T S B	T S A	D A C

Bits 0 through 14 indicates whether the corresponding module passed (0) or failed (1). Bits 16 through 23 contain the SR211 BIT test results; 0 = pass, 1 = fail:

- Bit # Test**
- 16 Memory
  - 17 Node
  - 18 LED
  - 19 EEPROM
  - 20 Switch
  - 21 Pulse
  - 22 Comparator
  - 23 DAC

## 2.4 VXI Plug&Play “tasr192a\_self\_test” function

The VXI Plug&Play self test function uses the “\*TST?” command described above. The results are returned as a number and a message. Refer to the VXI Plug&Play help for the return values and messages.

## 2.5 VXI Plug&Play “tasr192a\_executeBit” function

The SR192A VXI Plug&Play driver includes a function to execute a Built in Test function (BIT) on the Timing and I/O modules. There are two built in tests, Standard and Intermediate. Refer to the VXI Plug&Play help for the full description of the return value.

### 2.5.1 Standard BIT

The Standard BIT uses special sequencers on the Timing and I/O modules to test motherboard and inter-module signal connectivity.

The following logic/signals are tested:

Logic/Signals Tested	Returned Error Numbers
Timing Module TSA/TSB Idle, Reset and Link Modes	1-7
TSES 1-6 Signal Connectivity	8
FMA and Address Control Signal Connectivity	9-14
Inter Module board signal connectivity	15-18
Serial/Increment Mode signal connectivity	19

The standard BIT test does not drive any of the I/O module channels.

### 2.5.2 Intermediate BIT

The Intermediate BIT programs two pattern sequences to test the driver/receiver, I/O memory and error counter logic.

The following logic is tested:

Logic Tested	Returned Error Numbers
Timing Module Master Mode Idle, Reset and Link Modes	20-22
Pattern Sequence 1, Driver/Receiver Tristate	23-26
Pattern Sequence 2, Driver/Receiver Error Count	27-30

### 2.5.2.1 Pattern Sequence Timing

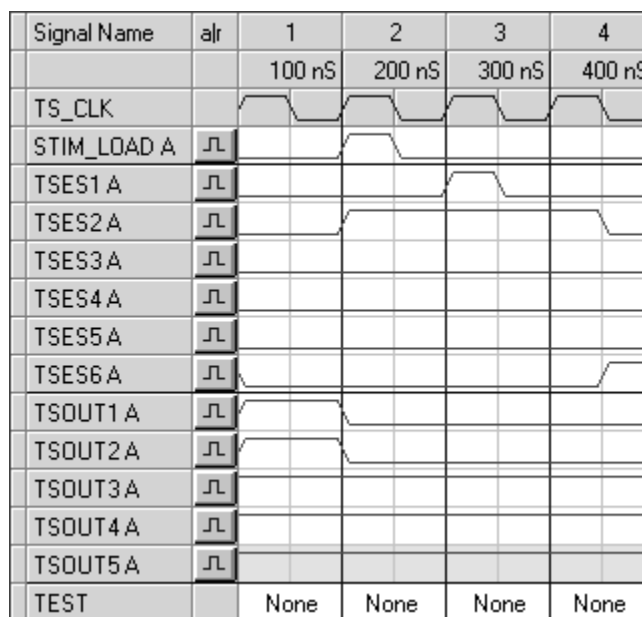
Both pattern sets use the same timing set;

Response/Record Strobe = TSES1

Output Enable = TSES2

Error Strobe = TSES6

4 cells 10MHz



### 2.5.2.2 Pattern Sequence 1 Data

SR214 Levels, VOH = 5V, VOL = -5V, VIH = 2V, VIL = -2V.

Pattern sequence 1 uses a two word table.

Word	Output	Tristate	Expect	Mask
1	0xFFFF	0x0000	0xFFFF	0x0000
2	0x0000	See note 1	See note 1	0x0000

Note 1: Tristate/expect programmed for each channel being tested (walking 0).

Pattern sequence 1 is used to test the driver/receiver logic as well as shorts on any of the I/O channels.

### 2.5.2.3 Pattern Sequence 2 Data

SR214 Levels, VOH = 5V, VOL = -5V, VIH = 2V, VIL = -2V.

Pattern sequence 2 uses an eighteen word table.

<b>Word</b>	<b>Output</b>	<b>Tristate</b>	<b>Expect</b>	<b>Mask</b>
1	0x0000	0x0000	0x0000	0x0000
2	0x0001	0x0000	0x0101	0x0001
3	0x0002	0x0000	0x0202	0x0002
4	0x0004	0x0000	0x0404	0x0004
5	0x0008	0x0000	0x0808	0x0008
6	0x0010	0x0000	0x1010	0x0010
7	0x0020	0x0000	0x2020	0x0020
8	0x0040	0x0000	0x4040	0x0040
9	0x0080	0x0000	0x8080	0x0080
10	0x0100	0x0000	0x0101	0x0100
11	0x0200	0x0000	0x0202	0x0200
12	0x0400	0x0000	0x0404	0x0400
13	0x0800	0x0000	0x0508	0x0800
14	0x1000	0x0000	0x1010	0x1000
15	0x2000	0x0000	0x2020	0x2000
16	0x4000	0x0000	0x4040	0x4000
17	0x8000	0x0000	0x8080	0x8000
18	0x0000	0x0000	0x0000	0x0000

Pattern sequence 2 is used to test the driver/receiver error counter memory/logic.