Tileboard update

S. Kim[1]

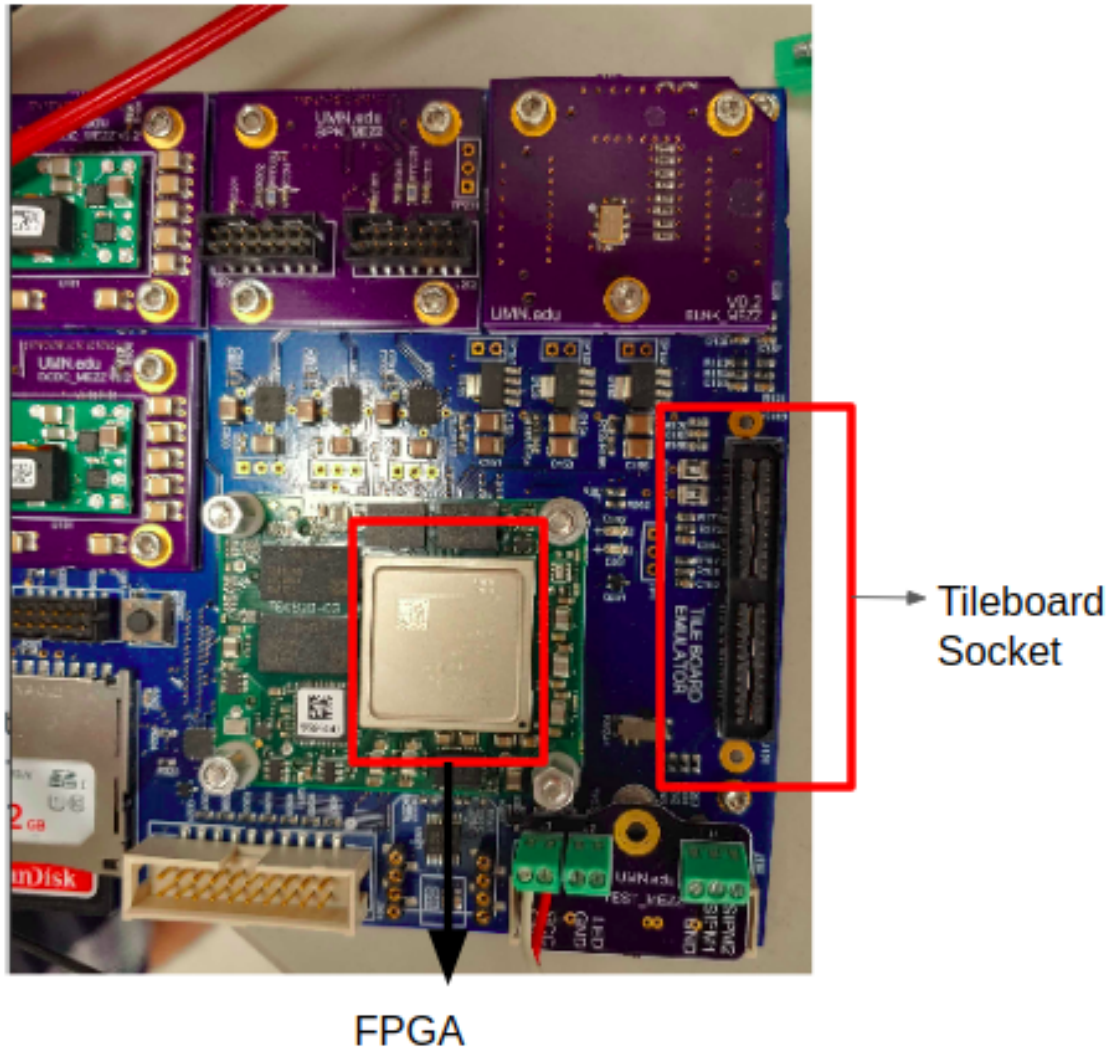
[1]Department of Physics
Florida State University

Tileboard update, 120420

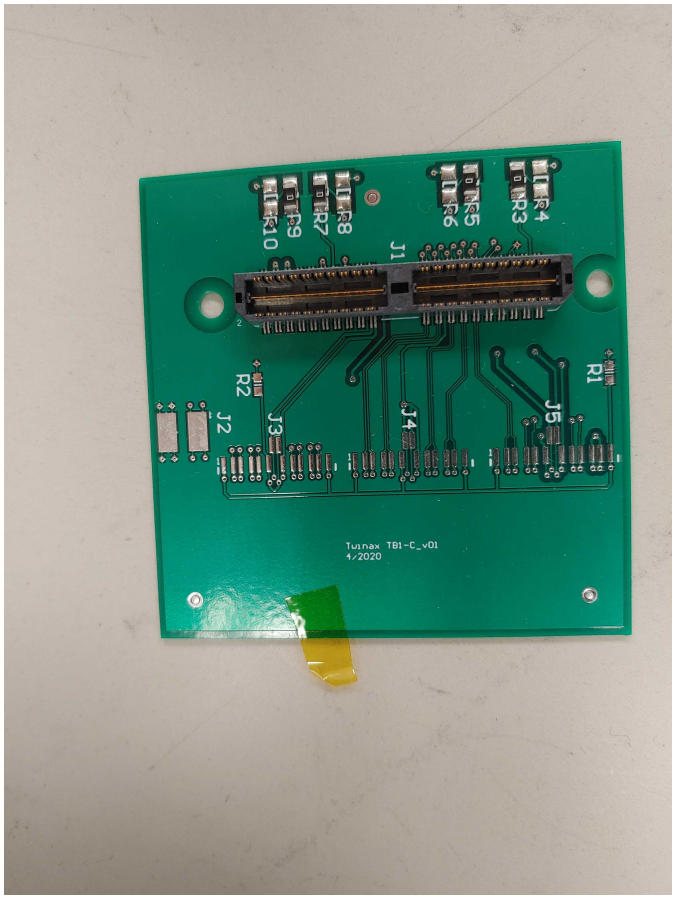
Introduction and Summary

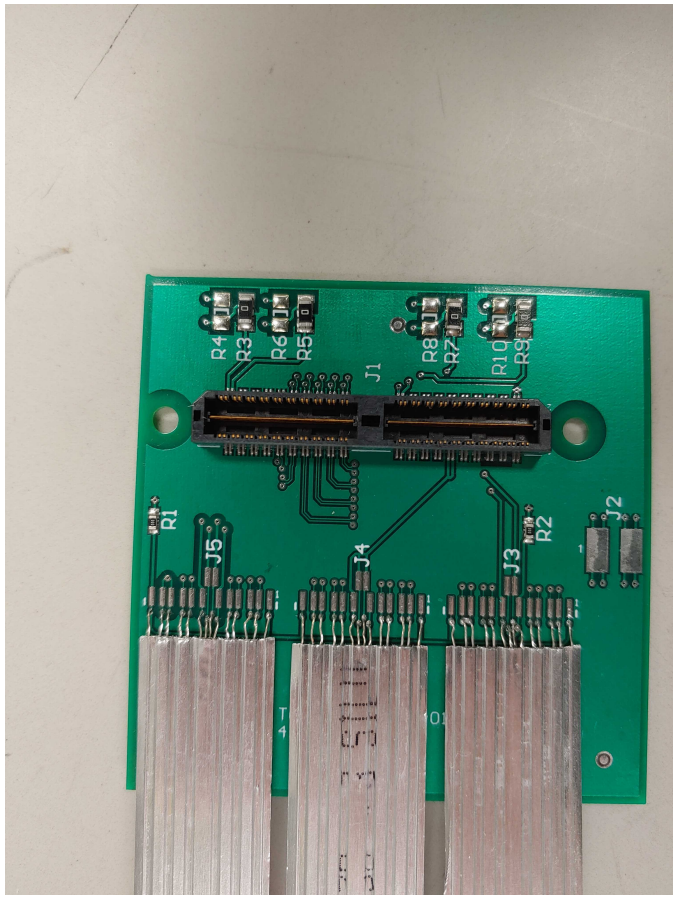
- FSU HGCAL team testing candidates of different brands of cables for Tileboard (TB) and Wingboard (WB) connection
- Experimental Setup
  - UMinn TB emulator
  - FSU self-designed adaptor boards
  - Cable brand: 3M SL8801/12-10DA5-00
  - WBs substituted with SAMTEC Twinax QSH Loopback RevB v1 for simple data transmission testing

Pictures of the MB

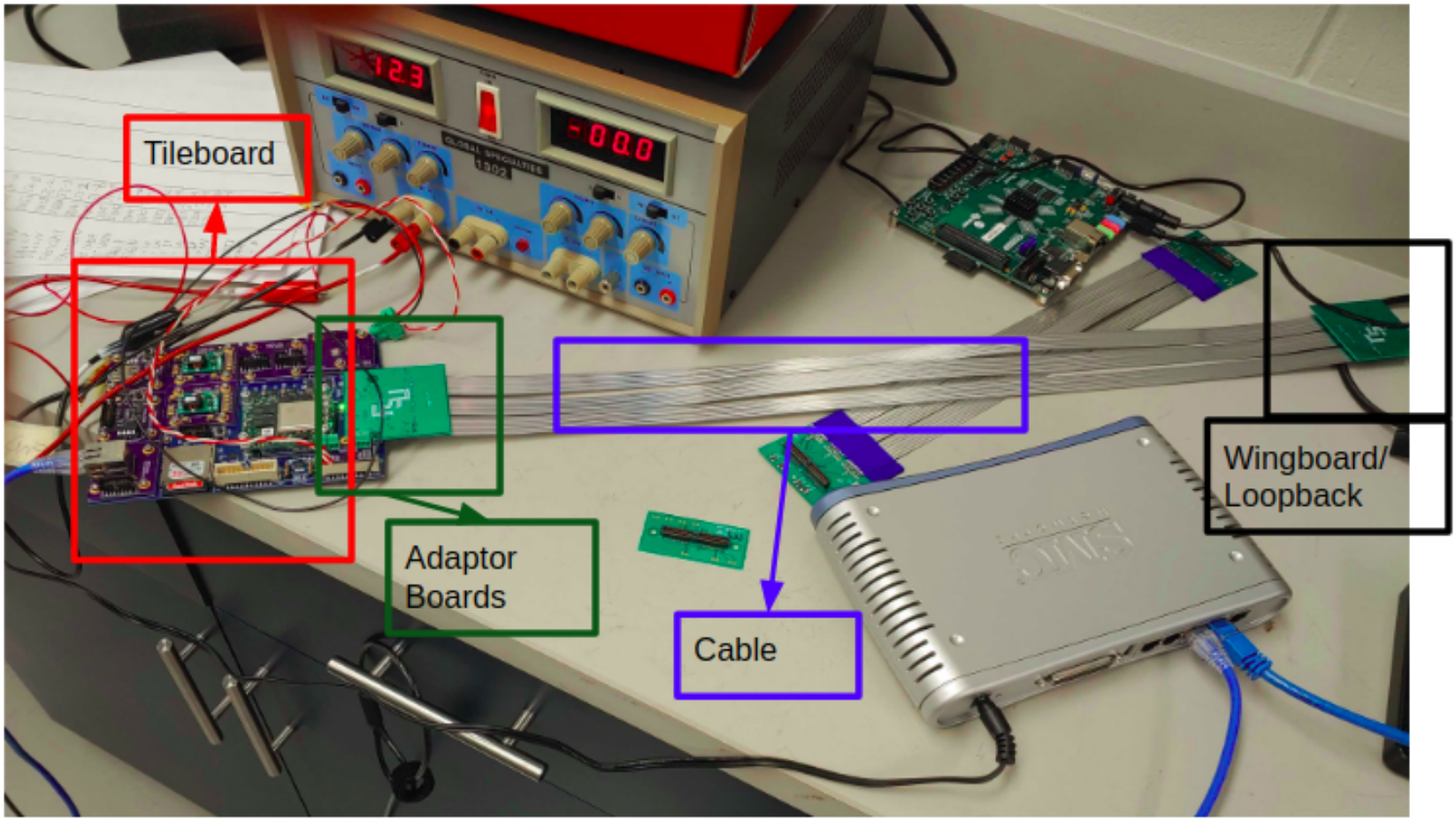


Pictures of the adaptor boards





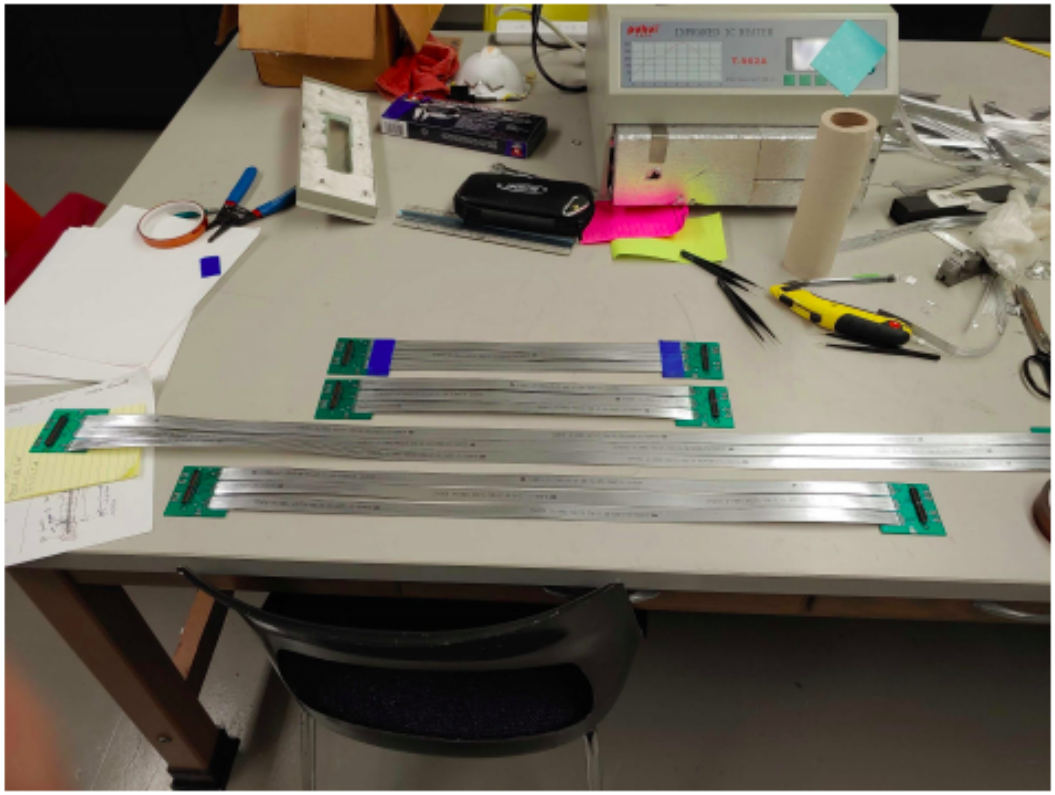
Pictures of the complete setup



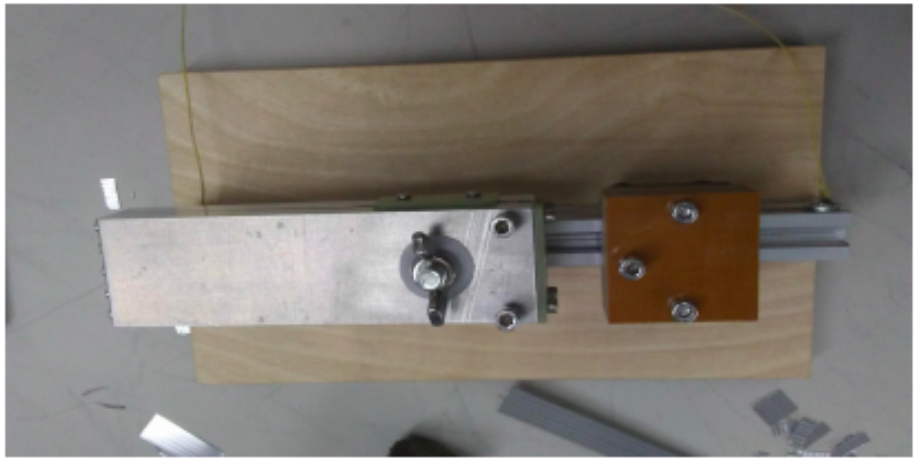
Adaptor-Cable-Adaptor Connection

- Adaptor-Cable-Adaptor-Loopback connection tested with FPGA bitstream to check its data transmission performance
- Adaptor-Cable-Adaptor soldered and created by the FSU HGCAL team
- Adaptor (AB) -Cable-Adaptor connection made for 3 different sets of length (369,655,1013 mm)
- Machine from FSU machine shop for precise strip off of the cable

AB-Cable-AB connections



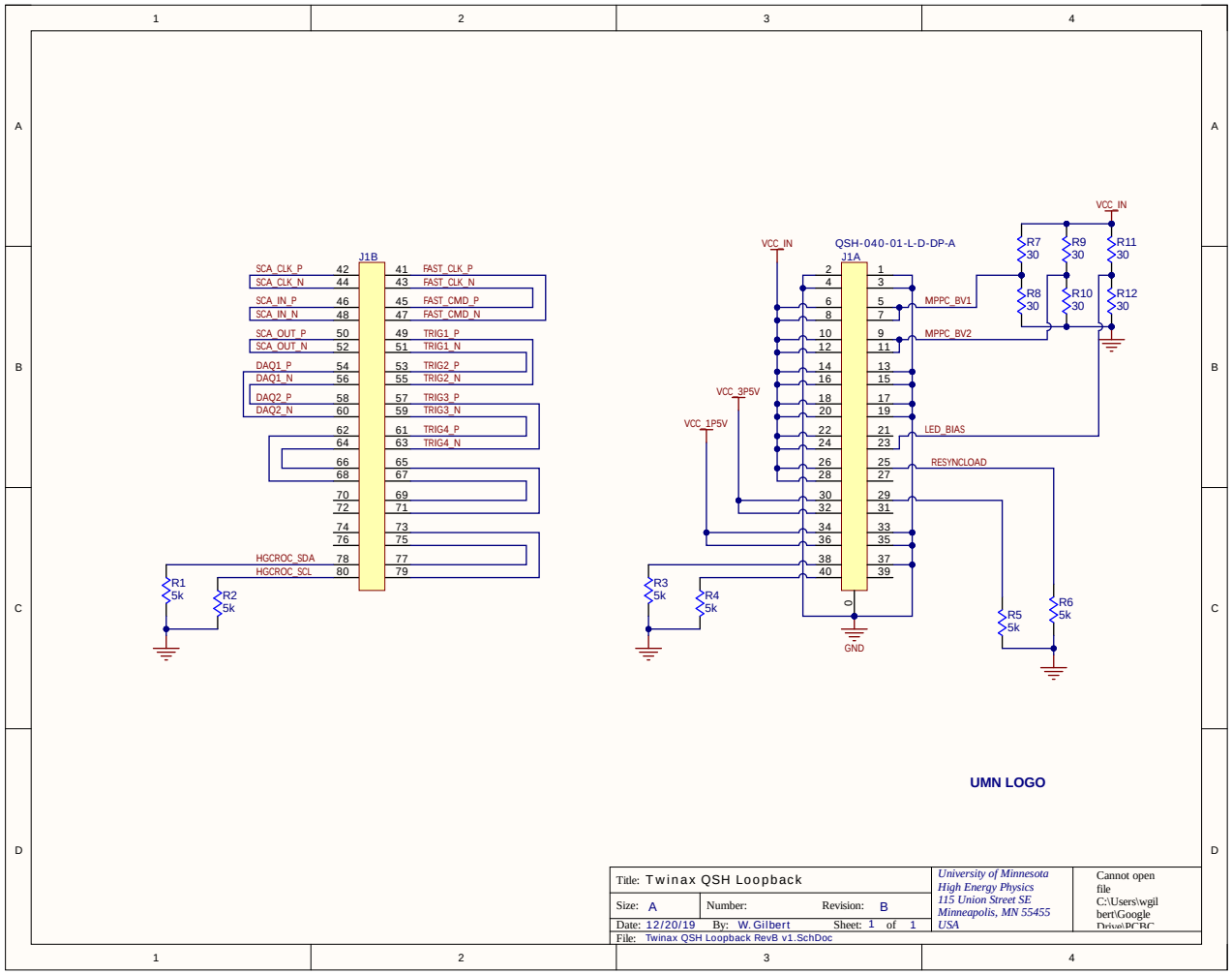
FSU Machine for Cable stripping



Bitstream of cabletester

- Xilinx UltraScale+ FPGA (xzcu2cg-sfvc784) on TB
- LoopbackFirmware bitstream
  - Test error rate by sending data on OBUFDS and measuring received data IBUFDS
  - Our loopback is **twinax QSH_v1 Rev.B**
  - Its I/O ports connection in next slide
  - Error rate printout on the PC terminal

QSH loopback



QSH loopback result

- 1 4 out of 9 I/O loopback connections function properly
- 2 3 out of 9 I/O loopback connections fail due to our adaptor board design
- 3 2 out of 9 I/O loopback connections fail due to absence of signal from the GBTSCA
- 4 In summary, all I/O loopback connections function as expected

QSH loopback table result

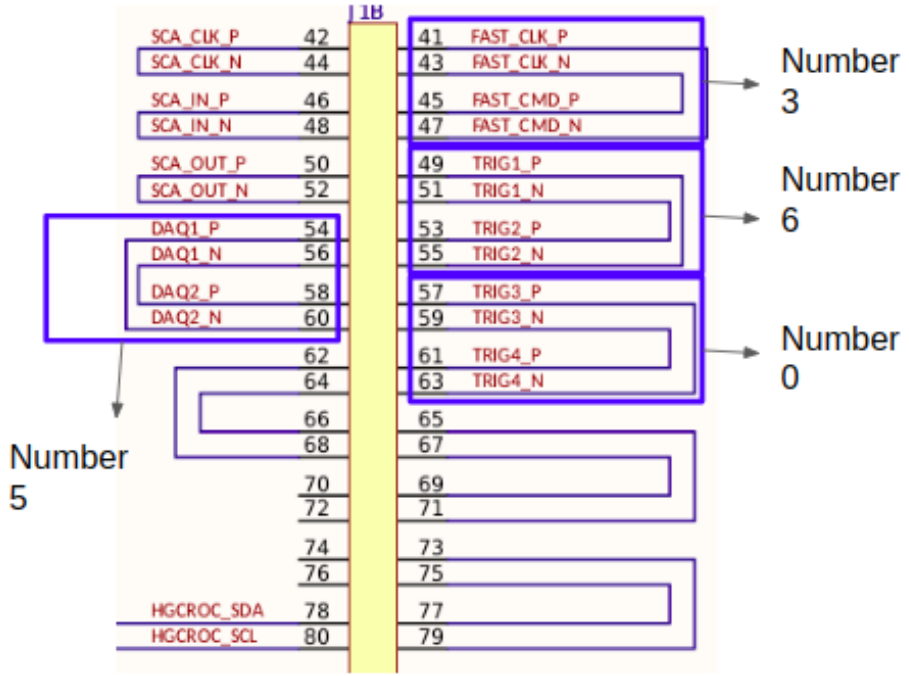
- 1 First Column means one of 9 I/O connections within the QSH loopback connection.
- 2 Second Column means percentage of data (POS DIFF) sent out on OBUFDS
- 3 Third Column means (POS DIFF) error rate received on IBUFDS

QSH loopback table result

- All HGCROC1 and ELINK-FAST connections are tested to have 0 error rate.

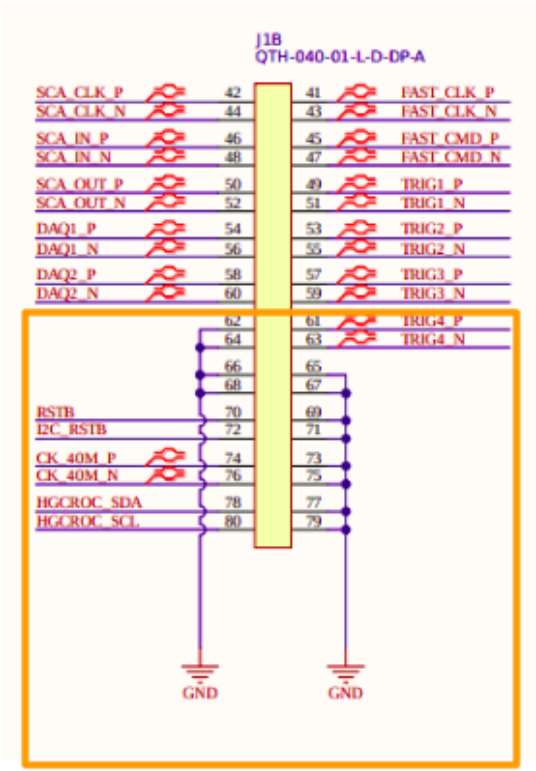
Loopback number	Data Out	Error rate
0	1	0.00000
1	1	0.09660
2	1	1.52438
3	1	0.00000
4	1	0.76844
5	1	0.00000
6	1	-0.00000
7	1	0.38116
8	1	0.77371

Table: Loopback Summary



Not working connections - Socket grounding

- Following loopback connections are NOT getting 0 error rate
- It's turns out to be a problem of the circuit design, not a problem with bitstream or cables.
- Currently designed FSU sockets grounds all HGCROC2 DAQ,TRIG signals

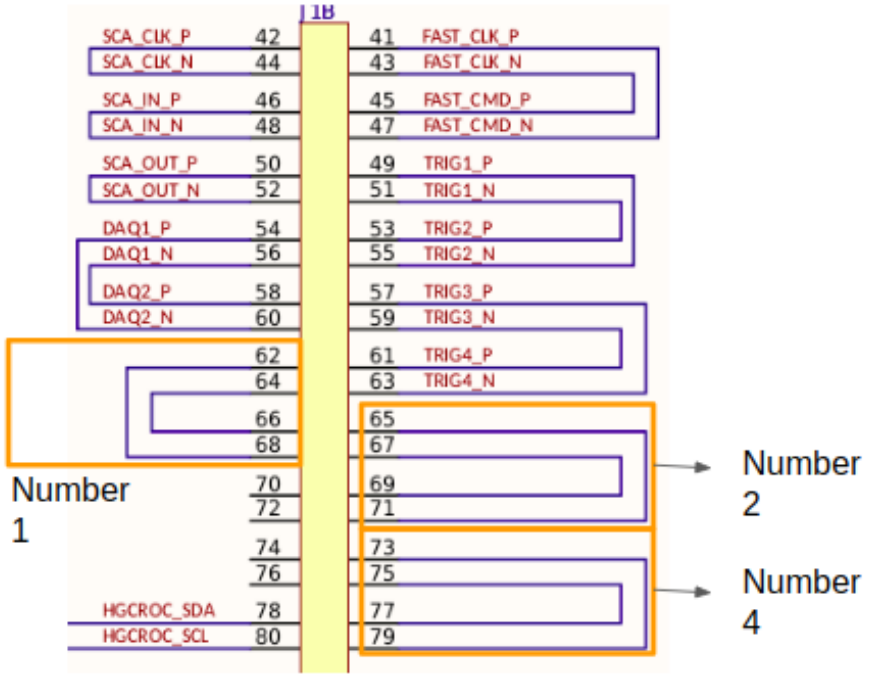


QSH loopback table result

- All HGCROC2 signals from OBUFDS are grounded in adaptor boards. Thus, signals don't loop-back and have non-zero error rate.

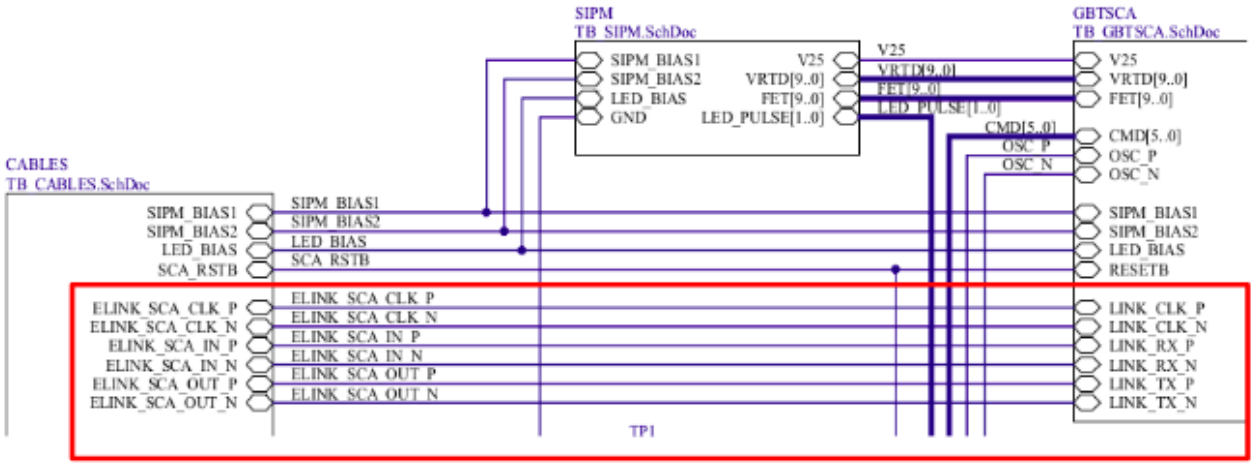
Loopback number	Data Out	Error rate
0	1	0.00000
1	1	0.09660
2	1	1.52438
3	1	0.00000
4	1	0.76844
5	1	0.00000
6	1	-0.00000
7	1	0.38116
8	1	0.77371

Table: Loopback Summary



Not working connections - GBTSCA

- Second, Our TB emulator has no GBTSCA as its component. Any I/O, which receives signal from GBTSCA, will show error rate

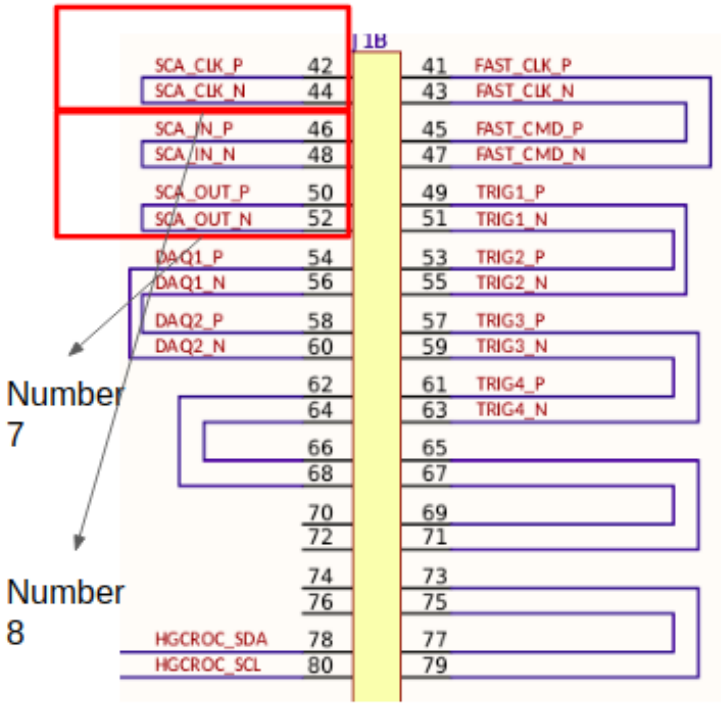


QSH loopback table result

- No signal is detected in IBUFDS due to absence of GBTSCA. Thus, we get non-zero error rate.

Loopback number	Data Out	Error rate
0	1	0.00000
1	1	0.09660
2	1	1.52438
3	1	0.00000
4	1	0.76844
5	1	0.00000
6	1	-0.00000
7	1	0.38116
8	1	0.77371

Table: Loopback Summary



Result and interpretation

- FSU's Adaptor-Cable-Adaptor-loopback connection
  - FPGA bitstream editable to adapt to loopback other than twinax_QSH_RevB_v1
  - Adaptor-cable-Adaptor connection was all properly soldered with no short-circuit
  - FSU machine shop's cable stripper also has decent performance.
  - Based on absence of GBTSCA and design of the socket board, All I/O loopback connections function as expected
- Cable 3M SL8801/12-10DA5-00's performance tested to be compatible for TB-WB connection

Future

- Presence of GBTSCA and change in the adaptor board design can lead to test of all I/O ports connections.
- Cable brands other than 3M SL8801/12-10DA5-00 can be tested as well.

Back-Up

QTH variation and understanding - 1

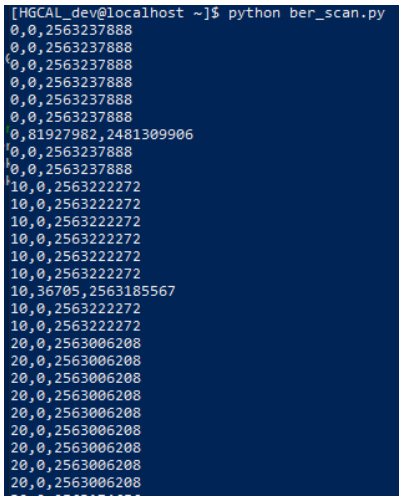
- Now, we can start creating QSH's XDC file.
- Before that, I tried various experiment in QTH, to better understand the terminal output info
  - 1 **QTHInverted**
  - 2 **QTHhalfInverted**
  - 3 **QTHwrongAssign**
- Each of items above taught me some useful information.
- Place of inversion discovered in Link : loopback-top.v at L76

QTH variation and understanding - 2

- After discovery of place of inversion, inverting differential signals in All bytegenerators tried

	0	1	2	3	4	5	6	7	8
Inv	Y	Y	Y	Y	Y	Y	Y	Y	Y

Table: Inversion table



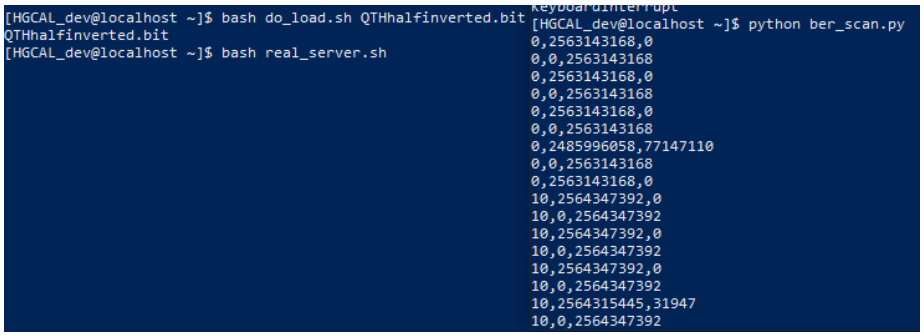
- When signals are inverted, the terminal reads the error rate received (0) in the second column and the data went out (2E10) in the third column

QTH variation and understanding - 3

- Inverting only odd number bytegenerators tried.

Inv	0 N	1 Y	2 N	3 Y	4 N	5 Y	6 N	7 Y	8 N

Table: Inversion table



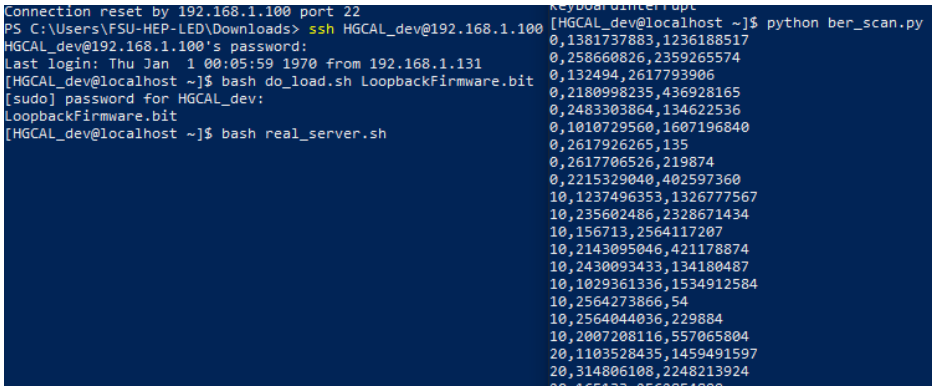
- Each row in time slot means each bytegenerator
- Used to think each row in time slot meant error rate for sub-time interval.

QTH variation and understanding - 4

- Assigning pins to pins that are not connected via loopback

	0	1	2	3	4	5	6	7	8
Loop	N	N	N	N	N	N	N	N	N

Table: Inversion table



- Not getting "0" value for either column

QTH variation and understanding - 5

- Although looking trivial, it gave principles that
  - 1 When signal is inverted, second and third columns are inverted
  - 2 Each row means bytegenerator number
  - 3 When no loop formed, both columns read non-zero value

QSH XDC assignment - 1

- To write down QSH XDC file, start with the easiest bytegenerator

QTH/QSH Output	QTH Input	QSH Input
HGCROC1-TRIG-IN-P(N)2 [F2,E2]	HGCROC1-TRIG-OUT-P(N)3 [A9,A8]	HGCROC1-TRIG-OUT-N(P)3 [A8,A9]

Table: bytegenerator0

- PACKAGE_PIN [get_ports A9 HGCROC1_TRIG_OUT_P3]
- PACKAGE_PIN [get_ports A8 HGCROC1_TRIG_OUT_N3]
- Inversion from A8,A9 design, Inverting those legs - No Inversion

- bytegenerator1

QTH/QSH Output	QTH Input	QSH Input
HGCROC1-TRIG-IN-P(N)1 [G6,F6]	HGCROC2-TRIG-OUT-P(N)0 [A7,A6]	HGCROC1-TRIG-in-N(P)0 [E8,F8]

Table: bytegenerator1

- PACKAGE_PIN [get_ports F8 HGCROC2_TRIG_OUT_P0]
- PACKAGE_PIN [get_ports E8 HGCROC2_TRIG_OUT_N0]
- Inversion from E8,F8 design, Inverting those legs - No Inversion

- bytegenerator2

QSH Output HGCROC2-TRIG-OUT-N(P)0 [A6,A7]	QTH Output HGCROC1-TRIG-OUT-P(N)0 [F8,E8]	QTH/QSH Input HGCROC2-TRIG-out-P(N)1 [B4,A4]

Table: bytegenerator2

- PACKAGE_PIN [get_ports A7 HGCROC1_TRIG_OUT_P0]
- PACKAGE_PIN [get_ports A6 HGCROC1_TRIG_OUT_N0]
- Inverting Output legs - Inversion

- bytegenerator3

QTH/QSH Output	QTH Input	QSH Input
fc-sig-in-p(n) [C9,B9]	HGCROC2-TRIG-OUT-P(N)2 [E1,D1]	fc-clk-in-n(p) [D6,D7]

Table: bytegenerator3

- PACKAGE_PIN [get_ports D7 HGCROC2_TRIG_OUT_P2]
- PACKAGE_PIN [get_ports D6 HGCROC2_TRIG_OUT_N2]
- Inversion from D7,D6 design, Inverting those legs - No Inversion

- bytegenerator4

QSH Output HGCROC2-TRIG-OUT-N(P)2 [D1,E1]	QTH Output fc-clk-in-p(n) [D7,D6]	QTH/QSH Input HGCROC2-TRIG-out-P(N)3 [E5,D5]

Table: bytegenerator4

- PACKAGE_PIN [get_ports E1 fc-clk-in-p]
- PACKAGE_PIN [get_ports D1 fc-clk-in-n]
- Inverting Output legs, Inversion from D1,E1 design, Inversion from E5,D5 design - Inversion

- bytegenerator5

QTH/QSH Output HGCROC1-DAQ-IN-P(N)0 [E9,D9]	QTH Input HGCROC1-DAQ-OUT-P(N)1 [G8,F7]	QSH Input HGCROC1-DAQ-OUT-N(P)1 [F7,G8]

Table: bytegenerator5

- PACKAGE_PIN [get_ports G8 HGCROC1_DAQ_OUT_P1]
- PACKAGE_PIN [get_ports F7 HGCROC1_DA1_OUT_N1]
- Inverting G8,F7 legs - Inversion

- bytegenerator6

QSH Output HGCROC2-DAQ-OUT-N(P)1 [B1,C1]	QTH Output SCA-MISO-p(n) [AE9,AE8]	QTH/QSH Input HGCROC2-DAQ-OUT-P(N)0 [E4,E3]

Table: bytegenerator6

- PACKAGE_PIN [get_ports C1 SCA-MISO-p]
- PACKAGE_PIN [get_ports B1 SCA-MISO-n]
- Inversion from B1,C1 design, Inverting those legs - No Inversion

- bytegenerator7

QSH Output	QTH Output	QTH Input	QSH Input
SCA-MOSI-p,SCA-MISO-n [AC9,AE8]	SCA-MOSI-p(n) [AC9,AD9]	HGCROC2-DAQ-OUT-P(N)1 [C1,B1]	SCA-MOSI-n,SCA-MISO-p [AD9,AE9]

Table: bytegenerator7

- PACKAGE_PIN [get_ports AE8 SCA-MOSI-n]
- PACKAGE_PIN [get_ports AE9 HGCROC2_DAQ_OUT_P1]
- PACKAGE_PIN [get_ports AD9 HGCROC2_DAQ_OUT_N1]

- bytegenerator8

QSH Output	QTH Output	QTH Input	QSH Input
SCA-CLK-p,CK-40M-n [AF7,C4]	SCA-CLK-p(n) [AF7,AF6]	CK-40M-p(n) [D4,C4]	SCA-CLK-n,CK-40M-p [AF6,D4]

Table: bytegenerator8

- PACKAGE_PIN [get_ports C4 SCA-CLK-n]
- PACKAGE_PIN [get_ports AF6 CK-40M-p]
- PACKAGE_PIN [get_ports D4 SCA-CLK-n]

QSH loopback printout

```
[HGCAL_dev@localhost ~]$ python ber_scan.py
0,2618980864,0
0,2388256709,230724155
0,1024982908,1593997956
0,2618980864,0
0,1481080272,1137900592
0,2618980864,0
0,3,2618980861
0,2076700364,542280500
0,1476282404,1142698460
10,2564436736,0
10,1856745667,707691069
10,950492067,1613944669
10,2564436736,0
10,1318877668,1245559068
10,2564436736,0
10,0,2564436736
10,2041804883,522631853
10,1502149069,1062287667
20,2563008512,0
20,1417696411,1145312101
20,919710335,1643298177
20,2563008512,0
20,1186319309,1376689203
20,2563008512,0
20,0,2563008512
```