VME2E: VME TO ETHERNET – COMMON HARDWARE PLATFORM FOR LEGACY VME MODULES

Y. Tian†, NSLS-II, Brookhaven National Laboratory*, Upton, New York, USA
J. Jamilkowski, Electron-Ion Collider, Brookhaven National Laboratory, Upton, New York, USA

Abstract

The VME architecture, developed in the late 1970s, has been a reliable control system platform for decades. Today, it faces challenges including throughput, computing power, the operating system, and obsolescence of legacy modules. Next-gen platforms like ATCA and microTCA require significant changes but may not always be suitable for legacy upgrades.

In this paper, we introduce VME2E (VME to Ethernet), an open-source hardware platform for replacing legacy VME modules without disassembling the system. VME2E retains the VME form factor and utilizes stable power and cooling. It features a Xilinx FPGA system-on-module (SOM) with GigE for high-speed computing and communication, along with a high pin count (HPC) FPGA mezzanine connector (FMC) for IO support. VME2E is designed as a low-cost, open-source solution for VME legacy upgrades.

INTRODUTION

The VME architecture [1, 2, 3], originally introduced in the late 1970s, has established itself as a robust hardware platform for controlling systems over the past four decades. VME, short for 'VERSAbus Module Eurocard,' was developed to serve various industrial, commercial, and military applications. The standard specifies both electrical and mechanical specifications.

The VME bus follows a master-slave computer architecture. Notably, its signalling scheme is asynchronous, meaning that data transfer is not tied to the timing of a bus clock. Additionally, a single VME crate can accommodate up to 21 modules.

However, today, the VME hardware platform faces the following four significant challenges.

Backplane Throughput

In a standard VME chassis, modules communicate with each other using shared bandwidth. Over the past four decades, the throughput of the VME backplane has been improved multiple times, including:

- VMEbus, introduced in 1981 with a throughput of 40MB/s.
- VME64, introduced in 1994 with a throughput of 80MB/s.
- VME64x, introduced in 1997 with a throughput of 160MB/s.
- VME320/2eSST, introduced in 1997 with a throughput of 320MB/s.

However, as a shared bus, the VME backplane's throughput is inherently limited. The VME protocol is based on master-slave communication, where the backplane is shared by all VME boards, essentially operating sequentially.

Today, each module can have a Gigabit Ethernet port to send data or communicate with external systems using high-speed fiber links. The shared nature of the VME backplane restricts the overall chassis throughput.

FEC Computing Power

In a typical VME chassis, the Front-End Controller (FEC) handles most of the computing. As computing power has improved, so has the performance of VME FECs. Here are some snapshots of the history of VME FECs:

- 1979: Motorola 68000 (m68k): 68k transistors, 3.5µm technology, 4-16.67MHz.
- 1988: MVME147 MC68030: 16-33MHz, 4K SRAM, 32MB DRAM.
- 1992: MVME162, 1.6 million transistors, 90-120MHz, 0.5µm technology.
- 2010: MVME3100, PowerPC MPC8540, 1GHz, 500MB DDR, 2eSST.

However, today, each module can host a powerful computing unit, such as a microprocessor or FPGA. Concentrating all the computing power solely on the FEC has its limits.

Operating System

Due to its centralized architecture, where all VME modules share computing power on the FEC and communication bandwidth on the backplane, it is often necessary to run a real-time operating system (OS) on the FEC. Over the years, several real-time operating systems have been developed to run on VME FECs, including VxWorks/Tornado (from WindRiver Systems), RTEMS (from OAR Corp), LynxOS (from Lynx Real Time Systems), OS-9 (from MicroWare), and SPECTRA/VRTX (from Microtech Research).

However, supporting the real-time OSs is becoming increasingly challenging, and many have ceased development.

* Work supported by NSLS-II, BNL, US Department of Energy
† Email address: ytian@bnl.gov

UPGRADE OPTIONS

ATCA/microTCA/AMC

Today, when designing a new chassis system, we have various options such as the Advanced Telecommunications Computing Architecture (ATCA) chassis, MicroTCA chassis, and Advanced Mezzanine Card (AMC) cards integrated into ATCA/microTCA systems. Like VME, ATCA/microTCA provides electrical, mechanical, thermal, and management specifications. ATCA/microTCA address the limitations of VME chassis.

From the backplane perspective, an ATCA/microTCA chassis' backplane serves as a switched fabric computer system, allowing each AMC card to communicate with others over a high-speed communication fabric, such as Gigabit Ethernet, 10G/40G Ethernet, or RapidIO. Each AMC card can possess varying computing capabilities, including CPUs, FPGAs, graphic cards with GPUs, storage, analog-to-digital and digital-to-analog conversion, clocking and triggering, FMC cards, and more. Each AMC card can function as an independent computing unit. Since all AMC cards can communicate with one another, the entire ATCA/microTCA system becomes a powerful distributed computing system with multiple computing units and interconnections.

For new system deployments or for legacy VME systems that can be entirely removed or replaced (with the necessary software migration to the new platform), ATCA/microTCA is an excellent choice. However, in cases where a legacy VME system cannot be completely removed or where only partial upgrades of specific VME boards are necessary, a less disruptive one-to-one in-place update option of legacy VME modules is needed.

NSLS-II VME Upgrade Case

As a third-generation light source, NSLS-II at Brookhaven National Laboratory was commissioned in 2014. When the machine was built, there were few VME chassis systems in use. These systems were sourced from external vendors, including:

- 1MSPS and 200MSPS ADC boards from the Budker Institute of Nuclear Physics (BINP).
- 100MSPS ADC board from Struck Innovative System.
- Timing systems from Micro-Research Finland (MRF), including VME-EVG-230 and VME-EVR-230.

All the VME boards were running the RTEMS operating system, which is no longer supported by the community. For the timing system, there was one VME-EVG-230 card and over 30 VME-EVR-230 boards. We migrated the EPICS driver from RTEMS running on the Front-End Controller (FEC) to an Ethernet-based driver. Each VME-EVR-230 now has an Ethernet port, and MRF enables TCP/IP socket communication on this port, allowing the VME-EVR-230 to communicate with an external EPICS IOC using TCP/IP sockets. For all NSLS-II EVRs, RTEMS and FEC are no longer required.

Regarding the three ADC cards, they have become obsolete. Since we only use a few of these cards, and they are installed in the system with dedicated cables, upgrading these boards involves an in-place replacement of the existing VME cards with boards of the same form factor. This way, we only need to replace the VME board without replacing the entire VME chassis. The VME chassis, including its power supply, cooling fans, and management, has proven to be very reliable and does not require an upgrade. Therefore, the ideal upgrade option is to replace the obsolete ADC board with another board that has a VME form factor but does not require the backplane for data communication, FEC, or RTEMS OS.

VME2E MODULE

Considering the points mentioned above, we designed a new VME2E module (abbreviated for VME-to-Ethernet) with the following requirements:

- Utilize the VME form factor for one-to-one replacement, or it can also be packaged as a standalone module with a separate power and ground connector.
- Incorporate a VME connector to provide mechanical support and deliver power and ground.
- Equip each module with its Ethernet port for communication.
- Include a high-speed fiber communication (SFP) port for point-to-point communication in each module.
- Ensure that each module possesses its computing power, including a CPU or FPGA.

Instead of designing different PCBs for various legacy ADC boards, the VME2E module should serve as a common hardware platform for other legacy VME board upgrades. Lastly, VME2E should be an open-source hardware design, allowing individuals to use it for their own legacy VME module upgrades. Figure 1 shows the board layout. As depicted in the figure, there are two VME connectors that provide power and ground only. There is one SOM connector to host an FPGA SOM, and a high pin count (HPC) FMC connector to accommodate various FPGA FMC boards.

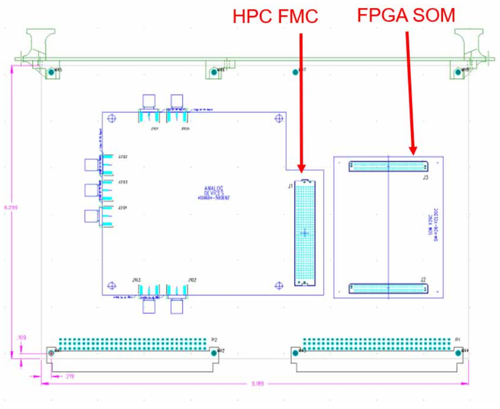

Figure 1: VME2E layout.

ISBN: 978-3-95450-238-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2023-WE1BC001

FPGA System-on-module (SOM)

As a common hardware platform, VME2E needs to be flexible and configurable for various applications. A natural choice is to use an FPGA. We selected Xilinx's latest FPGA System-on-Module (SOM), the Kria K26 [4], as the heart of VME2E, as shown in Fig. 2. The reason for choosing an FPGA SOM instead of directly using an FPGA chip on the VME2E PCB is to avoid reinventing the wheel, specifically the complicated PCB layout for components commonly found on an FPGA (FPGA, DDR, flash, power management, etc.). The SOM provides a reliable and cost-effective solution with long-term hardware guarantees from a commercial vendor. We chose the Xilinx Kria K26 because it features the latest ultra-scale FPGA chip, and Xilinx guarantees its availability for 10+ years. This SOM has a large user community with various applications. Furthermore, the future Kria SOMs, such as the K24 and high compute Kria SOM, will be connector-pin compatible. The features of the Kria K26 include:

- Low cost (~$300)
- 4GB DDR4 for waveform storage
- 256K logic cells
- 64-bit quad-core arm A53
- 1Gb to 40Gb Ethernet
- 26.6Mb SRAM
- 4 12.5Gb/s GTH transceiver
- 1248 DSP48 blocks
- 1.4 TOPS (Trillions of Operations per Second)
- 245 user IO

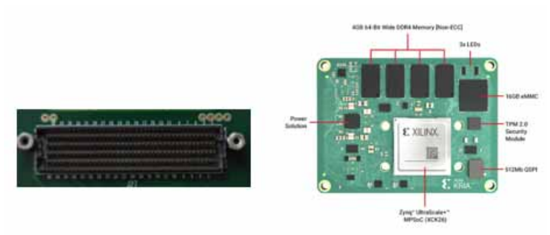

Figure 2: HPC (left) and Kria K26 FPGA SOM (right).

HPC FMC

FPGA Mezzanine Card (FMC) [5] is an ANSI/VITA standard that defines I/O mezzanine modules with connections to an FPGA. It specifies a low-profile connector and compact board size for compatibility with several industry-standard slot cards, blades, low-profile motherboards, and mezzanine form factors. It defines the features of the pins, such as power, ground, differential signals, high-speed transceivers, and JTAG, among others. There are two types of FMC connectors: low pin count (LPC) and high pin count (HPC). To ensure wider compatibility, VME2E uses an HPC FMC connector to allow a wide range of FMC cards to be plugged onto the board, as shown Fig. 2.

For the upgrade of NSLS-II legacy VME boards, we have selected the EVAL-AD9684 (14-Bit 500MSPS Dual ADCs) FMC module for the ADC cards. In future high-speed feedback applications, the VME2E will be packaged as a standalone 1U chassis with an FMC card containing ADC and DAC chips, such as AD-FMCDAQ2-EBZ and AD-FMCDAQ3-EBZ. FMC modules have a large ecosystem, and the VME2E can serve various functions with different FMC cards and FPGA firmware.

STATUS

The VME2E design commenced in April 2023. The schematic design is complete, and the PCB layout is currently in progress. To adapt to various applications, many design details have been implemented. All the different traces must have matched lengths and impedances, the high-speed transceiver clock pins must align with the pins on the FPGA SOM, and logic-level shift chips are used to ensure FMC compatibility with different logic-level standards. In addition, the FPGA firmware design (targeting NSLS-II's ADC board upgrade with EVAL-AD9684) and the EPICS IOC design are in progress. Once the PCB design is completed and released, we will undergo the approval procedure to make the VME2E an open-source hardware design.

SUMMARY

We report a one-to-one in-place upgrade solution for legacy VME boards. An FPGA SOM is used to provide flexible control functions. A high pin count (HPC) FMC connector is used to host various FMC modules for various functions. The board has a Gigabit Ethernet connector and four high-speed fiber links for communication. The board will be aimed at being an open-source hardware design.

REFERENCES

[1] J. Black, "*The System engineer's handbook: a guide to building VMEbus and VXIbus systems*", Morgan Kaufmann. 1992, p. 563.

[2] "VMEBUS TECHNOLOGY", `https://www.vita.com/VMEbus-FAQ`

[3] "Open Standards", `https://www.picmg.org/open-standards/`

[4] "Kria K26 System-on-Module", `https://www.xilinx.com/products/som/kria/k26c-commercial.html`

[5] "FPGA Mezzanine Card", `https://fmchub.github.io/appendix/VITA57_FMC_HPC_LPC_SIGNALS_AND_PINOUT.html`