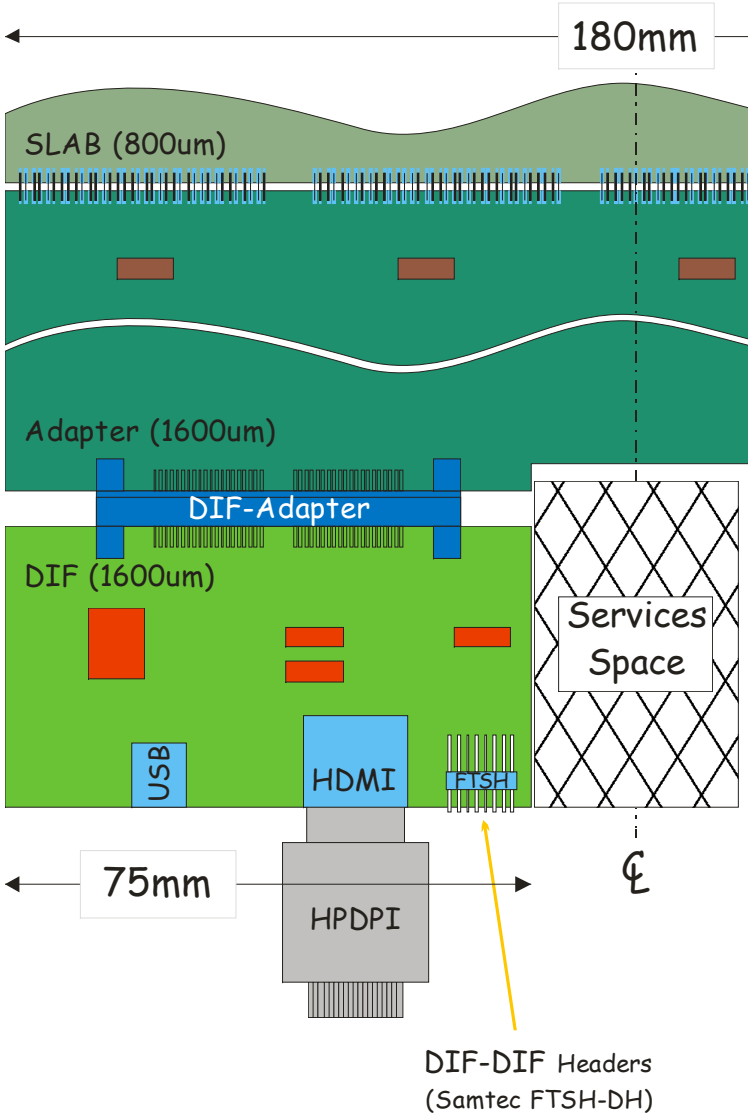
DIF Status and Overview

- Hardware
- LDA interface
- Communications protocol
- Firmware

DIF layout –horizontal plane



Intermediate board:

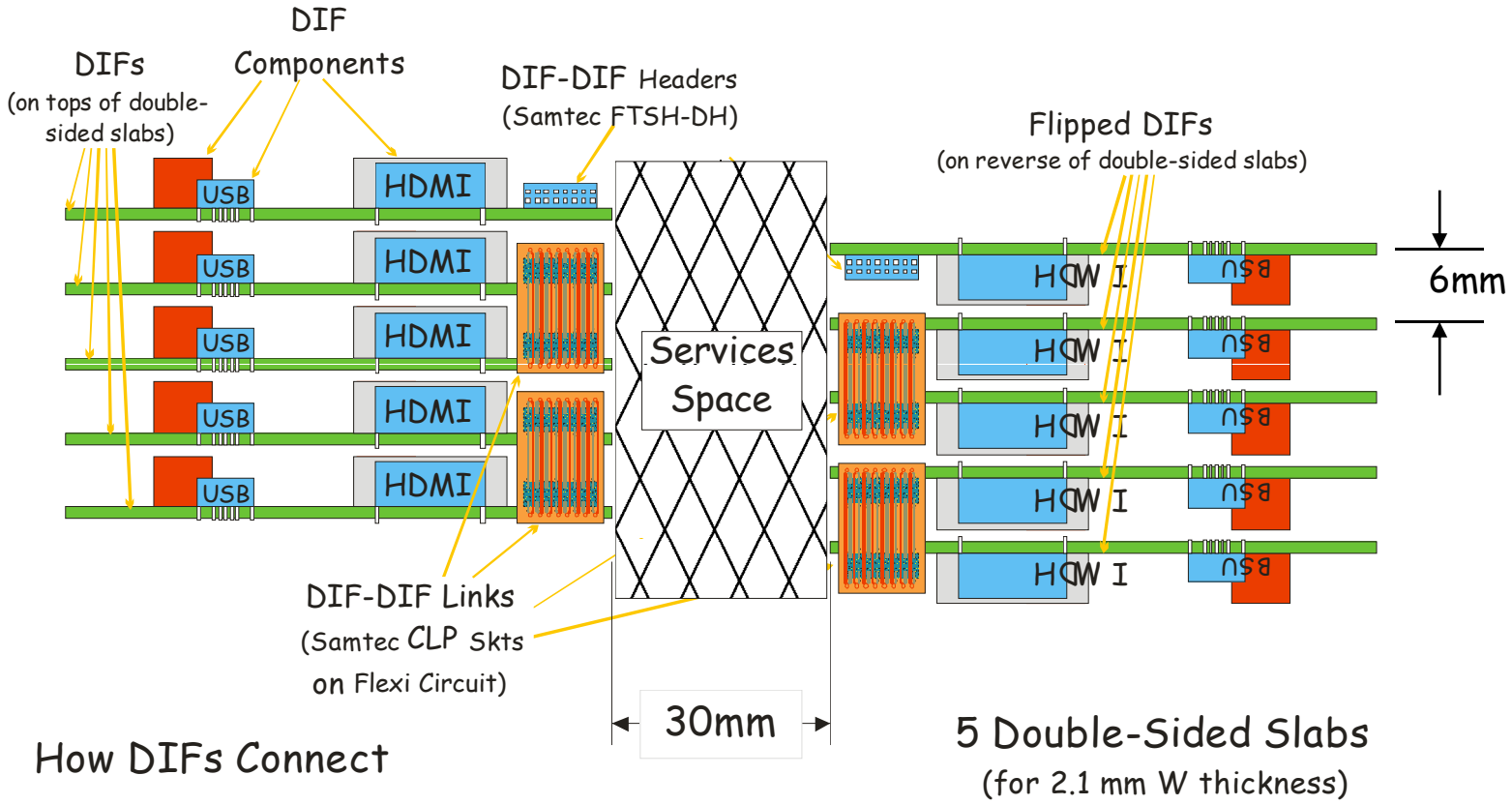
- Power distribution real estate
- Clock fanout
- Level conversion
- Sensors for T, V, I, etc.

DIF:

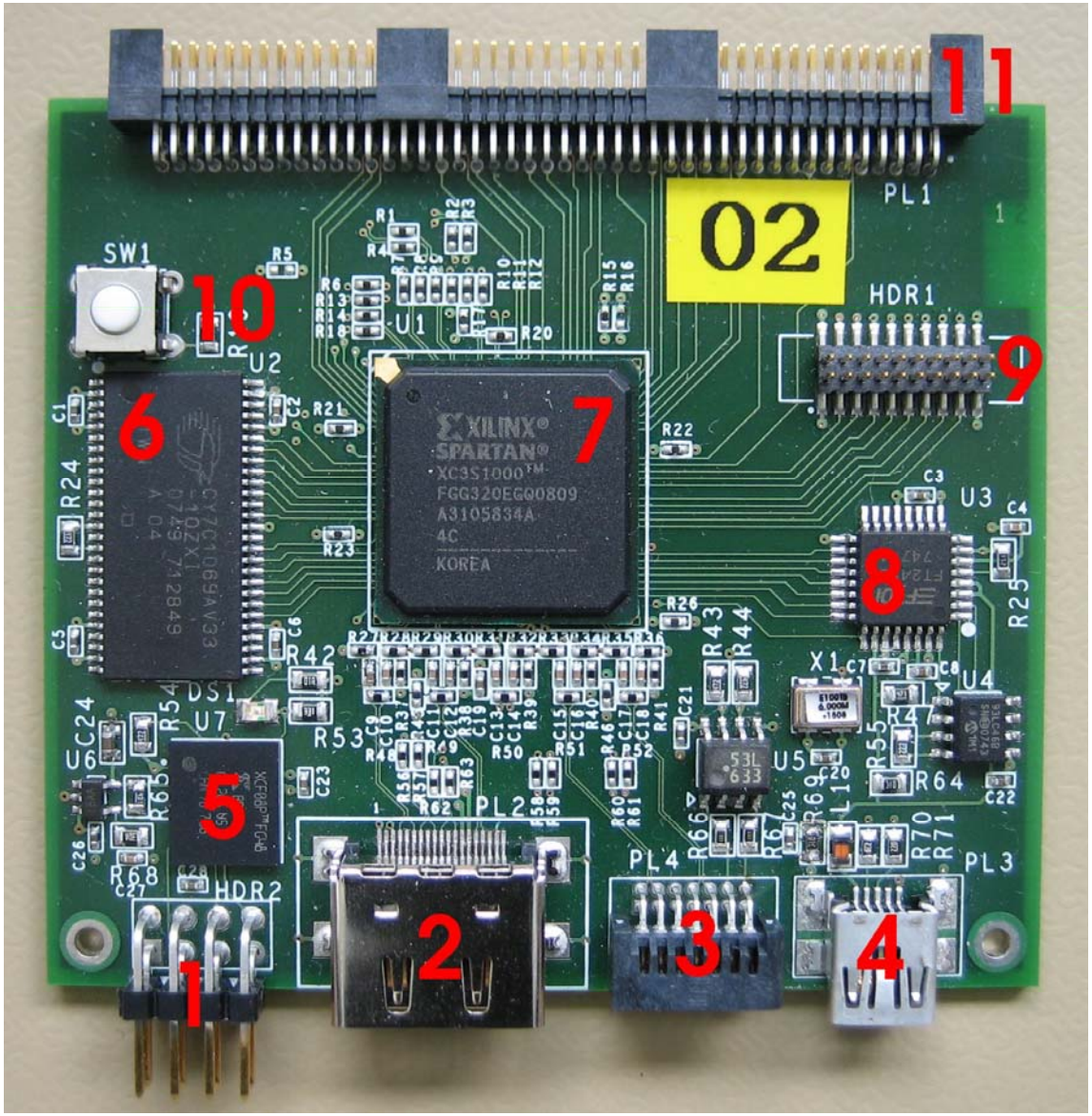
- FPGA, RAM, USB, DIF and LDA links, etc.
- Powered from I.B.
- Stand-alone operation

DIF layout – vertical plane

Tight space constraints and odd requirements: custom DIF board design



Existing DIF hardware

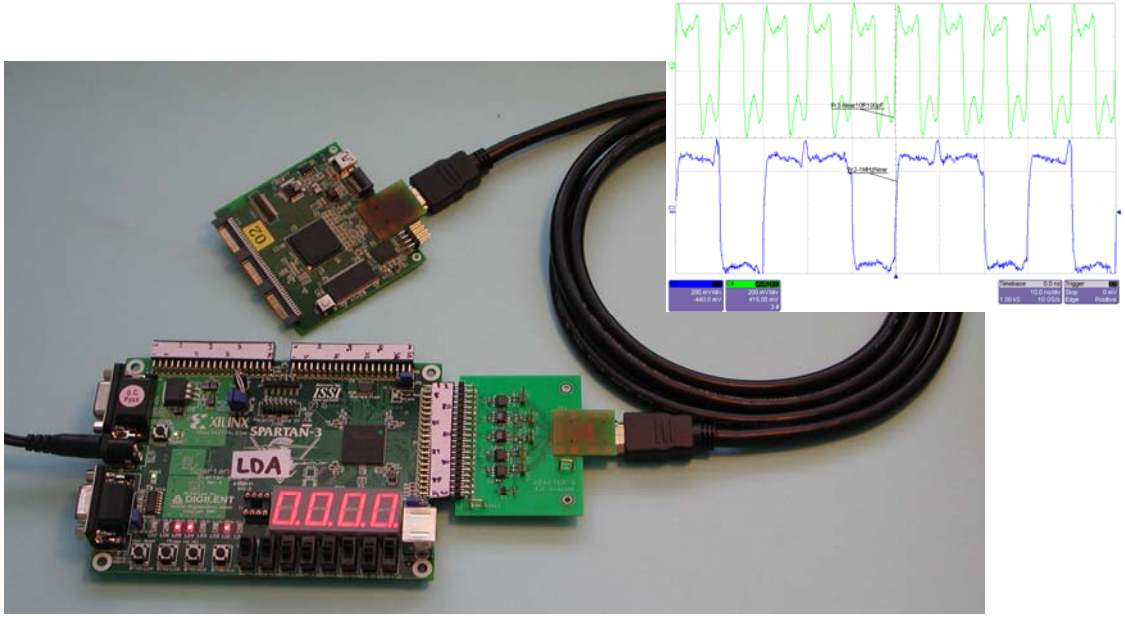


ECAL DIF prototype:
65x72mm, 8 layers

1. JTAG programming header
2. LDA link HMDI connector
3. DIF link connector
4. mini-USB connector
5. Xilinx PROM
6. Cypress 2MB 10ns SRAM
7. Xilinx Spartan3-1000 FPGA
8. FDTI FT245R USB controller
9. 20p user header connector
10. reset pushbutton
11. 90pin SAMTEC IB connector

DIF – LDA interface: hardware

- HDMI connector: 19 pins
- 4 shielded twisted pairs
- 2 additional lines
- cable shielding
- 3V3, 2V5 sourced from DIF side for add-ons
- AC coupled at receiving end



Pin	*Signal Name*	*Description*
1	CLOCK_L2D+	LDA to DIF LVDS Clock +
2	RG1	CLOCK_L2D shielding
3	CLOCK_L2D-	LDA to DIF LVDS Clock +
4	DATA_L2D+	LDA to DIF LVDS Data +
5	RG2	DATA_L2D shielding
6	DATA_L2D-	LDA to DIF LVDS Data -
7	DATA_D2L+	DIF to LDA LVDS Data +
8	RG3	DATA_D2L shielding
9	DATA_D2L-	DIF to LDA LVDS Data +
10	GEN_D2L+	DIF to LDA General Use +
11	RG4	GEN_D2L shielding
12	GEN_D2L-	DIF to LDA General Use -
13	2V5	Unused (2V5 power)
14	3V3	Unused (3V3 power)
15	ASYNC_L2D+	LDA to DIF Asynchronous + *(not shielded)*
16	ASYNC_L2D-	LDA to DIF Asynchronous – *(not shielded)*
17	RG5	Cable Shield
18	GND	Ground (for power)
19	GND	Ground (for power)

DIF – LDA interface: physical layer protocol

- 8B/10B encoded 20bit transfers deliver 8bit comma char. + 8 user bits for CMD encoding
- bit clock between 50MHz-150MHz
- Comma chars: 15 available, but not freely usable
- 8B/10B link firmware performs parity check
- 8B/10B firmware very user-friendly (thanks Marc!)

DIF – LDA interface data layer protocol

- Transfer is either (fast) command or block transfer
- CMD is 16(20) bit long, of which 8 user defined:
  - comma character
  - 8 bit command
- Block transfer:
  - block transfer header: comma char + data block CMD
  - subsequently 16 bit data transfers
  - concluding with CRC_16 checksum
  - proposed max. length: 1kB

DIF – LDA commands

- CMD bit [0] is address indicator:

  DIF on LDA link, or DIF on DIF-DIF link
- CMD bits [1:7] freely usable
- Special commands with reserved comma chars:
- SYNC_CMD: [1:3 CMD_ID][4:7 DELAY]

  CMD gets executed after DELAY slow clock edges
- CLK_RESYNC

  insert a phase shift of DELAY clock cycles at the ‘0’ level of the slow clock

DIF – LDA block transfers

- First transfer:

  Dedicated comma character + block transfer command
- 16b length field
- 16b start address
- N*16b data words
- 16b CRC_16 checksum

- Block transfer is not very efficient for short blocks of data (2B payload => 10B transfer)
- Max. length block transfer takes 103.2 mus at 100MHz
- Are we in a hurry when transferring data?

DIF – LDA block transfers: configuration

- VFE needs 900 bits (estimated)
- 128 chips on a slab -could be more
- single configuration daisy chain
- 1MHz JTAG clock speed -could well be less

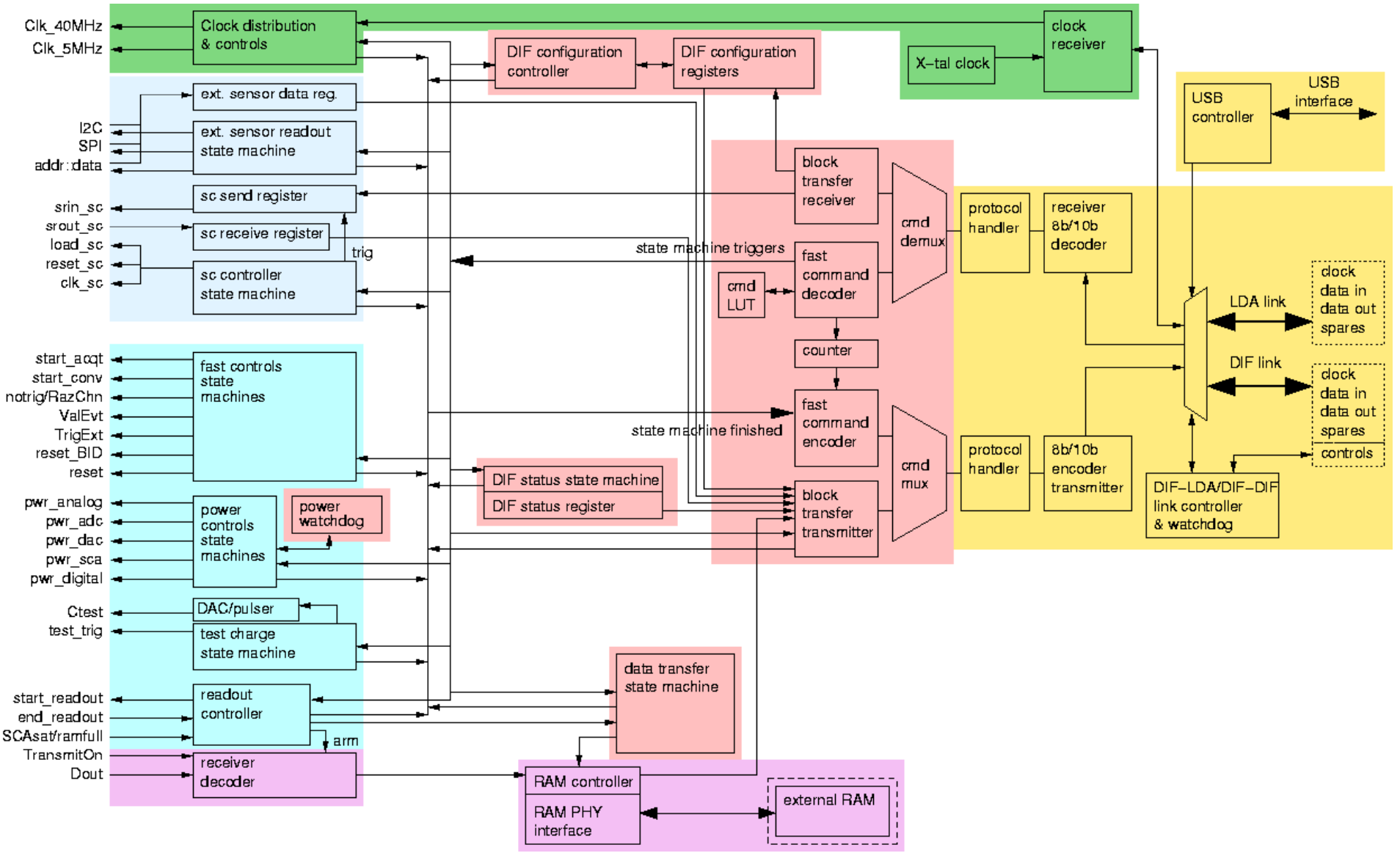
⇒0.12s needed for setting config. registers.

⇒Block data transfer time is negligible

DIF – LDA asynchronous link

- Pre-defined functionality
- Clock-encoded pulse (AC coupling!)
- …?

DIF block diagram – 1 level down



project planning & needs:

Revised project planning for 2008 & 2009:

- DAQtest ‘08:
  - ‘minimal DIF’ hardware & firmware to send data down the link
- EUDET beam test ‘09:
  - full-fledged DIF hardware&firmware
  - Intermediate boards for EUDET slabs