# Combined CCB-VME Interface

Petersburg Nuclear Physics Institute / University of Florida
Proposal
February 12, 2002

## Summary

The current Sector Processor (SP) PCB layout assumes [1] that the Clock and Control Board (CCB) interface is implemented in the Main FPGA chip, while the VME Interface occupies its own FPGA. We would like to suggest an alternative approach, when both the VME and CCB interfaces are combines in one FPGA, while the Main FPGA is totally dedicated to its processing functions. There are several reasons for this:

- We would like to have most of the SP functions working even without the Mezzanine Card installed; The list of functions might include:
  - Configuring all main board FPGAs;
  - Writing to and reading back from all the internal FPGA registers;
  - Loading to and reading back from all the look-up memories;
  - Running test patterns into Front FPGAs;
  - Modeling optical link synchronization procedures;
  - Generating fake L1Accepts and debugging DDU interface;
- It would have never happen had the CCB Interface be implemented in the Main FPGA;
- This can only be achieved, if the Timing and Trigger Control (TTC) functions are implemented on the main SP board;
- Some of the TTC functions are interdependent with or duplicated by the corresponding VME functions, and the two should be combined by some logic anyway;
- The CCB clock should also be combined with the on-board oscillator to be independent of the CCB presence and facilitate the debugging process;
- The number of pins that would have required to implement the CCB interface plus CCB-VME signal exchange plus TTC outputs for the rest of the SP board could reach 34+10+ 8=52. This would have added to more than 640 pins needed to bring in and take out muon stubs and tracks, totaling this number to almost 700 – the highest number of required pins for the Mezzanine board amongst interested parties.

## CCB Interface

The CCB interface provides the SP with the timing and trigger control signals distributed by the Clock and Control Board (CCB) over the backplane. The SP CCB Interface [2] on

the backplane side numbers as many as 34 signal lines coming in and going out of the SP. The complete list of lines is presented in Table 1[1].

**Table 1: SP CCB Interface Signals.**

| Signal | Lines | Direction | Type | Logic | Duration |
|---|---|---|---|---|---|
| **Clock Bus** | | | | | |
| CCB_CLOCK40 | 2 | IN | Point-to-point | LVDS | 40MHz |
| CCB_CLOCK40_ENABLE | 1 | IN | Bussed | GTLP | Pulse, n counts |
| **Subtotal** | **3** | | | | |
| **Fast Control Bus** | | | | | |
| CCB_CMD [5..0] | 6 | IN | Bussed | GTLP | Level |
| CCB_EVCNTRES | 1 | IN | Bussed | GTLP | 25ns |
| CCB_BCNTRES | 1 | IN | Bussed | GTLP | 25ns |
| CCB_CMD_STROBE | 1 | IN | Bussed | GTLP | 25ns |
| CCB_BX0 | 1 | IN | Bussed | GTLP | 25ns+ECL FP |
| CCB_L1ACCEPT | 1 | IN | Bussed | GTLP | 25ns+ECL FP |
| CCB_DATA [7..0] | 8 | IN | Bussed | GTLP | Level |
| CCB_DATA_STROBE | 1 | IN | Bussed | GTLP | 25ns |
| CCB_RESERVED [3..0] | 4 | IN | Bussed | GTLP | |
| CCB_READY | 1 | IN | Bussed | GTLP | Static level |
| **Subtotal** | **25** | | | | |
| **Reload Bus** | | | | | |
| SP_HARD_RESET | 1 | IN | Bussed | GTLP | 300ns |
| SP_CFG_DONE | 1 | OUT | Point-to-Point | GTLP | Level |
| **Subtotal** | **2** | | | | |
| **Reserved Lines** | | | | | |
| SP_RESERVED [3..0] | 4 | OUT | Bussed | GTLP | |
| **Subtotal** | **4** | | | | |
| **Total** | **34** | | | | |

The other side of the CCB Interface addresses the SP itself. The interface performs the following tasks:

- Switches to on-board oscillator if detects that the CCB clock has been lost. Unconditional switching will be implemented under the VME control as well.
- De-skews the system clock between multiple Virtex devices all over the board.
- Fan-outs CCB timing control signals, like BX0, EVCNTRES, BCNTRES, and L1ACCEPT to every interested Virtex FPGA on the board.
- Fan-outs SP_HARD_RESET to every Virtex FPGA, and collects DONE and INIT signals from them to report to the CCB.
- Decodes and fan-outs other CCB commands, related to the SP timing control (to be specified).

Implementing of all of these functions may consume as many as 10 to 25 I/Os. Interaction with the VME will be implemented inside the FPGA and does not require any additional I/Os.

[1] Table 1 of reference [2] enumerates only 3 ccb_reserved lines instead of 4, while Figures 2-4 show 4 lines.

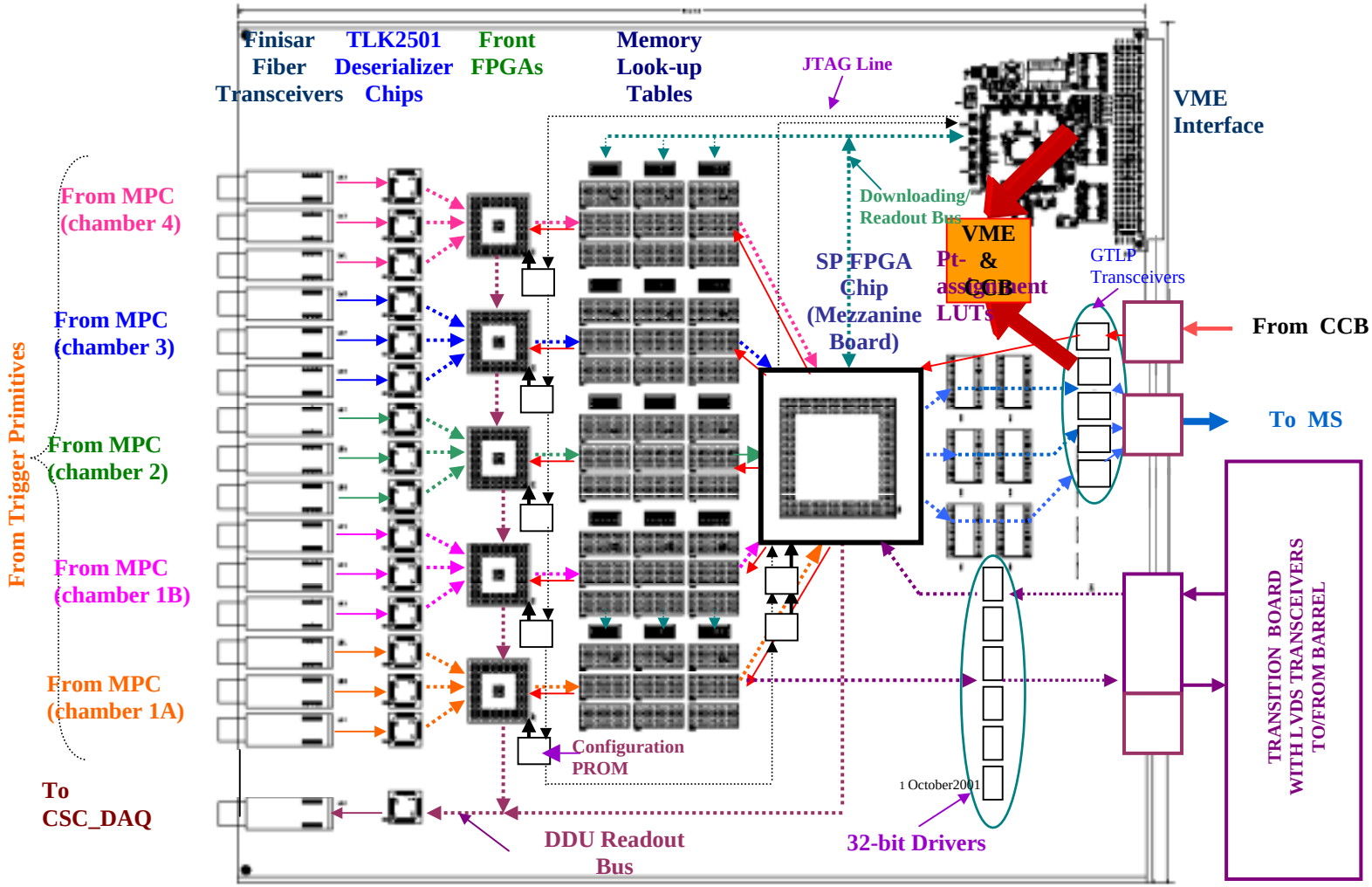


**Figure 1: SP Board Layout.**

# VME Interface

The SP implements an A24D16 Slave interface and complies with [3]. It responds to commands with the following address modifiers, see Table 2.

**Table 2: SP Address Modifier Codes.**

| AM | Description |
|---|---|
| 39 | A24 non-privileged data access |
| 3A | A24 non-privileged program access |
| 3D | A24 supervisory data access |
| 3E | A24 supervisory program access |
| 3B | A24 non-privileged block transfer (BLT) |
| 3F | A24 supervisory block transfer (BLT) |

It uses 5-bit geographical addressing to partition the A24 space, so A19 addresses is available for each SP. Table 3 lists all signals at P1/J1 VME connectors. Those used by the SP interface are emphasized in **Bold.** Total number of FPGA I/Os required equals to 59 (pairs IACKIN*/IACKOUT*, BG0IN*/BG0OUT*, … , BG3IN*/BG3OUT* - are simple jumpers on the board and do not occupy any FPGA pins). The total number of pins for the combined CCB-VME interface reaches so far 52 + 59. We have to add here CE* lines for each destination, which give us the total of no more than 130-150 I/Os. This count leads us either to VirtexE in PQ240 or FG256 package or again to Virtex-II in FG256 package.

**Table 3: VME64x P1/J1 Pin Assignment.**

| Pin | Row Z | Row A | Row B | Row C | Row D |
|---|---|---|---|---|---|
| 1 | MRP | **D00** | BBSY* | **D08** | VPC |
| 2 | **GND** | **D01** | BCLR* | **D09** | **GND** |
| 3 | MCLK | **D02** | ACFAIL* | **D10** | +V1 |
| 4 | **GND** | **D03** | **BG0IN*** | **D11** | +V2 |
| 5 | MSD | **D04** | **BG0OUT*** | **D12** | RsvU |
| 6 | **GND** | **D05** | **BG1IN*** | **D13** | -V1 |
| 7 | MMD | **D06** | **BG1OUT*** | **D14** | -V2 |
| 8 | **GND** | **D07** | **BG2IN*** | **D15** | RsvU |
| 9 | MCTL | **GND** | **BG2OUT*** | **GND** | **GAP*** |
| 10 | **GND** | SYSCLK | **BG3IN*** | SYSFAIL* | **GA0*** |
| 11 | RESP* | **GND** | **BG3OUT*** | BERR* | **GA1*** |
| 12 | **GND** | **DS1*** | BR0* | **SYSRESET*** | **+3.3V** |
| 13 | RsvBus1 | **DS0*** | BR1* | **LWORD*** | **GA2*** |
| 14 | **GND** | **WRITE*** | BR2* | **AM5** | **+3.3V** |
| 15 | RsvBus2 | **GND** | BR3* | **A23** | **GA3*** |
| 16 | **GND** | **DTACK*** | **AM0** | **A22** | **+3.3V** |
| 17 | RsvBus3 | **GND** | **AM1** | **A21** | **GA4*** |
| 18 | **GND** | **AS*** | **AM2** | **A20** | **+3.3V** |
| 19 | RsvBus4 | **GND** | **AM3** | **A19** | RsvBus11 |
| 20 | **GND** | **IACK*** | **GND** | **A18** | **+3.3V** |
| 21 | RsvBus5 | **IACKIN*** | SERCLK | **A17** | RsvBus12 |
| 22 | **GND** | **IACKOUT*** | SERDAT | **A16** | **+3.3V** |
| 23 | RsvBus6 | **AM4** | **GND** | **A15** | RsvBus13 |
| 24 | **GND** | **A07** | IRQ7* | **A14** | **+3.3V** |
| 25 | RsvBus7 | **A06** | IRQ6* | **A13** | RsvBus14 |
| 26 | **GND** | **A05** | IRQ5* | **A12** | **+3.3V** |
| 27 | RsvBus8 | **A04** | IRQ4* | **A11** | LI/I* |
| 28 | **GND** | **A03** | IRQ3* | **A10** | **+3.3V** |
| 29 | RsvBus9 | **A02** | IRQ2* | **A09** | LI/O* |
| 30 | **GND** | **A01** | IRQ1* | **A08** | **+3.3V** |
| 31 | RsvBus10 | -12V | +5VSTDBY | +12V | **GND** |
| 32 | **GND** | **+5V** | **+5V** | **+5V** | VPC |

VME FPGA, Front FPGAs, and Main FPGA share the SP internal address space. Each Front FPGA occupies 0x100 address space. Address space for Main FPGA will be detailed as soon as a list of internal LUTs, downloadable over VME becomes available. VME FPGA address space will be assigned shortly after all interested parties approve this proposal.

---

[1] CSC Track-Finder Update, D.Acosta's Presentation at UCLA, Dec. 2001; http://www.phys.ufl.edu/~acosta/cms/srsp_dec01.pdf
[2] CSC Track Finder Crate Specification, created by Mike Matveev and updated by Alex Madorsky, December 17, 2001; http://www.phys.ufl.edu/~madorsky/TrackFinder/TF_backplane_v3.doc
[3] ANSI/VITA 1.1-1997, American National Standard for VME64 Extensions.