Slice Testboard Update

Julia Gonski

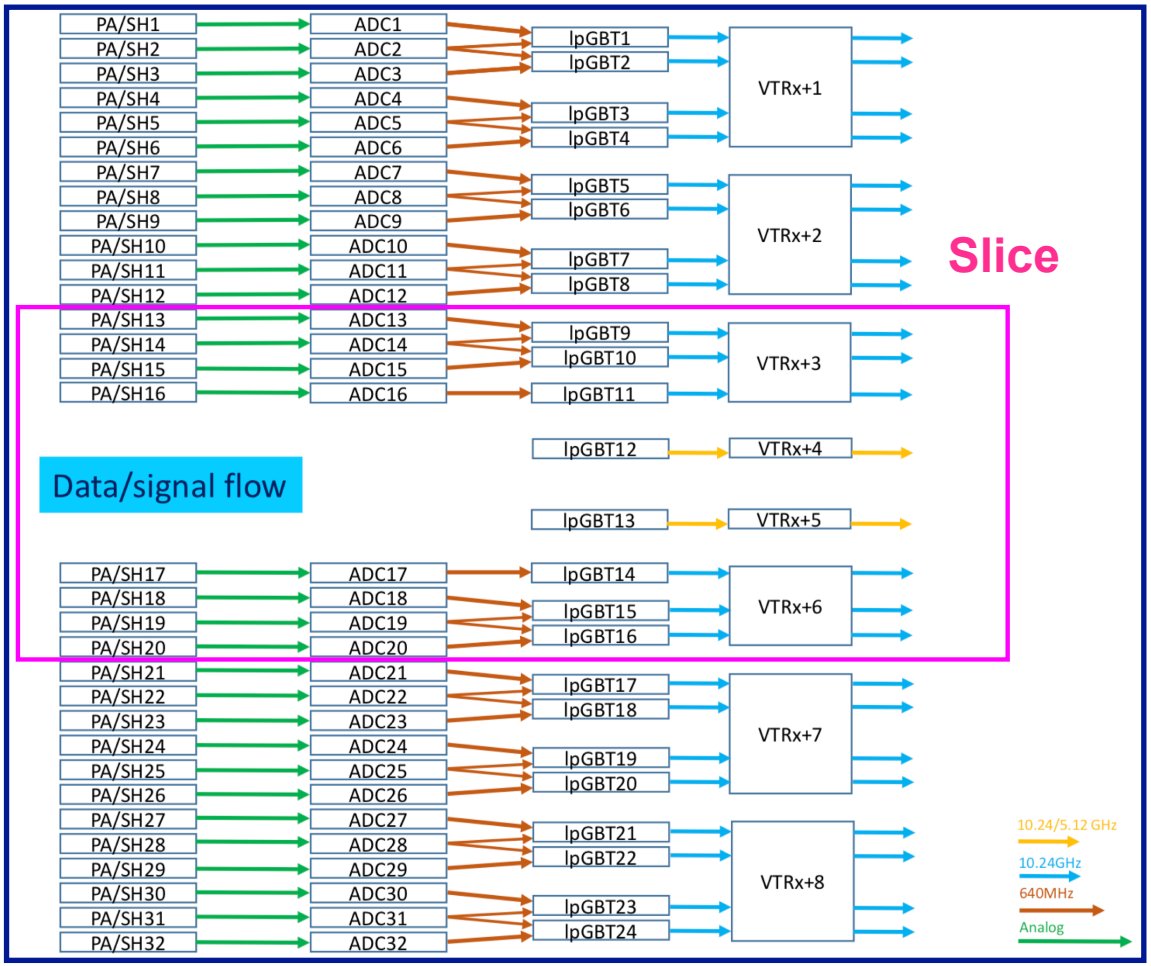
1 December 2021

LAr Week

NEVIS LABORATORIES
COLUMBIA UNIVERSITY

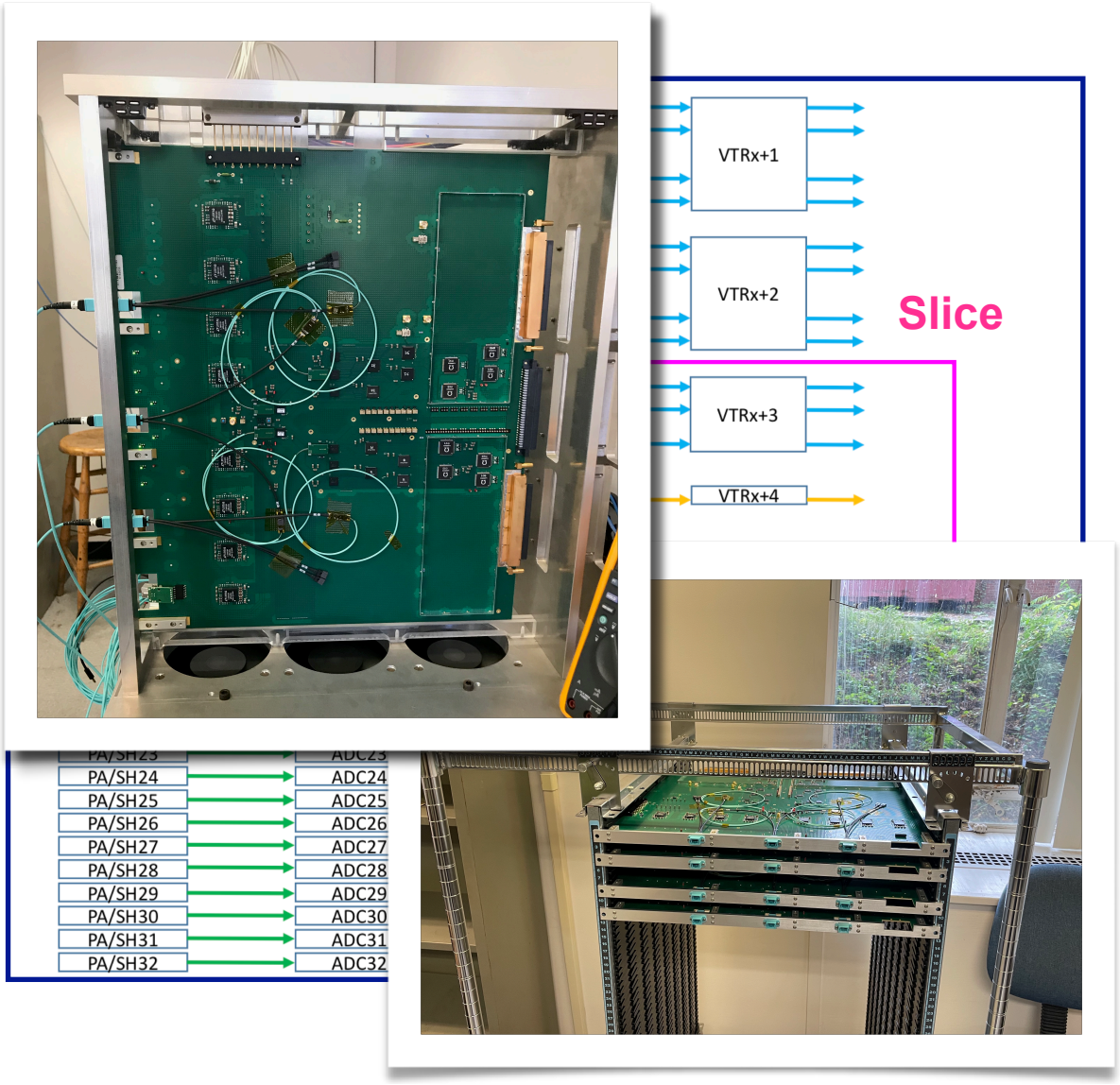
ATLAS EXPERIMENT

Slice Testboard Intro



- **Feb 15: 2 v1.1 boards**
  - 32 channels with LAUROCv2 PA/S + COLUTAv3 ADC + lpGBTv0
- **July 8**: received LAUROCs to assemble **3 additional v1.1** boards
- **Oct 5**: received assembled 3 additional v1.1 boards
  - Delay due to assembler mix up of “all good” lpGBTs
- **Next**: FEB2 prototype (full 128 channels)

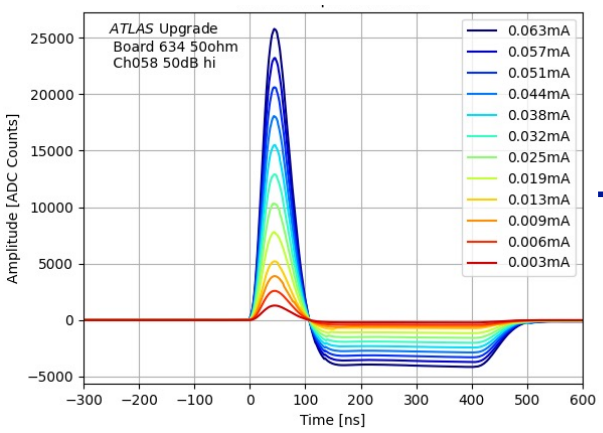
Slice Testboard Intro

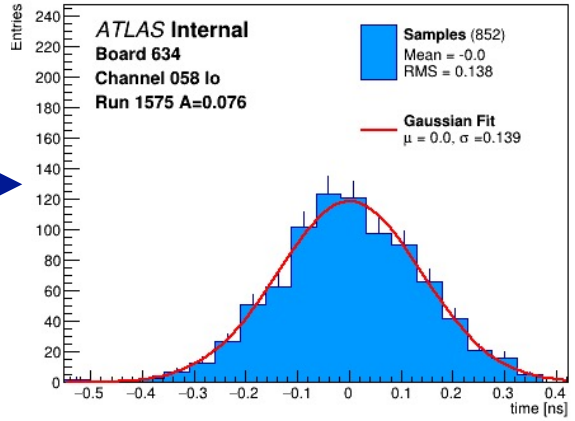


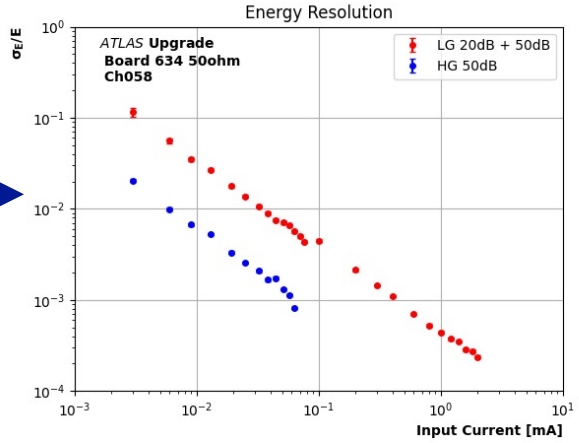
- **Feb 15: 2 v1.1 boards**
  - 32 channels with LAUROCv2 PA/S + COLUTAv3 ADC + lpGBTv0
- **July 8**: received LAUROCs to assemble **3 additional v1.1** boards
- **Oct 5**: received assembled 3 additional v1.1 boards
  - Delay due to assembler mix up of “all good” lpGBTs
- **Next**: FEB2 prototype (full 128 channels)

Overview

- **Last update:** LAr Week, October 2021; results with 2 old v1.1 boards
  - Fully functional data readout on all 32 channels
  - Demonstrated lpGBT 12/13 redundant control functionality
  - Can configure & read back from all LAUROC, COLUTA, and lpGBT chips
  - Full board ADC calibration in 25 minutes (parallelized)







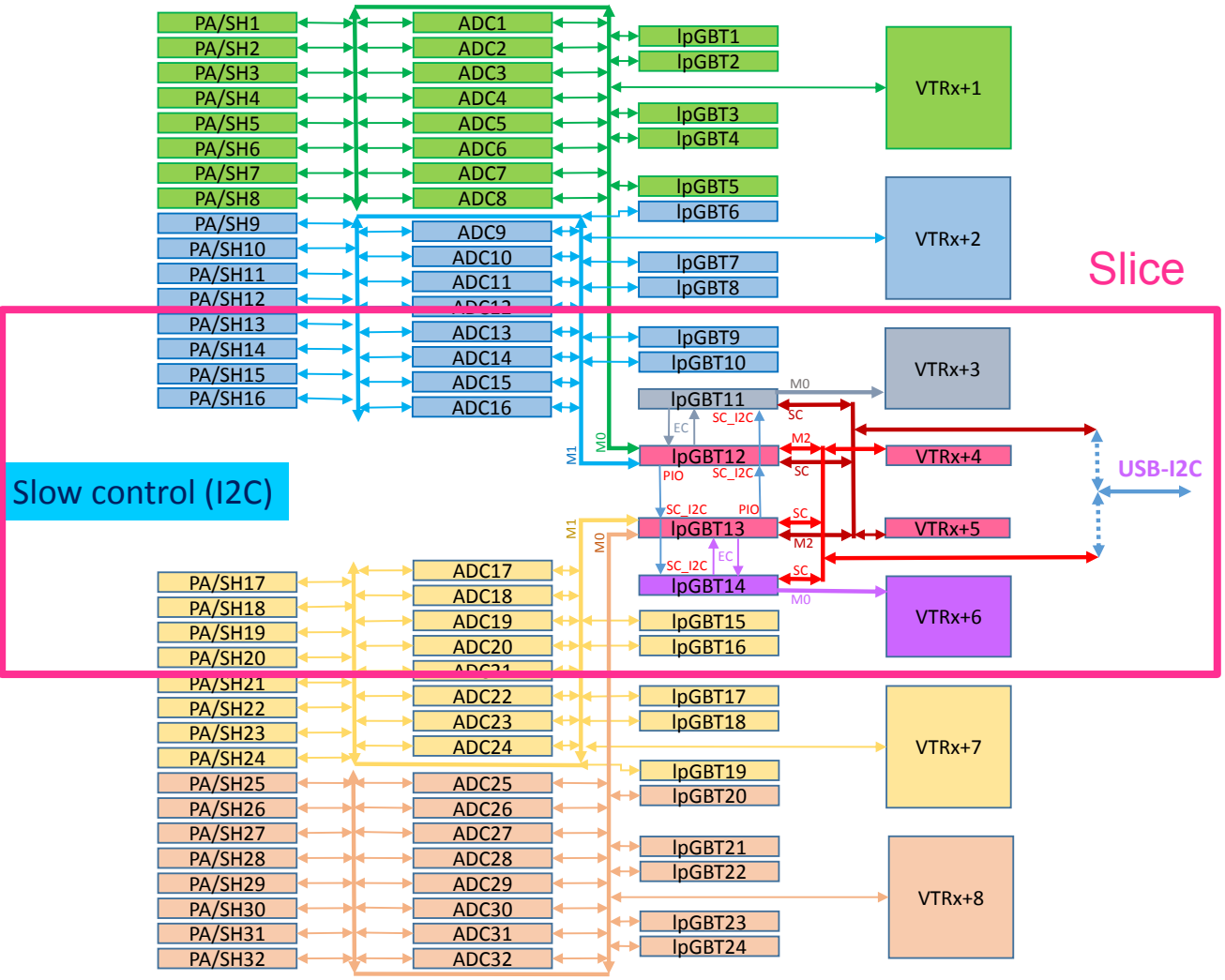
- **Today:** testing & validation studies with new 3 v1.1 boards

v1.1 Board Summary

- v1.1 original production (2 boards): fully functional I2C configurations, good performance results
  - Board 634
  - Board 633

- v1.1 new production (3 boards): I2C configuration issues
  - Board 503
  - Board 504 *(heavily modified)*
  - Board 505

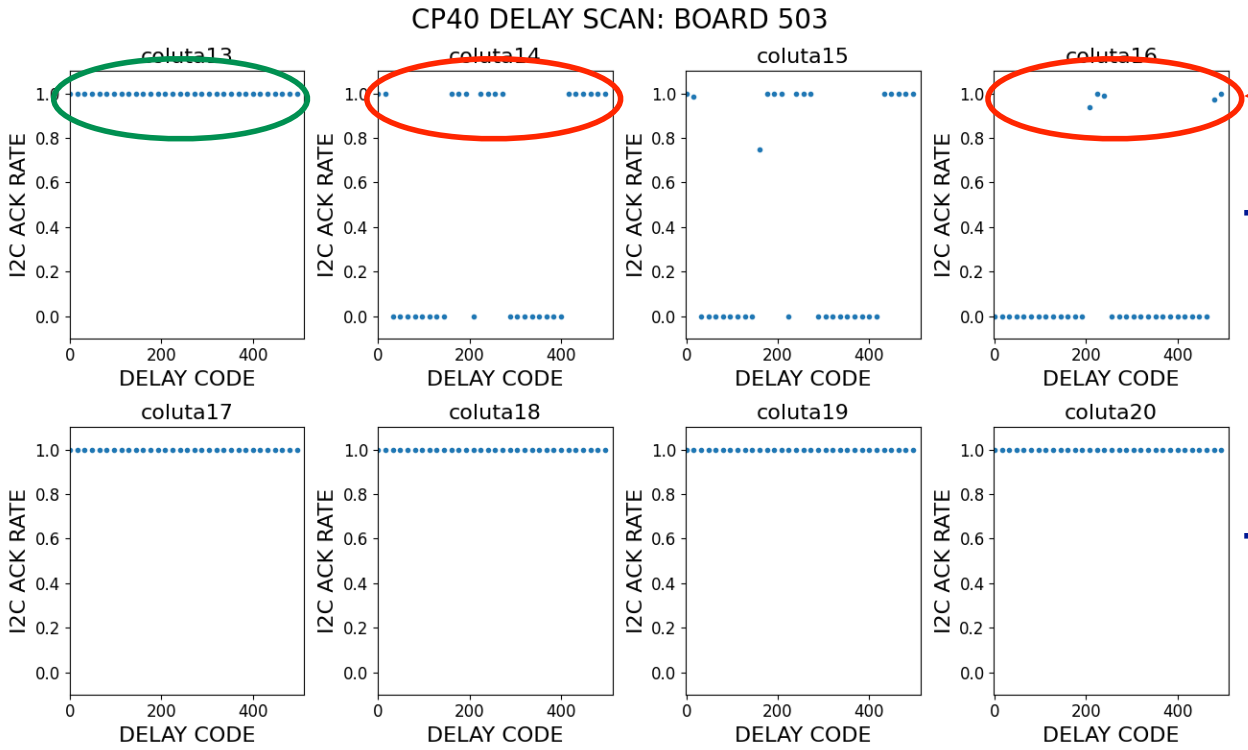
I2C Distribution



- lpGBT M1 bus connects to/configures everything
  - SCL = I2C clock (programmable O(100s) kHz)
  - SDA = bidirectional I2C data line
- CP40: 40 MHz clock on COLUTAs/LAUROCs that strobes lpGBT signal
  - Programmable delay for COLUTAs vs. fixed phase LAUROC (with inversion)
  - Phase is important w.r.t. SCL: **clock scan** needed to find good operational phases

CP40 Phase Scans

- Clock scan procedure: 512 delays of CP40 w.r.t. SCL, power cycle in between
  - Step size = lpGBT register (LSB = 48.8ps). 512 delay codes corresponds to 24.9856ns. Here every 16th delay code was tested (roughly 781ps per step), 32 of these steps is 25ns
- Expect some failures (where CP40 and SCL edges are perfectly aligned)
  - On reliable board halves, COLUTAS respond to 100% of I2C commands for most CP40 delays, even when LAUROCs are not reset
  - On board halves with prominent I2C issue, almost all delay codes do not work



I2C commands **not** acknowledged for most CP40 delay values

I2C commands acknowledged for most CP40 delay values

I2C Status

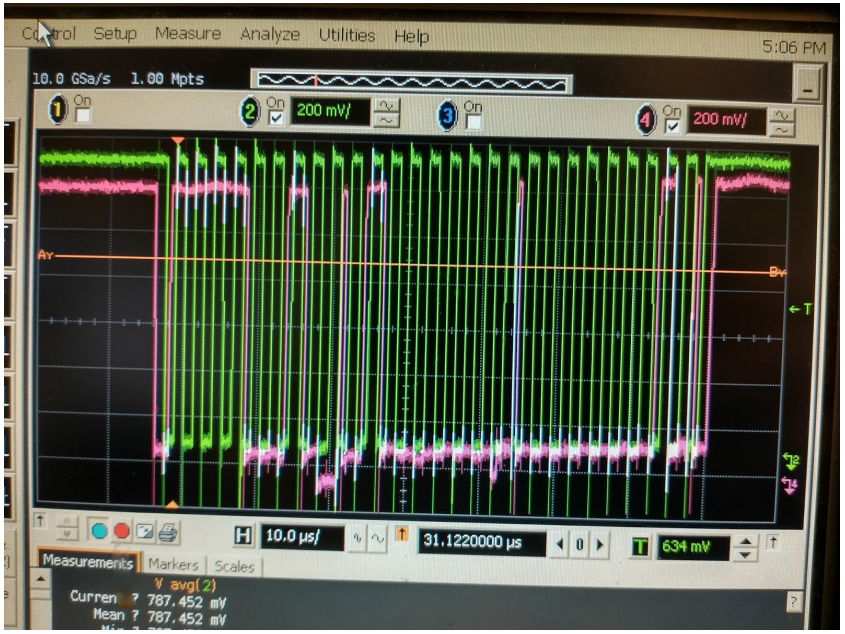
- Good stability (on 2 good boards): for example, overnight test gave 0 errors / 3.55 million writes to COLUTA register (Board 634)

! 3 new boards: many ASICs do not configure reliably…
  - Highly board dependent; some boards all devices acknowledge every I2C command, others acknowledge none
  - SCL + CP40 clock edges are clean with well-defined phases (below)
  - LAUROCs are more affected: very low success rate → **focus today on COLUTAs**

Scope Images of Good I2C

- Full length I2C Byte Write command to a COLUTA with ACK

Full I2C Byte Write with ACK



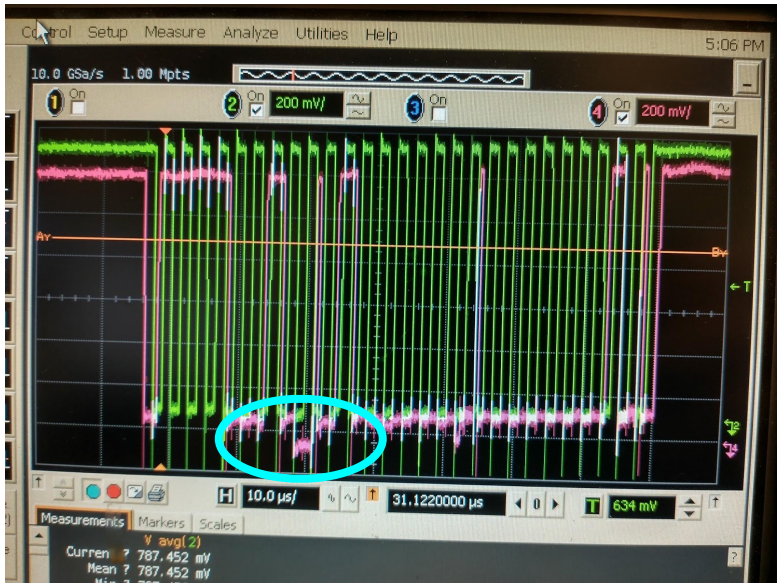
START condition

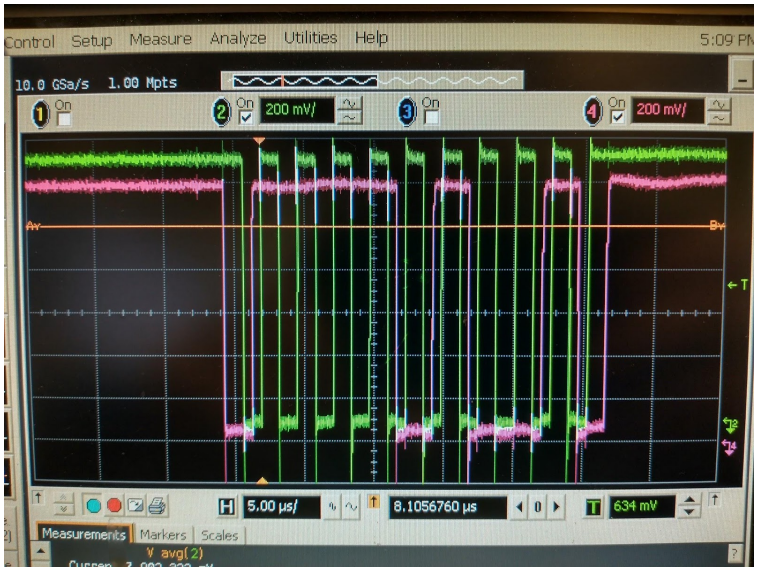
STOP + ACK

Description of I2C Problem

- All COLUTAs tested respond with ACK to the first I2C command after reset
- COLUTA "freeze-up": second I2C command is not acknowledged, no register written
  - Soft reset recovers COLUTAs, but then clears the registers that have already been written
- In some cases, COLUTA I2C communication can also freeze if valid I2C command sent to non-COLUTA device on same bus (eg. LAUROC)

Full I2C Multibyte Write with ACK → **Second I2C Bye Write: NO ACK**





A Possible Cause: SDA Glitch

- Observed “glitch” in lpGBT SDA occurring when lpGBT drives bus
- **Reflection from mismatched impedances in termination of M1 bus?**
- **Does glitch cause freeze-up?** If spike occurs right around threshold, does it give fake start signal to COLUTA?
  - Then COLUTA remains in waiting state and never receives data, nor stop command
  - Watchdog circuit should protect against this…

Hypotheses

- SDA edges are too slow? vary value of pullup resistor → no impact
- Related to I2C_MULTI_READ_EXT from LAr Week? → no, problem occurs even with single byte write

➡ **Current best hypothesis: glitch causes "false start" which leads to COLUTA freeze-up:**

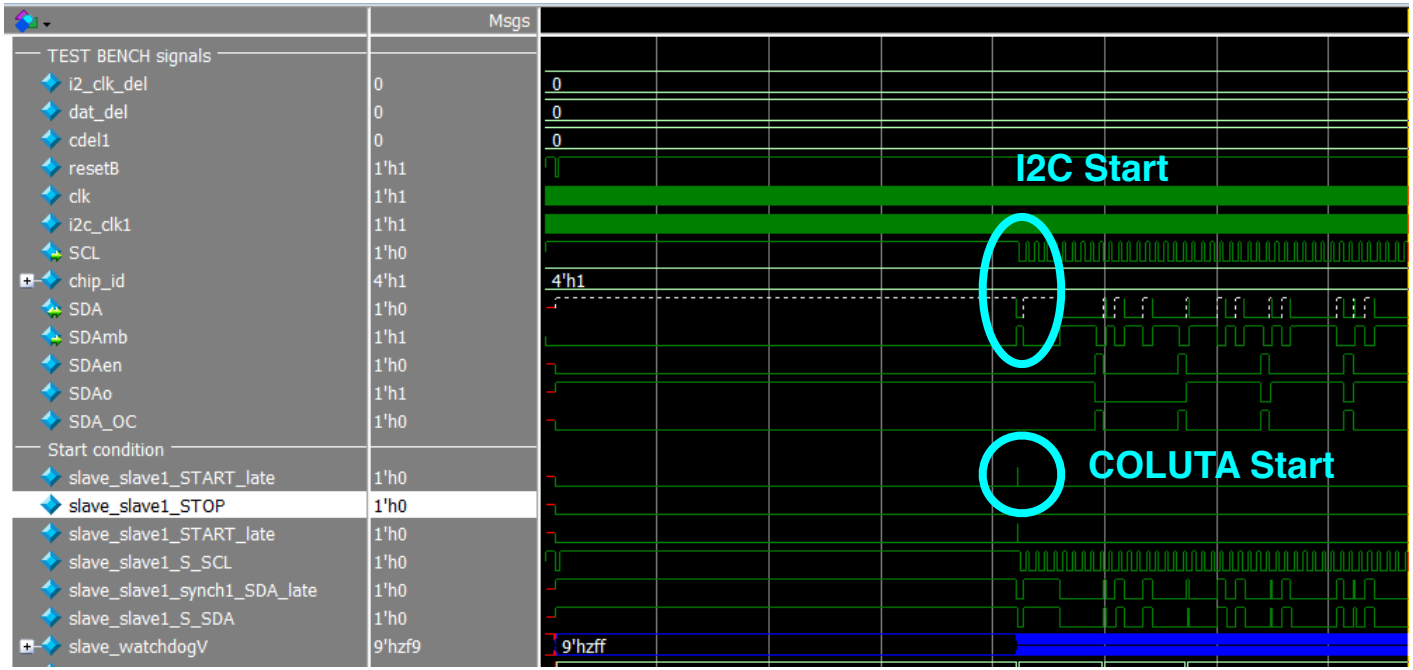
- If glitch is from reflection on M1 bus: try varying termination of line? → no impact
- Glitch is from noisy power? (could explain why LAUROCs fare worse; fewer power inputs for I2C block than COLUTA + coherent noise fix meant adding a lot of capacitance to LAUROC power lines)
  - Bypass DCDC, bring in clean power from linear supply (not a switcher) → improved (increase in number of working CP40 phases) but not on par with old boards
- Glitch is from interference of other devices?
  - Cut bus between COLUTAs and LAUROCs, put each on its own separate lpGBT master → does not fully resolve problem

CV3 Simulation Studies

- Simulation suggests a false Start could indeed cause COLUTA to noACK the next valid I2C comment (despite fact that watchdog is working), since subsequent Start is not recognized

Good I2C Transaction

Start and Stop is detected all bytes ACK-ed

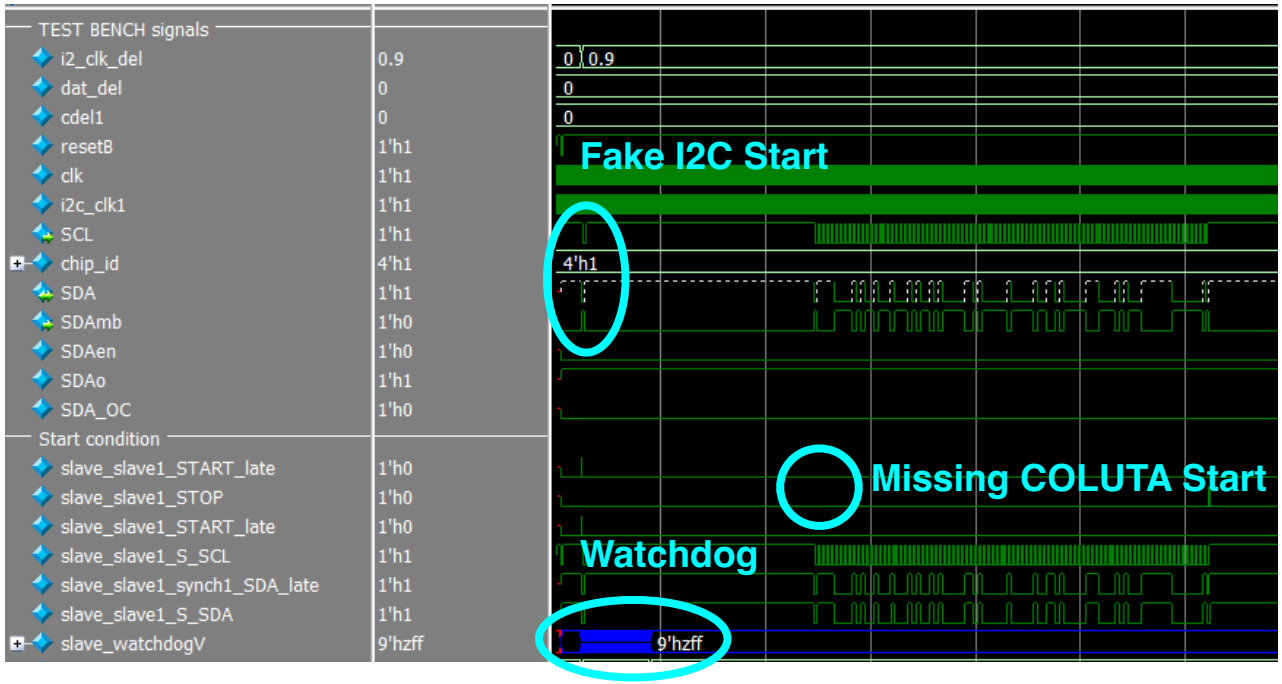


CV3 Simulation Studies

- Simulation suggests a false Start could indeed cause COLUTA to noACK the next valid I2C comment (despite fact that watchdog is working), since subsequent Start is not recognized

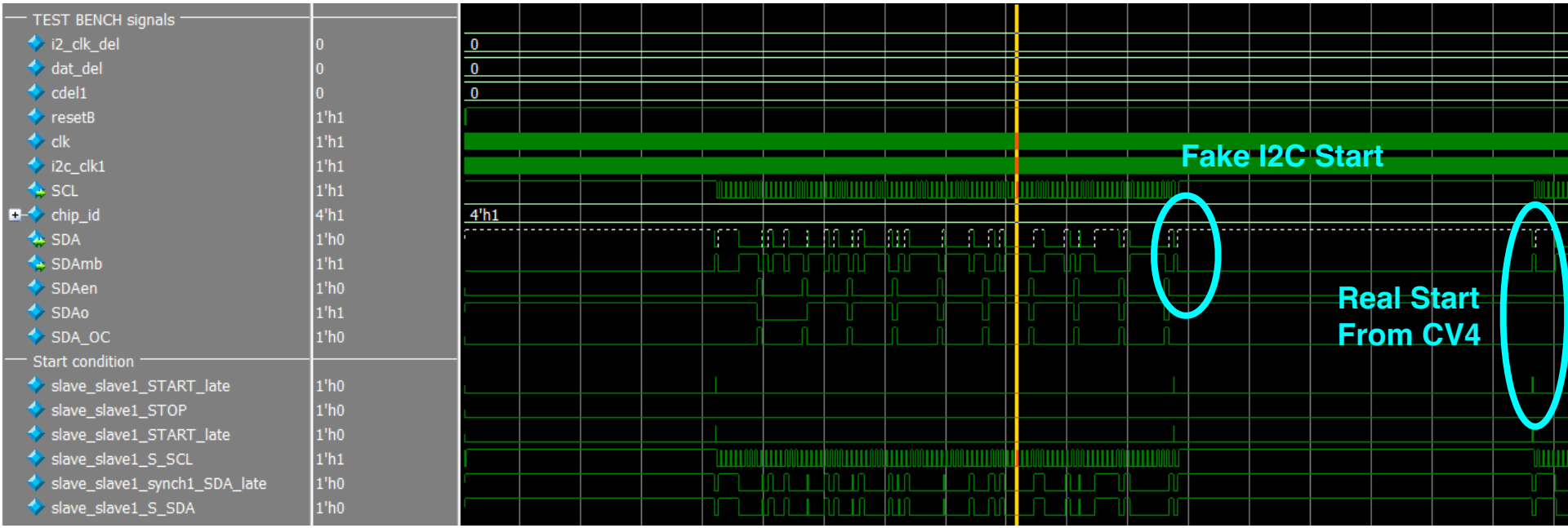
False start after reset

First I2C command no ACK-ed, start condition not detected. Second I2C command: start condition detected, operation ACKed



CV4 Simulation Studies

- **Fixed in CV4**: preliminary simulation verifies false start hypothesis; shows next start after false start is acknowledged



Options to Prove “False Start” in Hardware

- **Goal**: reproduce conclusions seen in simulation studies in hardware
- Disconnect SCL and SDA from lpGBT12 or 13 & connect to devices where we can independently program SCL and SDA, and purposely introduce a false Start.

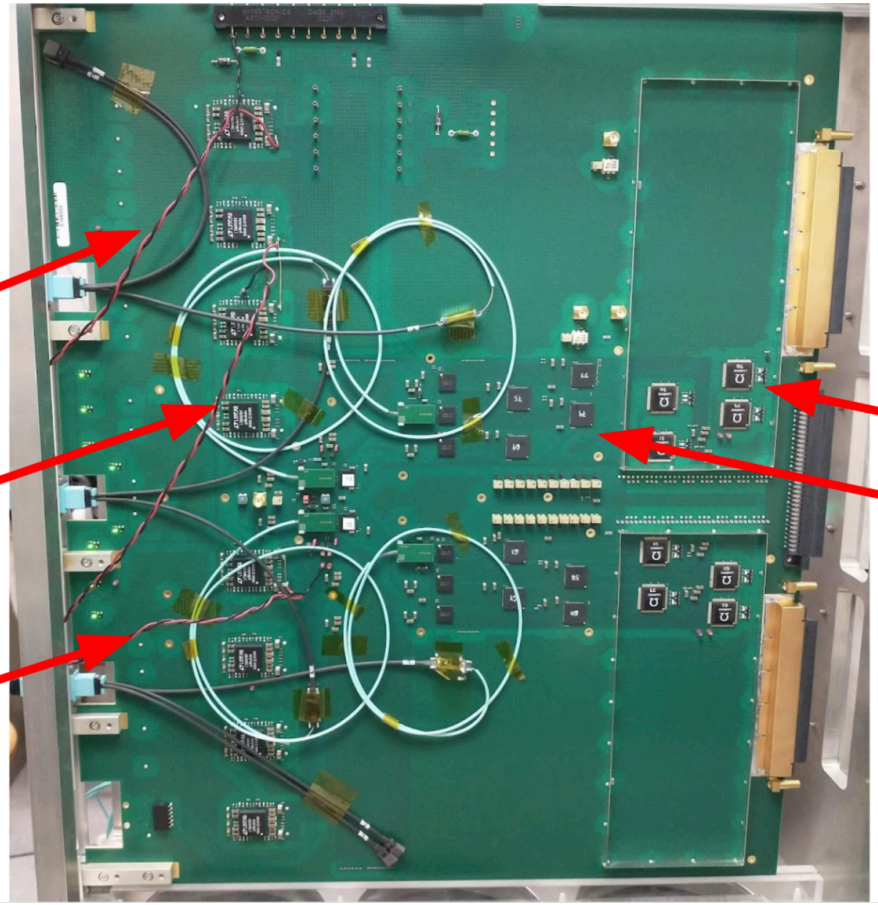
- Possible devices:
  - 2-channel external AWG,
  - 2 unused GPIO pins of an on-board lpGBT via “bit-banging”
  - External USB dongle
  - Some commercial device, eg. Raspberry Pi
  - Use Cv3 test board, where I2C commands generated by FPGA

- We are considering best & most time effective approach to pursue

List of Performed I2C Tests

- ✓Checked quality of contacts, touch up solder
- ✓CP40:
  - Clock delay scans for COLUTA
  - Turned off CP40 clocks to other chips
  - Adjustment of parameters: drive strength, pre-emphasis, etc.
- ✓Add filtering to SDA/SCL lines
- ✓Using SCL CMOS driver vs. not
- ✓Use of external I2C dongle to program devices
- ✓Scanned many parameters:
  - Drive strength of bus: spurious clock ticks/slow SCL rise time? (preference for low)
  - SCL frequency: 100 kHz — 1 Mhz (preference for 400 kHz)
  - Adjustment of SDA/SCL line termination impedances: disable 1kΩ resistor, OR adding 50Ω + 1pF across (no impact)
- ✓Power
  - Extra filtering to COLUTA + LAUROC Vdd, shorting of Vdd inductors
  - Changing DC-DC converter frequency (240 kHz nominal — 780 KHz)
  - Took each device off central power & tried on their own clean power
- ✓Interference from other devices
  - Disconnect power, DC-DC switching frequency, hold in reset
  - Cut traces, put COLUTAs & LAUROCs on their own bus

Board 504 Modifications



2.2V External Supply for data lpGBTs

2.2V External Supply for COLUTAs+LAUROCs

4.5V External Supply

LAUROCs 17-20 placed on separate I2C bus from COLUTAs 17-20

I2C Summary & Steps Forward

- I2C configuration status:
  - COLUTA: 2 old v1.1 boards configure (100% of commands work at most CP40 phases); 3 new v1.1 boards are unreliable
  - LAUROC: similarly less reliable on 3 new boards (2 original boards functional)

- ***The good news:*** two changes in CV4 that we can expect will improve the situation
- Based on current hypothesis of glitch → false start → freeze-up
  1. If glitch generates false start, Schmitt trigger (present on CV4) will help
  2. Synchronization logic of I2C in CV4 is different w.r.t. CV3: we should not have issue of not recognizing valid start, even after false start
- Starting to validate in simulation, CV4 chips & testboards arriving ~end of year

Conclusions

- 5 v1.1 Slice Testboards have been under testing at Nevis since August 2021
- I2C configurations are sporadically unreliable on 3 new boards
  - Hypothesis: “glitch” in lpGBT SDA line causes a "fake start” in COLUTAs, which disrupts state machine & causes subsequent starts to not be recognized until after a new Reset
  - Simulations preliminarily support this, but more simulations and studies are being done
  - Changes implemented in CV4 expected to fix issues

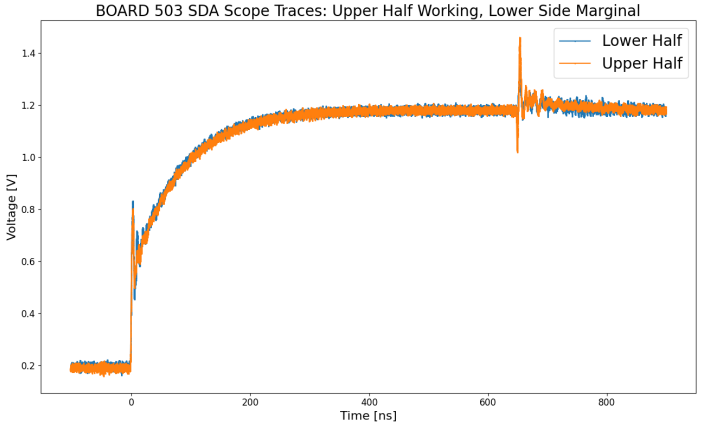
- Next steps:
  - Continue studying I2C on Slice Testboard
  - Once functionality is established, distribute v1.1 boards to collaborators: CERN, Milano, Brookhaven (already has v1.0)

Backup

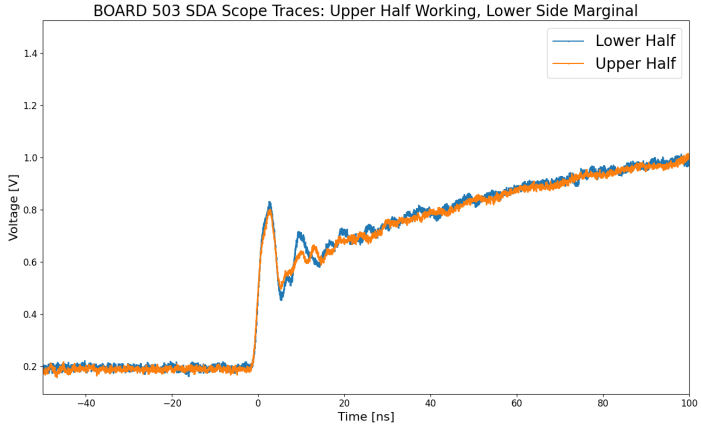
I2C SDA Comparisons

- Upper half = consistent I2C ACK; lower half = inconsistent

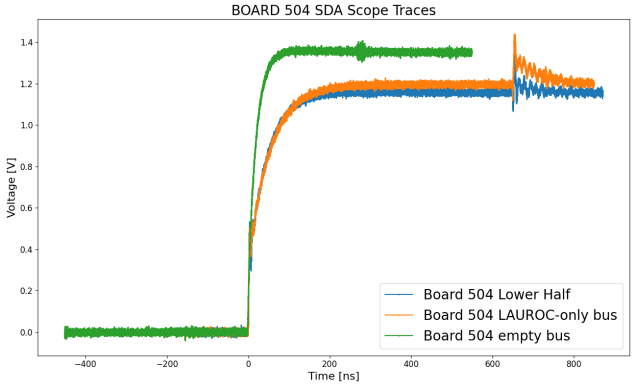
SDA Rising Edges

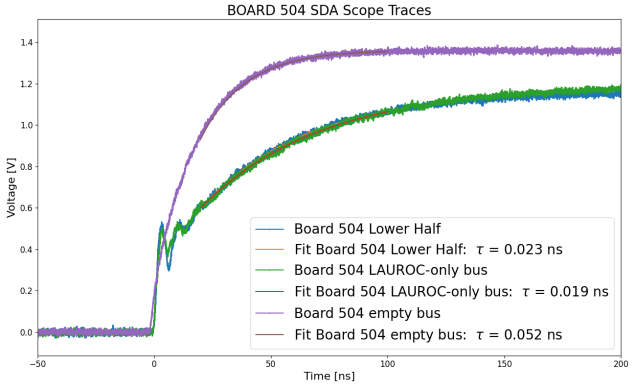


Close Up



No COLUTA





I2C COLUTA Comparisons

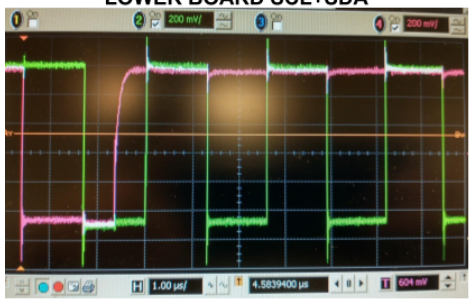
I2C NOT Working I2C Working

PIN	BOARD 503 COLUTA13	BOARD 503 COLUTA20
ADC VDDA	1.2V	1.20
VDDD DDPU	1.2V	1.197
CID 0	1.2V	0V
CID 1	0v	0V
CID 2	1.2V	1V
CID 3	1.2V	0V
GND	0V	0V
SCL high	1.21V	1.20V
SCL low	110mV	148mV
SDA high	1.18V	1.09V
SDA low	150mV	204mV
CP40N high	714mV	778mV
CP40N low	462mV	406mV
CP40P high	712mV	630mV (?!)
CP40P low	470mV	554mV (?!)

-all values measured wrt nearest point on PA/S cage

COLUTA13 - I2C NOT WORKING

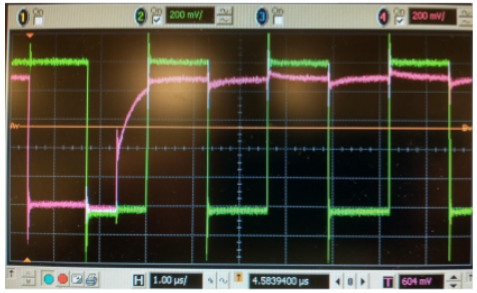
LOWER BOARD SCL+SDA



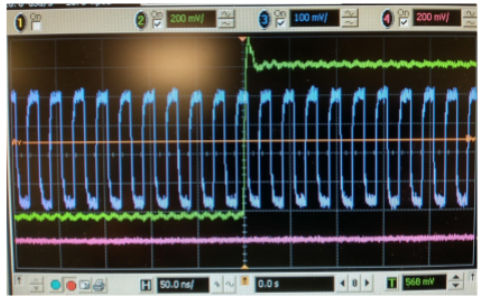
SCL + CP40 ON COLUTA13 PINS

COLUTA20 - I2C WORKING

LOWER BOARD SCL+SDA



SCL + CP40 ON COLUTA20 PINS

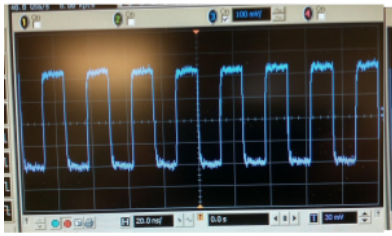


COLUTA CP40 Comparison

- No features observed in clock traces to explain i2c issues

I2C NOT Working

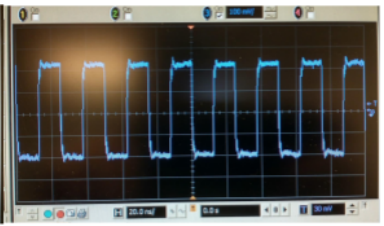
COLUTA13 CP40



COLUTA14 CP40

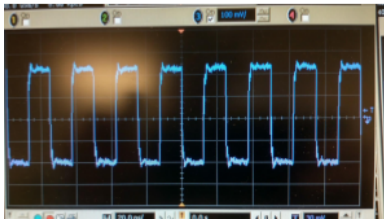
COLUTA16 CP40

COLUTA16 CP40

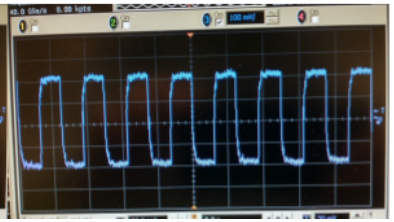


I2C Working

COLUTA17 CP40

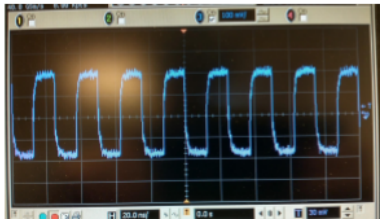


COLUTA18 CP40



COLUTA19 CP40

COLUTA20 CP40



Strobing CP40 vs. SCL

SCOPE Traces of SCL + CP40N/P for I2C Write Command to Board 503 COLUTA 14 NO ACK

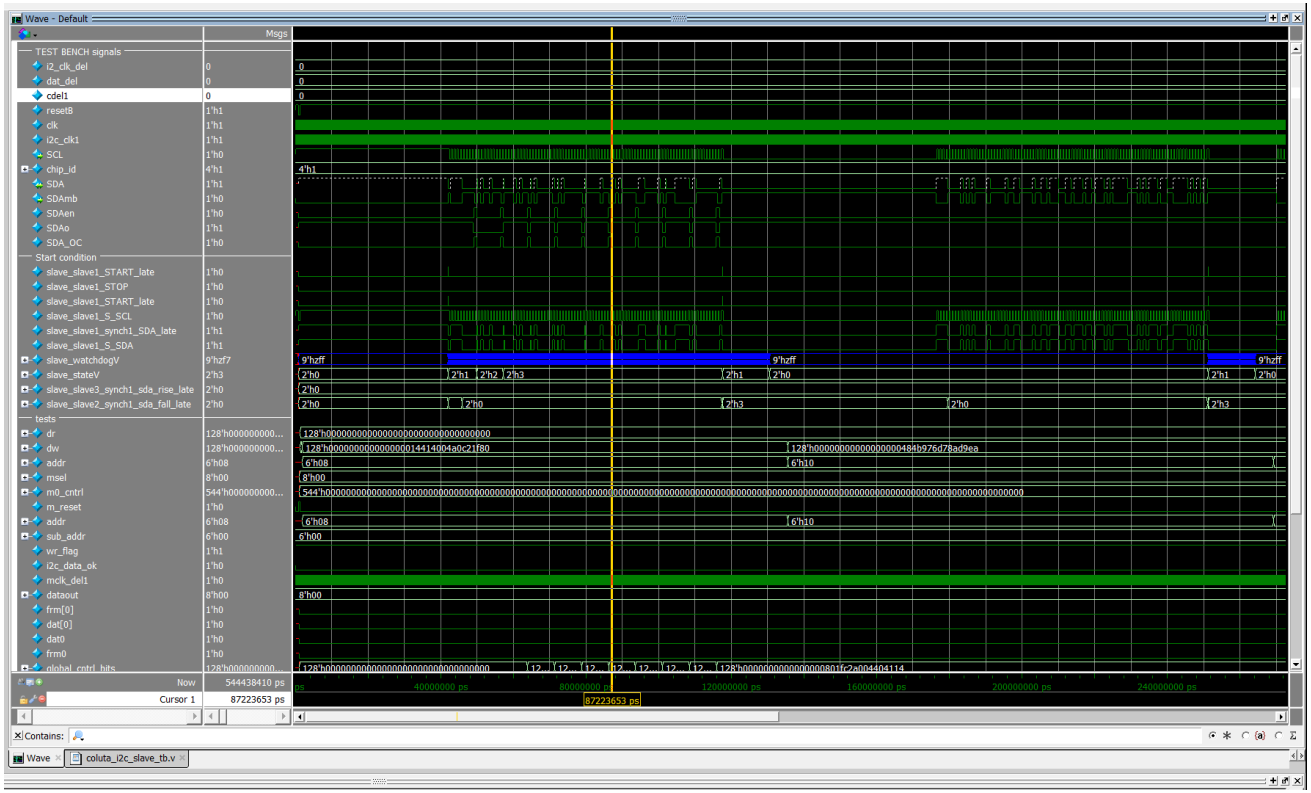
- Board 503, COLUTA 14
- I2C SCL and CP40P/N signals going to COLUTA14 for a command that failed
- Sharp rising edge on the SCL signal looks like it's inducing a transient on the CP40 signal?

Simulation Studies

- Simulation suggests a false Start could indeed cause COLUTA to noACK the next valid I2C comment (despite fact that watchdog is working), since subsequent Start is not recognized

False start after each command

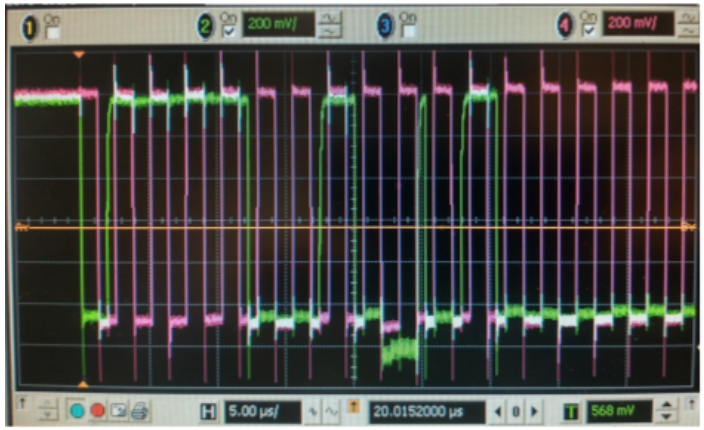
Makes next command not ACK-ed. First command ACK-ed but because of false start after each command no next command ACK-ed



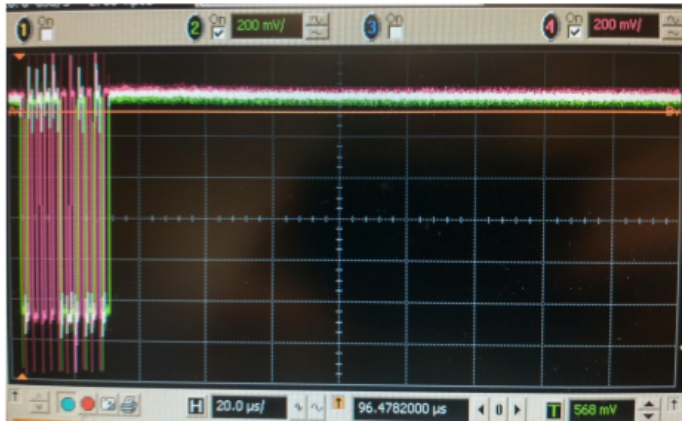
Role of Reset

BOARD 503 COLUTA13 I2C SDA+SCL SIGNALS

FIRST I2C COMMAND AFTER COLUTA13 POWER CYCLE + RESET



SUBESQUENT I2C COMMANDS UNTIL NEXT COLUTA13 POWER CYCLE + RESET

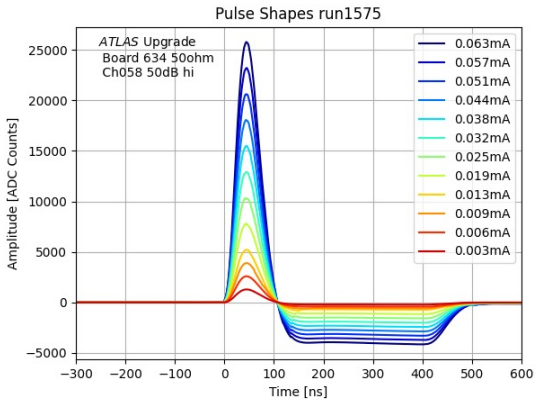


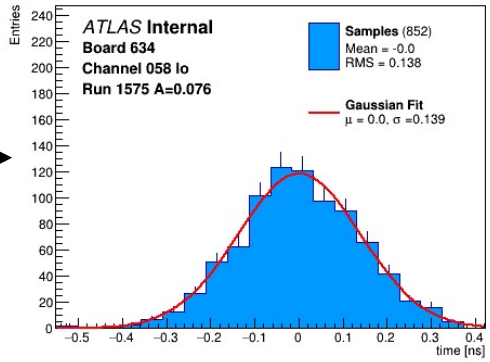
Configuration/Setup Scans

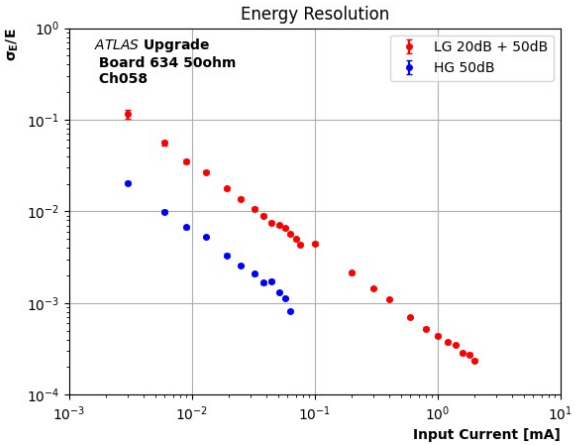
	BOARD 503	Board 504	Board 505	Board 633	Board 634
All VTRxs work	YES	YES	YES	YES	YES
lpGBT configure current draw	1.163A	Multi-supplies no	1.065A	1.354A	1.342A
SDA pullup lpGBT12 side	1kOhm		1kOhm	1kOhm	1kOhm
SDA pullup lpGBT13 side	2kOhm		1kOhm	1kOhm	1kOhm
SDA Drive	LOW	LOW	LOW	LOW	LOW
SDA Pullup	NO	NO	NO	NO	NO
SCL Drive	LOW	LOW	LOW	LOW	LOW
SCL Pullup	NO	NO	NO	NO	NO
SCL CMOS	YES	YES	YES	YES	YES
Frequency	400kHz	400kHz	400kHz	400kHz	400kHz
Voltage	DCDC	External upper, lo	DCDC	DCDC	DCDC
SCL resistor	YES	NO	YES	YES	YES
LAUROC I2C isolated	NO	YES	NO	NO	NO
Other chips reset/no clks	YES	YES	YES	YES	YES
CP40 only for test chip	YES	YES	YES	YES	YES
COLUTA17-19 inductors removed	???	???	???	???	???
DCDC convertor freq	Nominal (240kHz	500kHz (?)	Nominal (240kHz	Nominal (240kHz	Nominal (240kHz

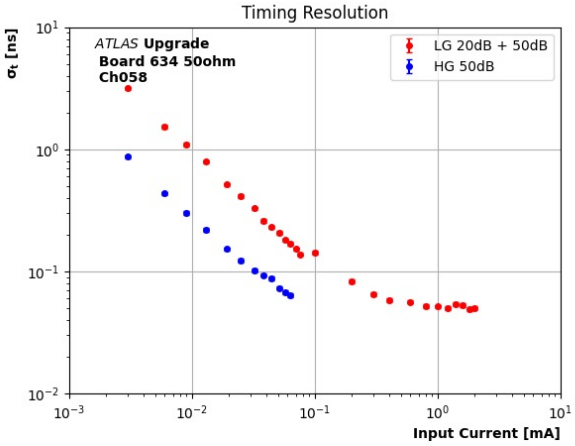
Single Channel Performance (50Ω)

- ADCs derives 40 MHz CLK from FELIX, which is synchronized to AWG signal source
  - Pulse HG+LG channel at amplitudes spanning dynamic range
  - Combine different attenuations at input to access full range
  - Apply OFCs to repeated measurements, perform gaussian fit on results to obtain Energy, timing resolution
  - Energy resolution ~.02% for large pulses
  - Timing resolution ~ 50 ps (dominated by system CLK jitter, not by Slice Testboard)

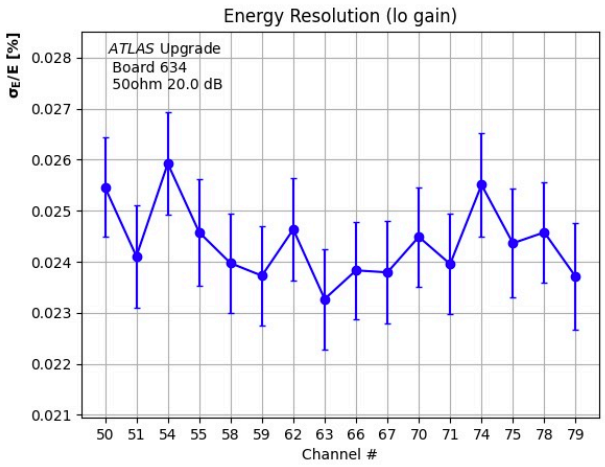


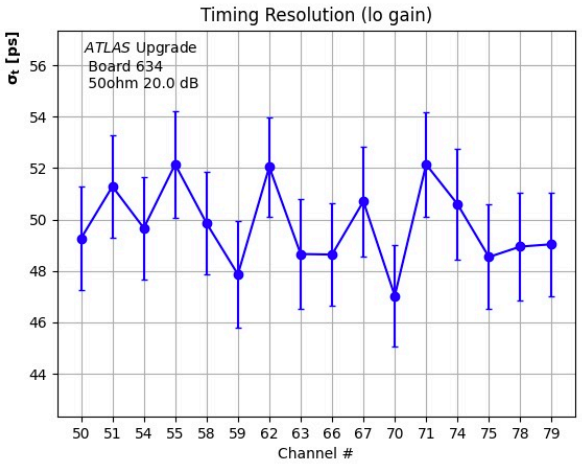


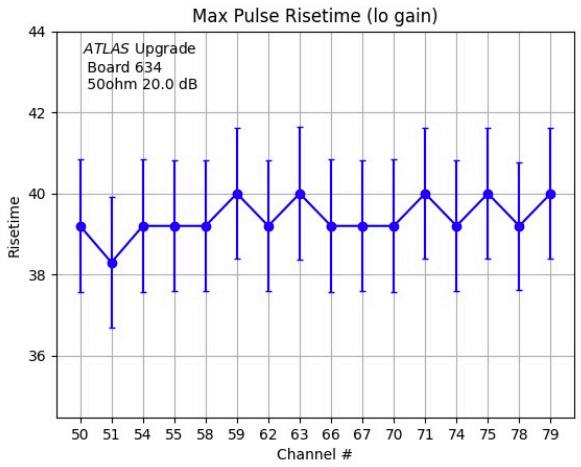




Multi-Channel Performance (50Ω)

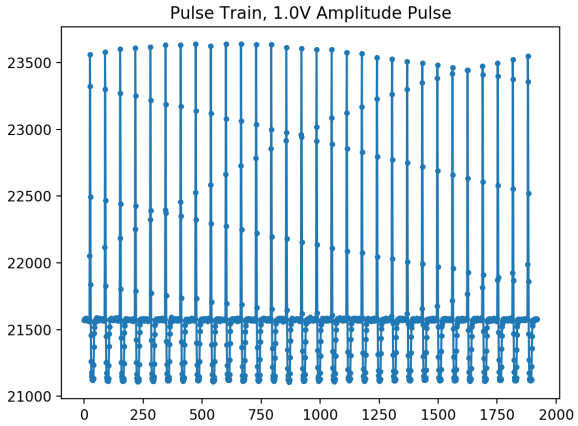


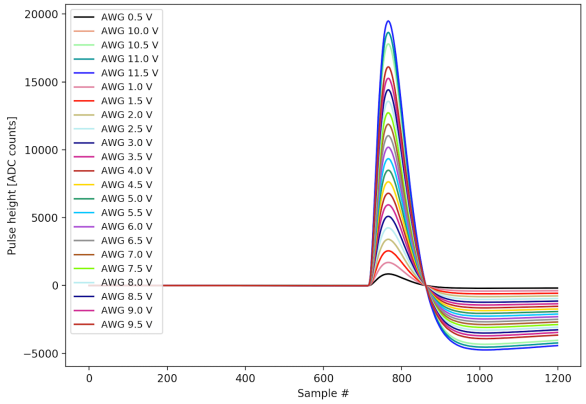


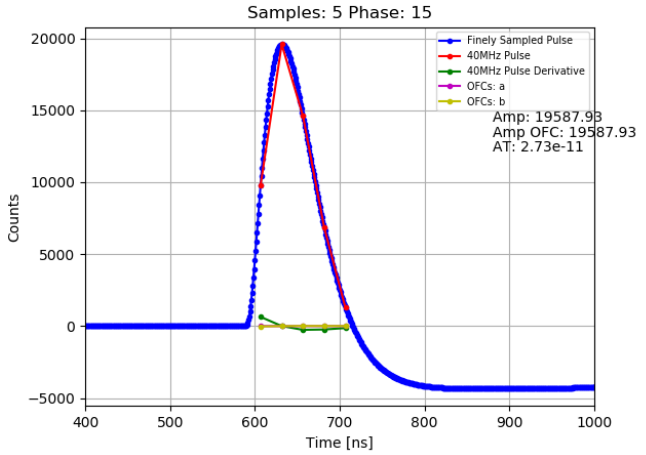


- Automated **Full-board characterization** software package in development
- Energy resolution, timing resolution, and pulse risetime for large pulse heights consistent across channels on PCB #634

LAr Pulse Analysis







AWG sends a pulse train of known amplitude to ADC chip, sampled at different phases

→

Pulse train is interleaved to reconstruct fine pulse for each amp. Check that maxima and zero point match across amplitudes

→

Samples from one phase (containing peak) and derivatives are used to calculate OFCs, then used to find energy and timing of each pulse