BCM1F Beamgas & Albedo Logic

Outline

- Aim
- Existing solution (wired NIM modules)
- Scheme
- Requirements
- Using a V1495 module
- Detailed Proposal
- Current Status

BCM1F Beamgas & Albedo Logic - Aim

Aim

Gated Counting for separation of

- Beamgas events
- Afterglow events

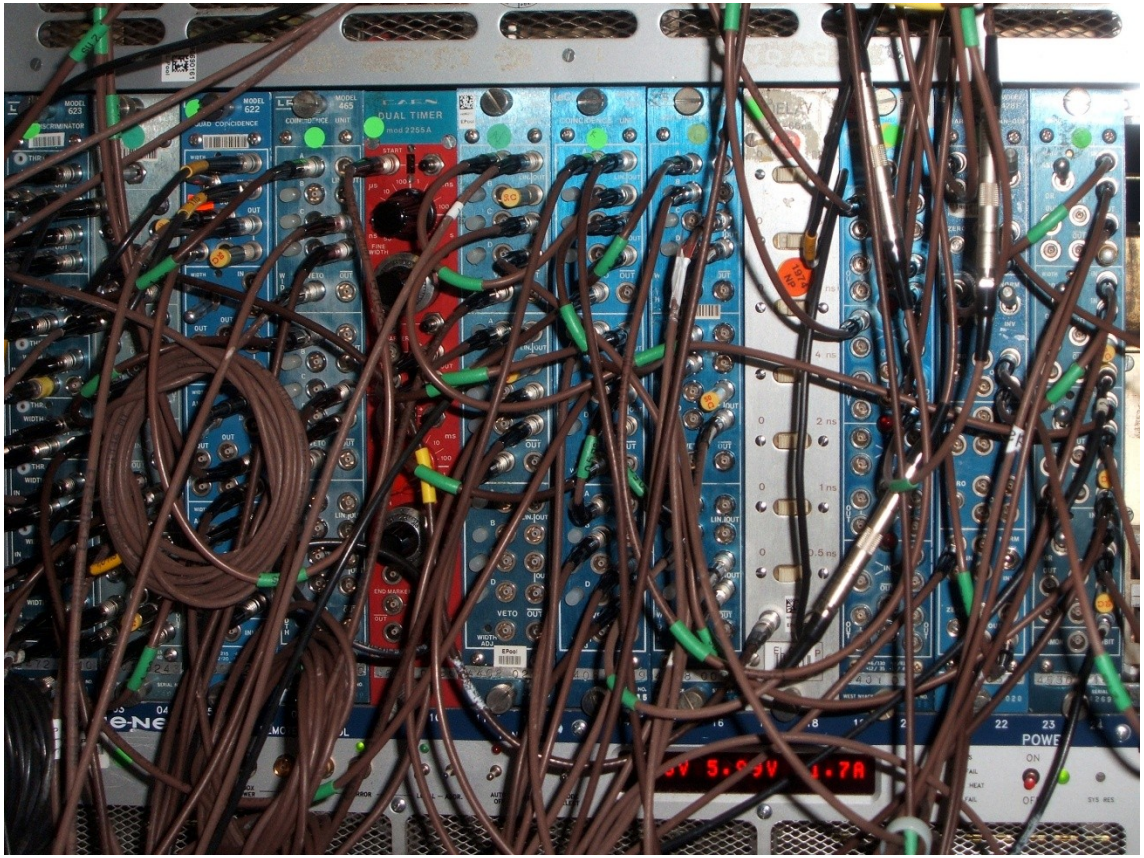
see talks earlier this day!

Existing solution: A crate full of wired NIM modules

BCM1F Beamgas & Albedo Logic - Weird Solution

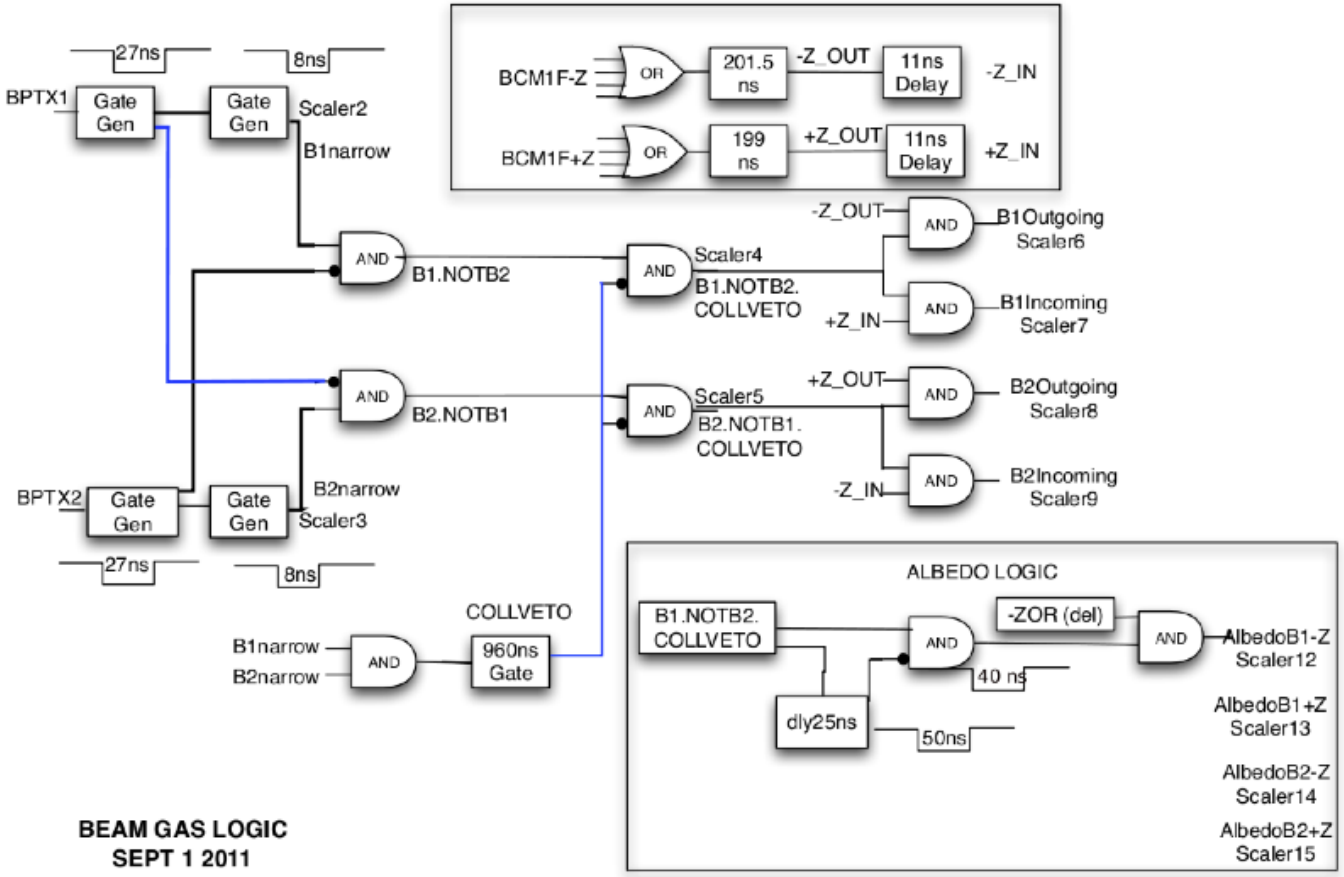
Existing solution: A crate full of wired NIM modules

- lots of shaky switches
- bunches of cables
- …



BCM1F Beamgas & Albedo Logic - Schematic

Logics Scheme



… and more requests to come (?)

BCM1F Beamgas & Albedo Logic - Requirements

Requirements:

2 **BPTX** inputs

8 **BCM1F** inputs

4 outputs - **Albedo** Scalers

8…32 outputs - **Beamgas** Scalers

8 **Gates** (8ns … 50ns, 960ns)

21 **Delays** (11ns … 202ns)

BCM1F Beamgas & Albedo Logic - Proposal

Solution with V1495:

Why?

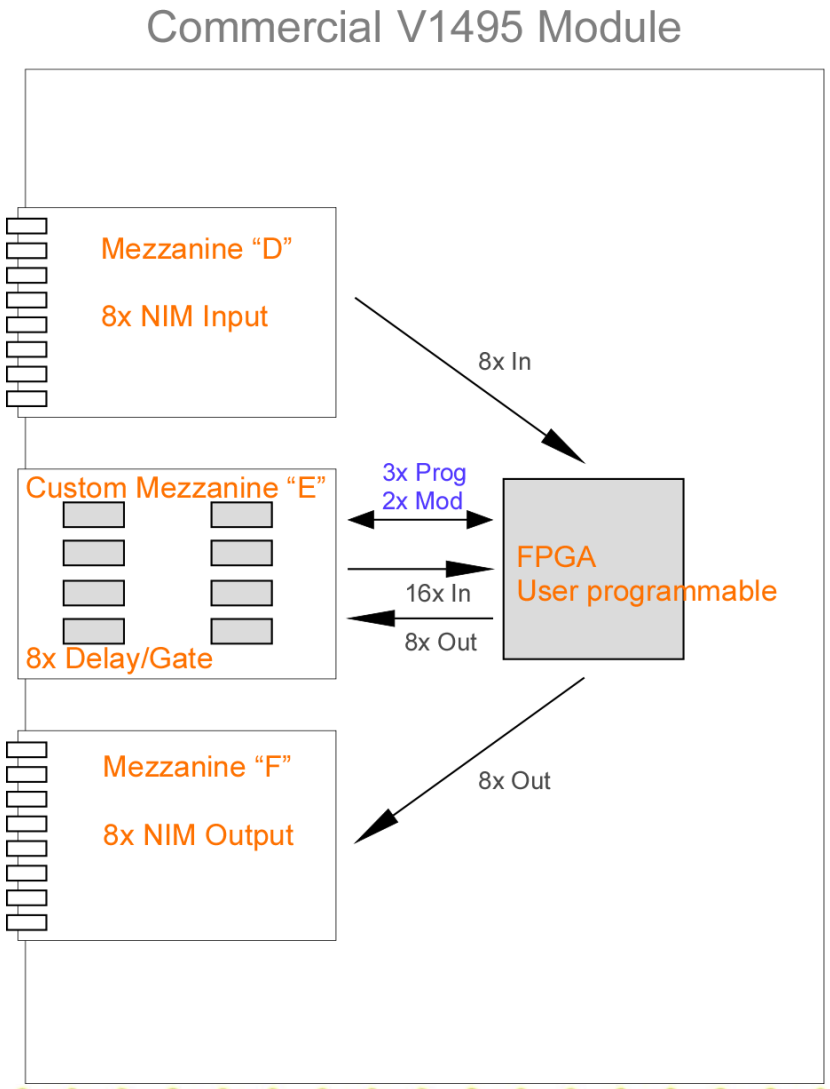
- NIM and ECL I/O provided
- VME interface
- ease of programming

How?

- custom designed mezzanine
- 8 chips per board
- choice between gate or delay
- fully programmable

Problems:

- bidirectional interface to FPGA
- 25MHz fmax (chip specs)



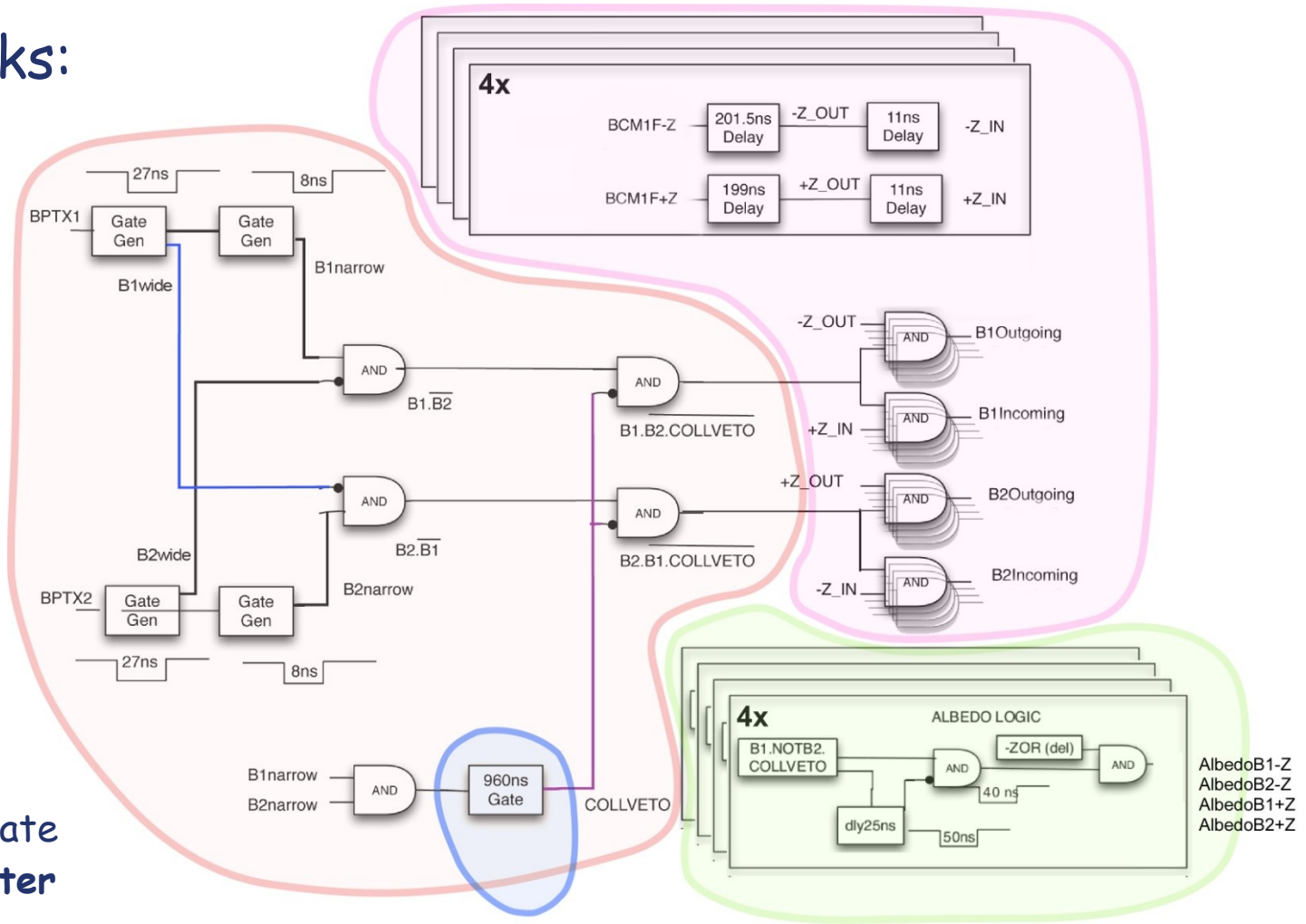
BCM1F Beamgas & Albedo Logic - Proposal

3 basic blocks:

Beam Gas

BPTX1

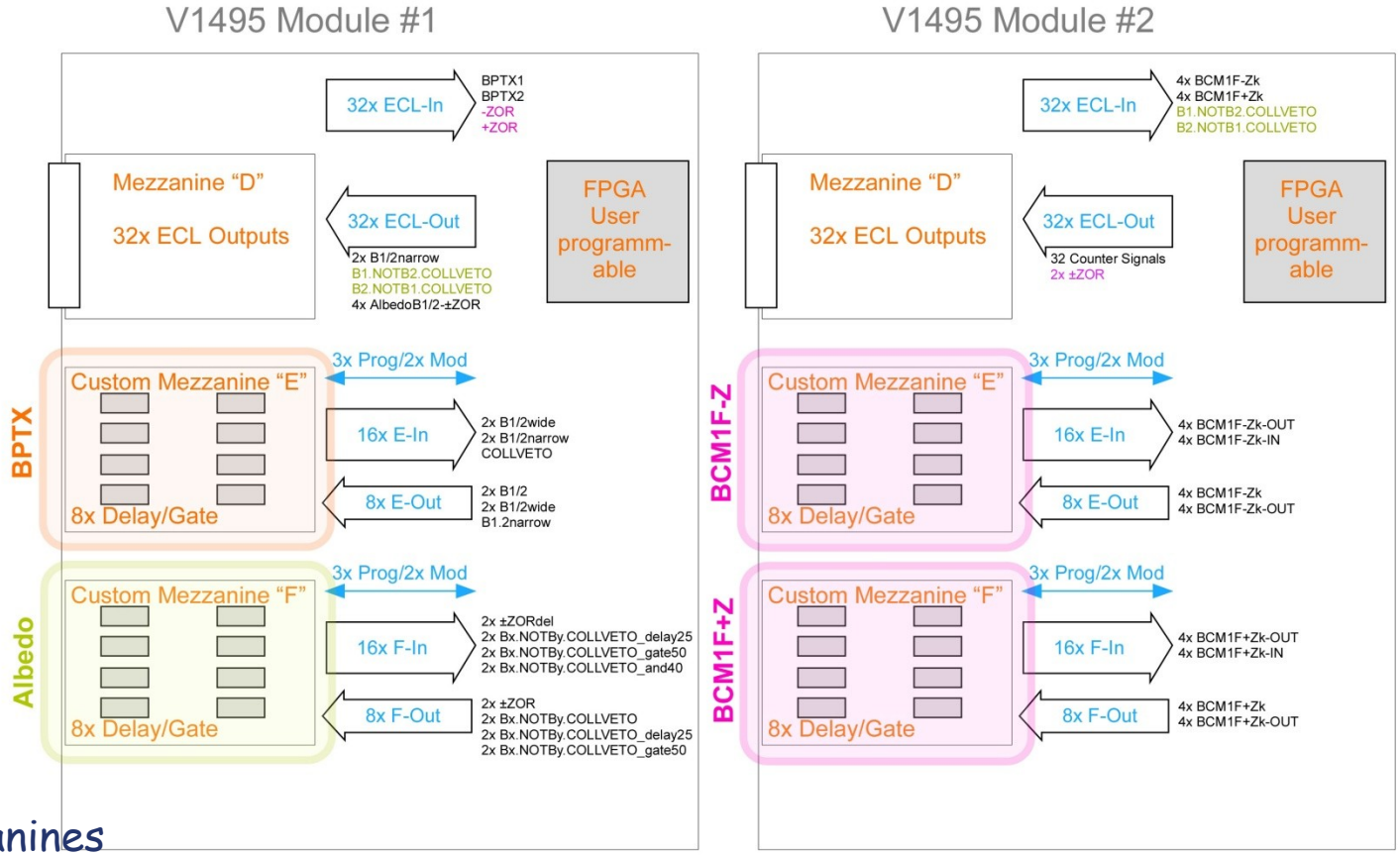
Albedo



plus one wide gate
utilizing a **counter**

BCM1F Beamgas & Albedo Logic - Proposal

Distribution among 2 modules:

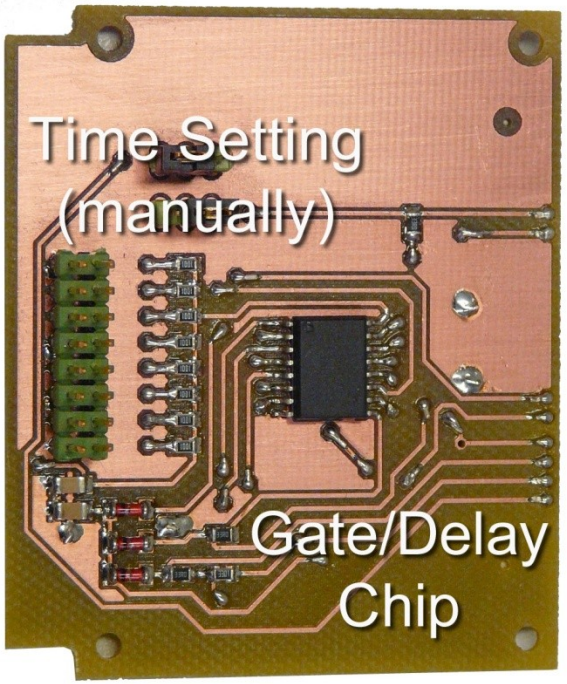


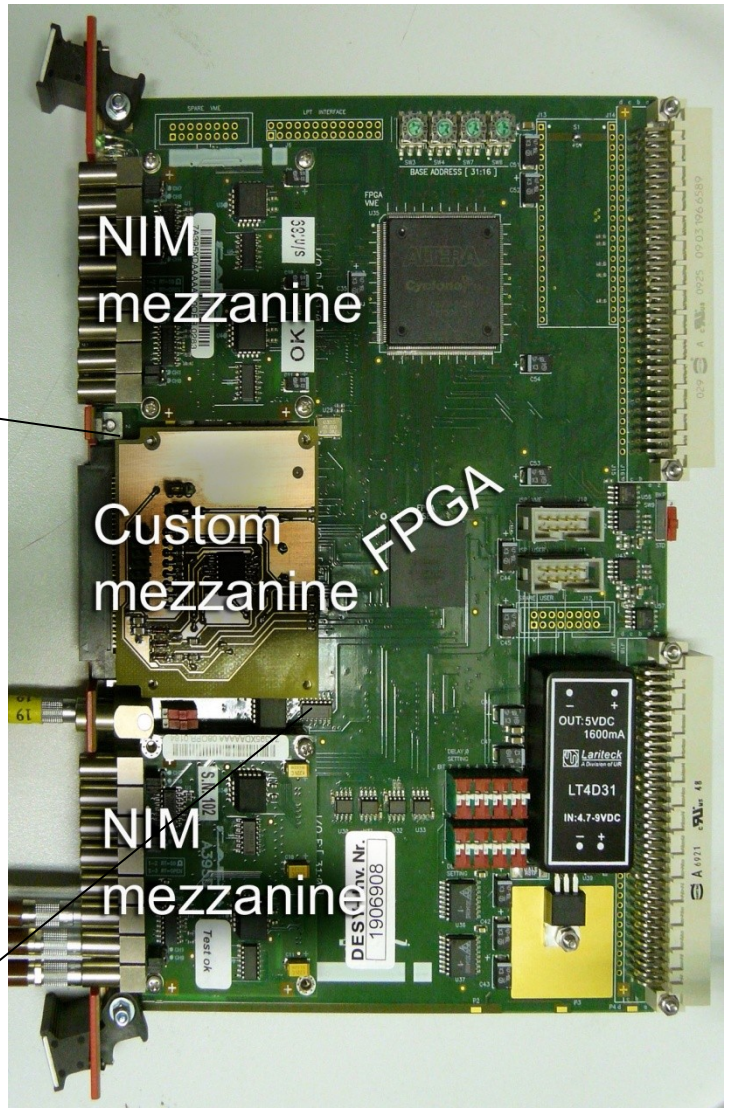
2 types of mezzanines
2 types of delay chips (255ns, 65ns)
ECL outputs to scalers, LVDS links between modules

BCM1F Beamgas & Albedo Logic - Status

Prototyping mezzanine

- test bidirectional ports to FPGA
- test serial protocol for setting delay/gate width
- test frequency limit
- test calibration





BCM1F Beamgas & Albedo Logic - Status

Serial Protocol

- registers for 8 chips and state machine implemented in FPGA
- readback to be tested

Bidirectional I/O and 3.3V interface

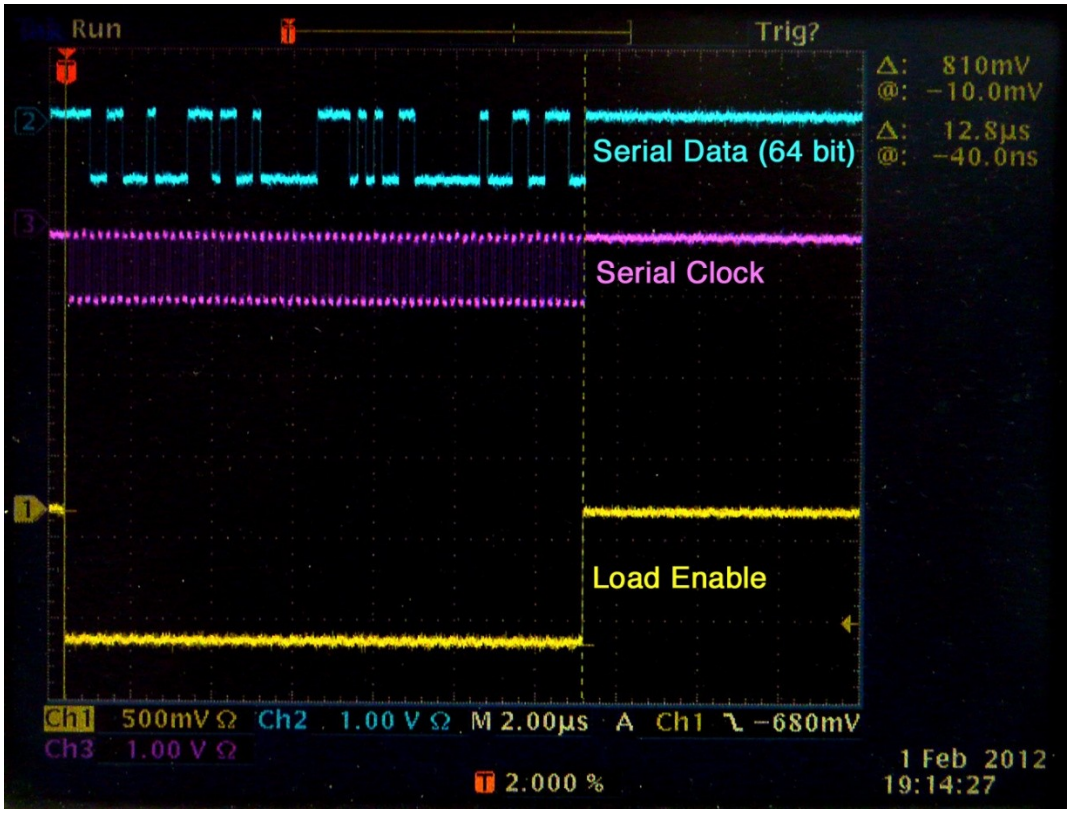
- under development

Timing Tests

- due

8-chip version

- design started



Thank you for the attention!

Special thanks for challenging ideas and thorough discussions go to David Stickland.