

Track Finder Crate: Configuration and Testing

Proposal

Track Finder Crate Architecture

The Track Finder (TF) Crate houses a set of dedicated 9U x 400 VME64x cards:

- 12 Sector Processor (SP) cards;
- Clock and Control Board (CCB);
- Muon Sorter (MS) card.

All cards are A16D16 Slaves and feature the extensive use of programmable devices that comply with the following IEEE standards:

- IEEE Standard Test Access Port and Boundary-Scan Architecture¹ (IEEE 1149.1);
- IEEE Standard for Boundary-Scan-based In System Configuration of Programmable Devices² (IEEE 1532)

The TF VME bus is based on J1-only version of the VME64x backplane³, which utilizes 160-pin 5-row connectors. All communications between TF modules are performed over custom backplane, which is located below J1 VME backplane. The number of connectors per slot and number of pins per connector vary from slot to slot.

IEEE Standards

The essence of the IEEE 1149.1 is that circuitry that may be built into an integrated circuit to assist in the test, maintenance, and support of assembled printed circuit boards, as defined in the standard. The circuitry includes a standard interface through which instructions and test data are communicated. A set of test features is defined, including a boundary-scan register, such that the component is able to respond to a minimum set of instructions designed to assist with testing of assembled printed circuit boards.

The IEEE 1532 standard is backward compliant with the IEEE 1149.1 Test Access Port (TAP) and state machine. It describes a series of mandatory and optional boundary-scan instructions and associated data registers that define a standard methodology for accessing and configuring programmable devices that already support IEEE 1149.1.

Both standards use the same four dedicated pins, called JTAG pins, for communicating with the IEEE 1149.1 compliant devices:

- TCK – Test Clock input
- TMS – Test Mode Select input
- TDI – Test Data Input
- TDO – Test Data Output

Hardware Implementations of Boundary Scan:

The on-board programmable and other IEEE 1149.1-compliant devices, like memories, are grouped into one or more JTAG chains. The chains end with the JTAG ports that allow both device configuration (IEEE 1532) and board-level hardware testing and maintenance (IEEE 1149.1).

For the SP02, for example, at least 3 chains are required:

- The first chain connects 5 Front FPGAs, DDU FPGA, VME FPGA and 7 EEPROMs (one per each FPGA);
- The second chain connects Main FPGA and 4 EEPROMs on the Mezzanine Card;
- The third chain (optional) connects 48 LUTs – to facilitate board testing.

Ideally, the JTAG port access should be available right from the moment the board is assembled and powered for the first time. Furthermore, we probably would like to keep JTAG access available all the time for board reconfiguration, test and maintenance, especially if the board experiences some hardware / firmware problems. There are several options how this can be achieved.

Solution 1: JTAG-Connector on every board

Advantages:

- It is easy to implement, especially if there is a free space on the front panel (which is questionable in case of SP02). Each chain requires a separate JTAG connector, or the jumpers should be foreseen to switch the same connector to control different chains.
- It is flexible – one can change cards and slots in the way he wants.
- It is possible to add / remove JTAG support for cards without disturbing the system.

Disadvantages:

- The design is limited: only one board under test at a time
- JTAG cables have to be switched all the time from one JTAG port to another.

Solution 2: Use a Boundary-Scan Controller on every board

It could be any of available test-bus/boundary-scan/ controllers/masters with either an 8-bit (Texas 74LVT8980, Fairchild SCANPSC100F, or Agere 497AE) or a 16-bit (Texas 74ACT8990, Agere 1215E) interface. 74ACT 8990 and SCANPSC100F devices are suitable for the 5V applications, while 74LVT8990 and Agere devices are designed for the 3.3V applications.

Advantages:

- Flexible and comfortable board switching.
- It is possible to configure and test the whole crate automatically.
- Remote configuring and testing is possible.

Disadvantages:

- Big hardware overhead: access to the controller requires a secondary VME interface on the board, since the primary VME interface is implemented in the FPGA, which needs to be configured and debugged itself first
- Big software overhead: different software libraries for each card type. It is very unlikely that card designers would use the same addressing scheme for different designs.

Solution 3: Integrate a Boundary-Scan Controller into a chip on every board**Advantages:**

- See Solution 2

Disadvantages:

- Because the Controller is integrated into a chip that very chip cannot be configured and tested.
- The integrated controller requires considerable efforts to implement.

Solution 4: Use a VME64x Test and Maintenance Bus to get access to every board in the crate.

The VME64 Extensions Standard⁴ defines a Test and Maintenance Bus (T&Mbus) and allocates 5 pins on the P1/J1 connector for it:

Rule 3.11:

The 14 RsvBus pins, the 5 test and maintenance bus (**MPR, MCLK, MSD, MMD & MCTL**) pins and the RESP* pin **shall be bused and terminated** on VME64x backplanes that implement a 160 pin connector for J1 in the same fashion and have the same rules as the other VME64 bused signals defined in the VME64 Standard.

Rule 3.13:

The five test and maintenance bus signal lines (MPR, MCLK, MSD, MMD & MCTL) shall be reserved for specification by the IEEE 1149.5 MTM-Bus Standard.

Here:

- MCLK - IEEE 1149.5 MTM-Bus Clock - A three-state driven signal that is used to carry the T&Mbus clock signal.
- MCTL - IEEE 1149.5 MTM-Bus Control - A three-state driven signal that is used to carry the T&Mbus control signal.
- MMD - IEEE 1149.5 MTM-Bus Master Data - A three-state driven signal that is used to carry the T&Mbus master data signal information.
- MPR - IEEE 1149.5 MTM-Bus Pause Request - A three-state driven signal that is used to carry the T&Mbus pause request signal information.
- MSD - IEEE 1149.5 MTM-Bus Slave Data - A three-state driven signal that is used to carry the T&Mbus slave data signal information.

The IEEE Standard for Module Test and Maintenance Bus (MTM-Bus) Protocol⁵ (IEEE 1149.5) specifies a serial, backplane, test and maintenance bus (MTM-Bus) that can be used to integrate modules from different design teams or vendors into testable and maintainable subsystems. Physical, link, and command layers are specified. Standard interface protocol and commands can be used to provide the basic test and maintenance features needed for a module **as well as access to on-module assets (memory, peripherals, etc.) and IEEE 1149.1 Boundary-Scan**. Standard commands and functions support fault isolation to individual modules and test of backplane interconnect between modules.

As one can easily figure out that the T&Mbus signals follow the JTAG port signals:

Table 1 T&Mbus and JTAG signal correspondence.

No	JTAG	T&Mbus
1	TCK	MCLK
2	TMS	MCTL
3	TDI	MMD
4	TRST*	MPR
5	TDO	MSD

TRST is an optional JTAG Test Reset Input signal.

The TI 74LVT8996 ASP (10-bit Addressable Scan Ports / Multidrop-Addressable TAP Transceivers) offers a hardware support for the T&Mbus usage.

Conceptually, the ASP is a simple switch that can be used to directly connect a set of multidrop primary TAP signals to a set of secondary TAP signals – for example, to interface backplane TAP signals to a board-level TAP. The ASP provides all signal buffering that might be required at these two interfaces. When primary and secondary TAPs are connected, only a moderate propagation delay is introduced – no storage/retiming elements are inserted. This minimizes the need for reformatting board-level test vectors for in-system use.

When connected, the primary test data input (PTDI) and primary test mode select (PTMS) input are buffered onto the secondary test data output (STDO) and secondary test mode select (STMS) output, respectively, while the secondary test data input (STDI) is buffered onto the primary test data output (PTDO). When disconnected, STDO is at high impedance, while PTDO is at high impedance, except during acknowledgment of a shadow protocol. Upon disconnect of the secondary TAP, STMS holds its last low or high level, allowing the secondary TAP to be held in its last stable state. Upon reset of the ASP, STMS is high, allowing the secondary TAP to be synchronously reset to the Test-Logic-Reset state.

In system, primary-to-secondary connection is based on shadow protocols that are received and acknowledged on PTDI and PTDO, respectively. These protocols can occur in any of the stable TAP states other than Shift-DR or Shift-IR (i.e., Test-Logic-Reset, Run-Test/Idle, Pause-DR or Pause-IR). The essential nature of the protocols is to receive/transmit an address via a serial bit-pair signaling scheme. When an address is received serially at PTDI that matches that at the parallel address inputs (A9–A0), the ASP serially retransmits its address at PTDO as an acknowledgment and then assumes the connected (ON) status, as above. If the received address does not match that at the address inputs, the ASP immediately assumes the disconnected (OFF) status without acknowledgment.

The ASP also supports three dedicated addresses that can be received globally (that is, to which all ASPs respond) during shadow protocols. Receipt of the dedicated disconnect address (DSA) causes the ASP to disconnect in the same fashion as a nonmatching address. Reservation of this address for global use ensures that at least one address is available to disconnect all receiving ASPs. The DSA is especially useful when the secondary TAPs of multiple ASPs are to be left in different stable states. Receipt of the reset address (RSA) causes the ASP to assume the reset condition, as above. Receipt of the test-synchronization address (TSA) causes the ASP to assume a connect status (MULTICAST) in which PTDO is at high impedance but the connections from PTMS to STMS and PTDI to STDO are maintained to allow simultaneous operation of the secondary TAPs of multiple ASPs. This is useful for multicast TAP-state movement, simultaneous test operation (such as in Run-Test/Idle state), and scanning of common test data into multiple like scan chains. The TSA is valid only when received in the Pause-DR or Pause-IR TAP states.

Figure 1 shows typical crate architecture.

There is only one Test Bus Controller in the crate, which occupies one slot and connects to the 160-pin J1 connector. Each card that needs to be configured /tested uses an ASP chip to provide access to the IEEE 1149.1 compliant device chain. If the card carries more than one device chain, the corresponding number of ASP chips per board could be used. A 5-bit geographical address, implemented in VME64x, provides a base ASP address, while lower 5 address bits can be used at designer's discretion.

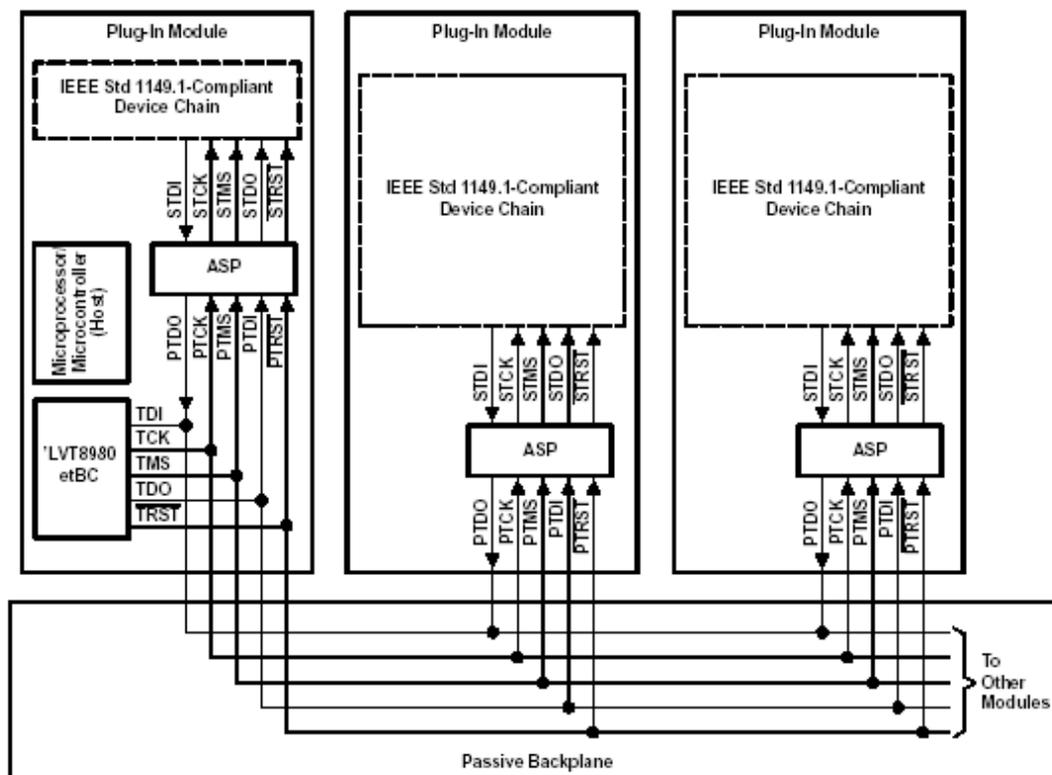


Figure 1 Backplane application with centralized test control architecture.

Advantages:

- The T&Mbus and its termination are already placed on the VME64x-compatible J1/P1 backplane.
- The card overhead is limited to one ASP (74LVT8996) 3.3.V device.
- It is possible to configure and test the whole crate automatically.
- Remote configuring and testing is possible.
- Access does not depend on the VME interface, so every chip can be configured and tested.
- Configuration and Test access to every card in the crate is unified. Only one VME library is needed to communicate with the Test Bus Controller

Disadvantages:

- Scan Bus Controller should be either purchased⁶ or designed, but it should have been designed anyway in each card.

Conclusion:

Utilization of T&Mbus in the TF crate design gives much more advantages than drawbacks.

¹ Available at [IEEE Std 1149.1-1990 - Description](#)

² Available at [IEEE Std 1532 - Description](#)

³ CSC Track Finder Crate Specification, Draft 12/17/2001, available at [UFL](#)

⁴ ANSI/VITA 1.1-1997 American National Standard for VME64 Extensions

⁵ Available at [IEEE Std 1149.5-1995 - Description](#)

⁶ [ETW: Abstract: A System Level Boundary Scan Controller Board for VME Applications](#)