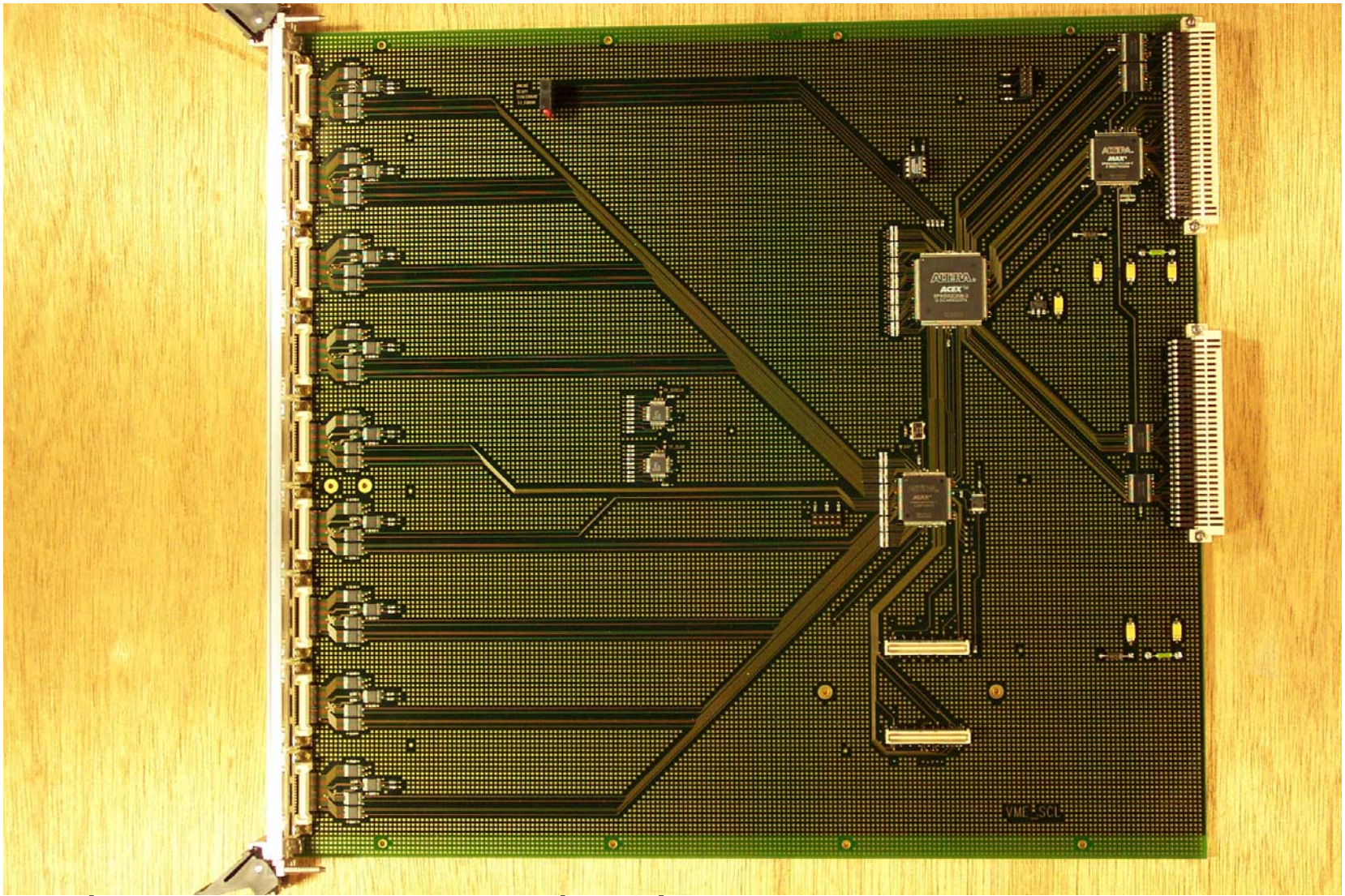
The VME/SCL Board

Jovan Mitrevski
Columbia University
Run 2b L1 Cal Trigger Meeting
December 11, 2003

Purpose

- The VME/SCL board has two main purposes:
  - Provide a compact serial VME interface for the TAB and GAB boards.
  - Fan out the SCL signals to the TAB and GAB boards.
- The VME/SCL board also provides testing support:
  - Generates fake SCL signals for offline testing.
  - Provides control signals to capture transactions to onboard memories.

Photo



Main Components

- VME bus connectors
- Buffers for multi-directional VME bus data lines
- Altera MAX 3000A CPLD for basic VME bus controls, such as AS, DS, and determination if the system is being addressed.
- Altera ACEX 1K FPGA for the main logic of the board: serializing and directing the remote VME transactions and directing the SCL controller
- SCLR daughter card connector
- Altera MAX 3000A CPLD to fan out or fake SCL data and manage the clocks
- 53MHz crystal
- Clock selector (53MHz/7 or SCLR clock)
- 2 Clock fanout chips: serial VME (53MHz/2 currently) and 7MHz SCL clocks
- 9 serial VME/SCL connectors (with corresponding LVDS converters) to connect to the 8 TABs and 1 GAB

LVDS Signals

- serial VME clock
- serial VME frame: framing signal followed by VME A[14:7]; there are five or six unused bits here if needed in the future
- serial VME address: originally VME D[31:16]
- serial VME data: originally VME D[15:0]
- serial VME frame out: frame signal for data from remote
- serial VME data out: 16 bits (serially) of data from remote
- SCL clock
- SCL init
- SCL turn (filtered–only after init is seen)
- SCL accept (formed from l1_accept & 11_period, or faked)
- pulse (used to tell boards to store transaction in memory for testing purposes)
- spare to remote, except on GAB port, where it is l1_error from remote

SCLR Signals Watched

- scl_syncerror :input; -- not locked
- scl_ack :output; -- ack lock
- sync_lost :output; -- sync lost to hub
- scl_ready :input; -- scl ready
- scl_l1_error :output; -- request init to hub
- scl_daterr :input; -- scl data error to vme
- init_section :input; -- init to modules
- init_ack :output; -- init ack to hub
- first_period :input; -- turn to modules
- l1_accept :input; -- accept to modules
- l1_period :input; -- accept
- sync_gap :input; -- gap marker
- l1_qual7 :input; -- monitor to modules
- bx[2..0] :input; -- a check for internal bc[2..0]

Address Map

- AM == 27 and A[23:19] == !(slot) required for board to respond
- A[18:15] is the module number:
  - module 0-7 are the TABs
  - module 8 is the GAB
  - module 15 is the VME/SCL board, i.e. local
- A[14:11] is the chip number, used to select between chips on the remote boards, or functions locally
- A[10:7] is the group number, used to select functions in a chip remotely, or SCL control functions locally
- A[6:4] is the subgroup, currently unused
- other address bits are not available internally

VME/SCL Address Map

- Module = 1: go offline (make SCL signals internally)
- Module = 2: go online (use actual SCL signals)
- Module = 4: read status
- Module = 6: read/write local scratch register
- Module = 3: talk to SCL CPLD
  - chip = 1: set pulse time
  - chip = 2: set pulse width
  - chip = 3: set accept time (when offline)
  - chip = 4: generate SCL init (when offline)
  - chip = 5: generate pulse
  - chip = 6: generate pulse and accept
  - chip = 15: reset local data structures: offline/online, run started, etc.