

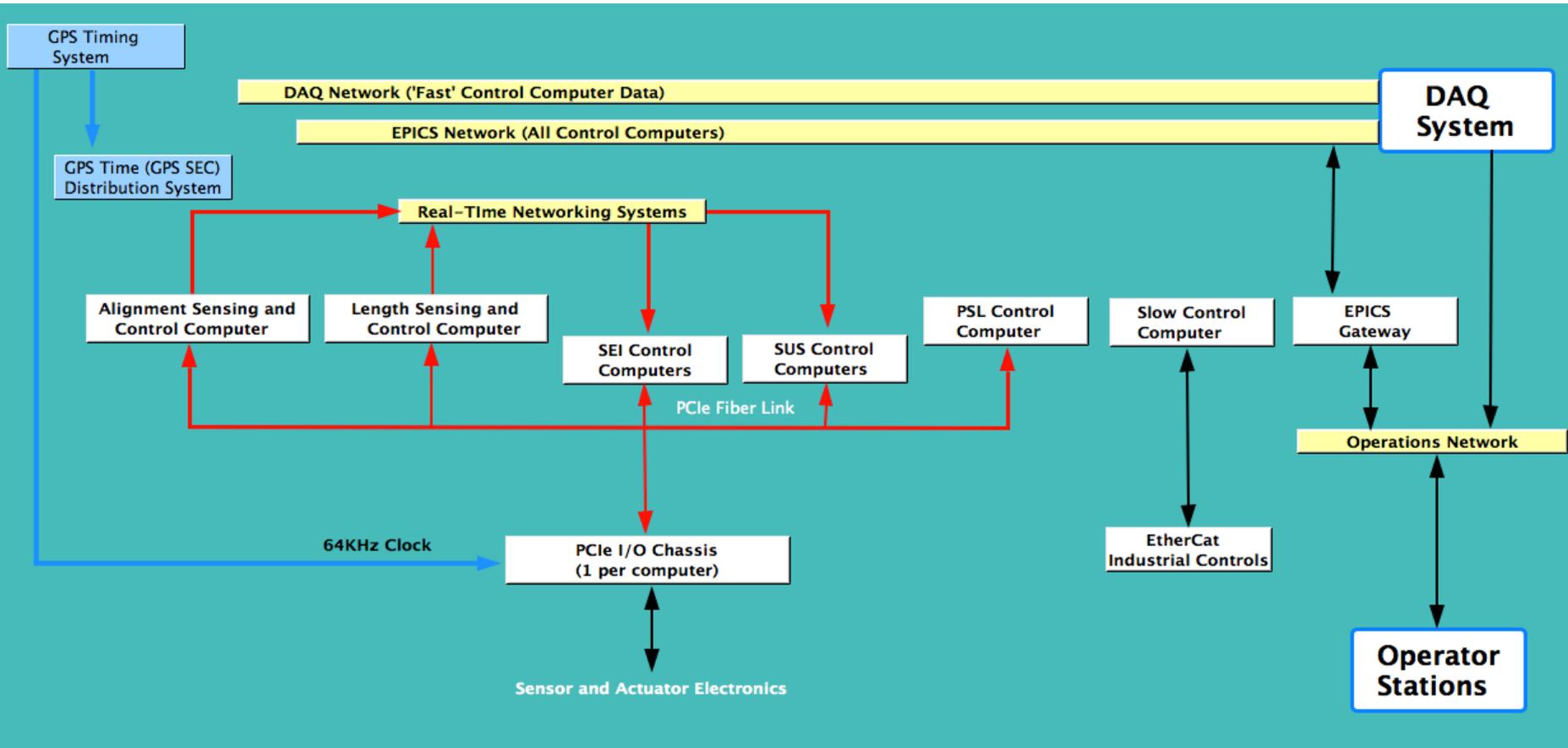


aLIGO CDS Pros and Cons

GWADW Conference
May 20, 2015

R. Bork

CDS Architectural Overview



Pros of CDS System

- Modular, common design, hardware and software
- Scalable
 - » Standalone, 'DAQ in the Box'
 - » Fully distributed system with inclusion of real-time networks
- Hardware makes use of commercially available and supported equipment as much as possible
 - » Beckhoff industrial controls for 'slow' control systems
 - » Standard server class computers and PCIe I/O system
- Software leverages off of iLIGO developments and software available from other sources, such as EPICS tools.
- Set of common software tools for use by controls engineers and scientists to develop control algorithms
 - » Real-time Code Generator (RCG) and Guardian

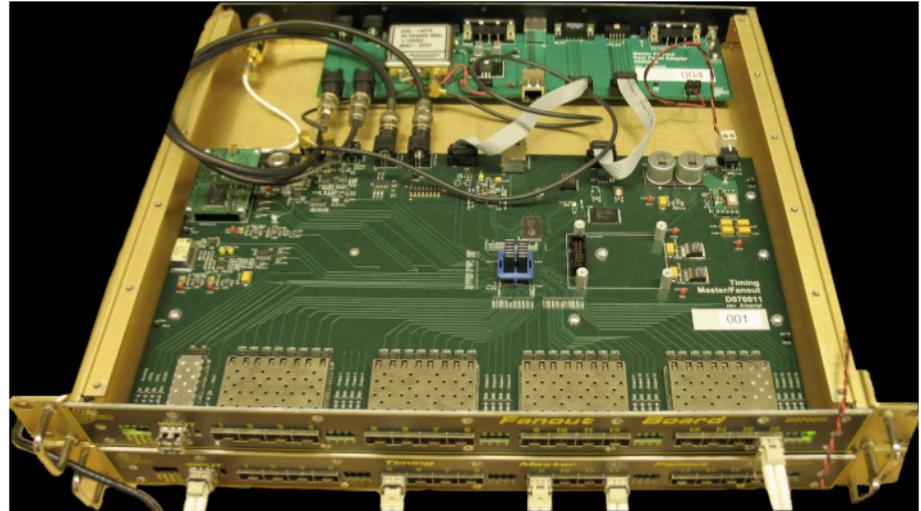
Timing Distribution System (TDS)

- Pros

- Central distribution point
- Auto length correction
- GPS time (seconds)
- Used to synchronize both CDS I/O and various RF sources
- Modular and expandable

- Cons

- Would like to have a PCIe version of the timing slave units
 - Turn clocks on/off on initialization and read time w/o need for extra cards



IRIG-B Timing Fanout

LIGOPC-500693 or LIGOPC-500694 provide accurate time information to computers.



Timing Slave provides accurate clocks At 65536Hz to ADC/DAC modules.

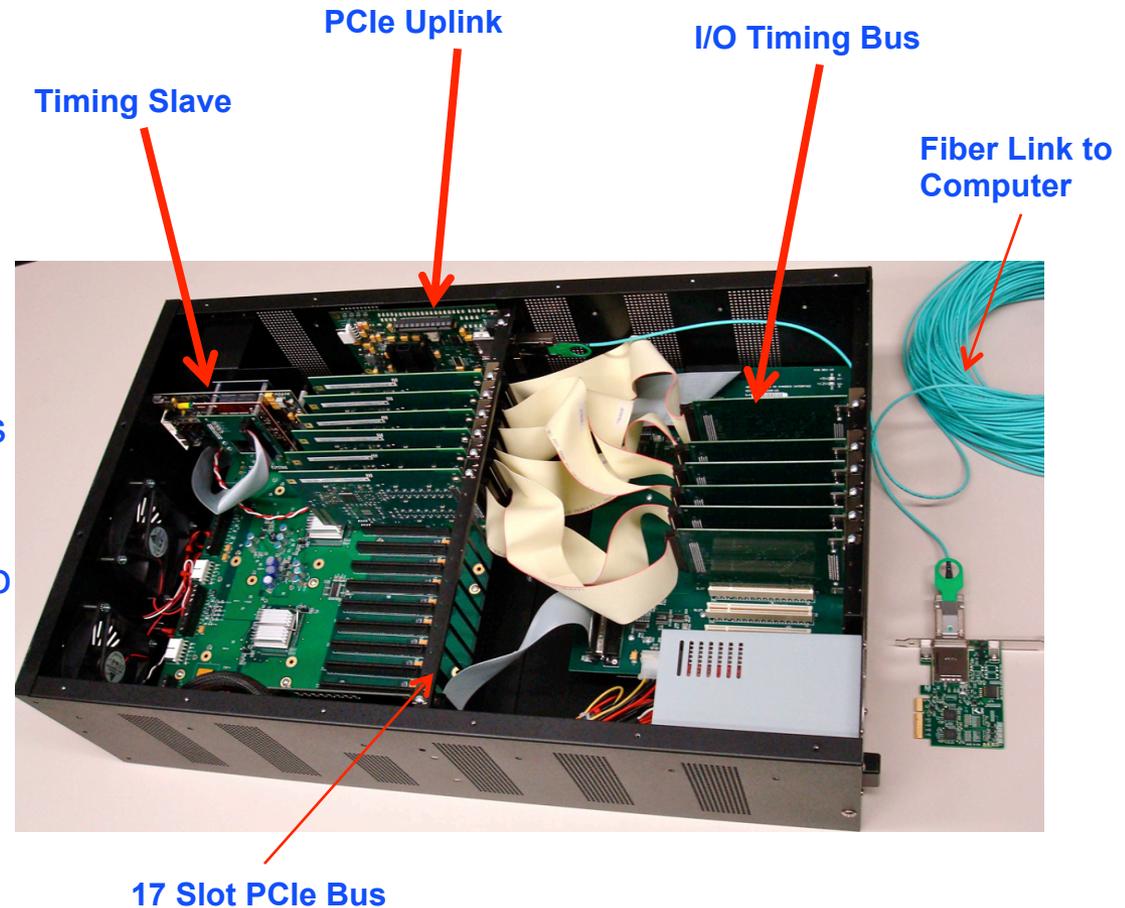
CDS I/O Chassis

Pros:

- Standard design
- Allows I/O to be separated from the computer and remotely located with electronics

Cons

- DC power supplies noisy
 - Being replaced
- Contains fans
 - Conductively cooled versions of PCIe cards now available
- Requires special electrical to fiber to electrical (EFE) links, ordered to length
 - Expensive and custom
 - New systems now available with simple fiber pair and length up to 10km



Pros:

- All equipment is Commercial Off-The-Shelf (COTS)
- Computers can be readily replaced with later technologies, as needed
- Standard Ethernet used for DAQ and supervisory control
- Real-time networks highly deterministic

Cons:

- RFM network used between end/corner stations has slower than desired I/O read/write times.

Standard Real-time Control / DAQ Computer



Reflective Memory Real-time Network



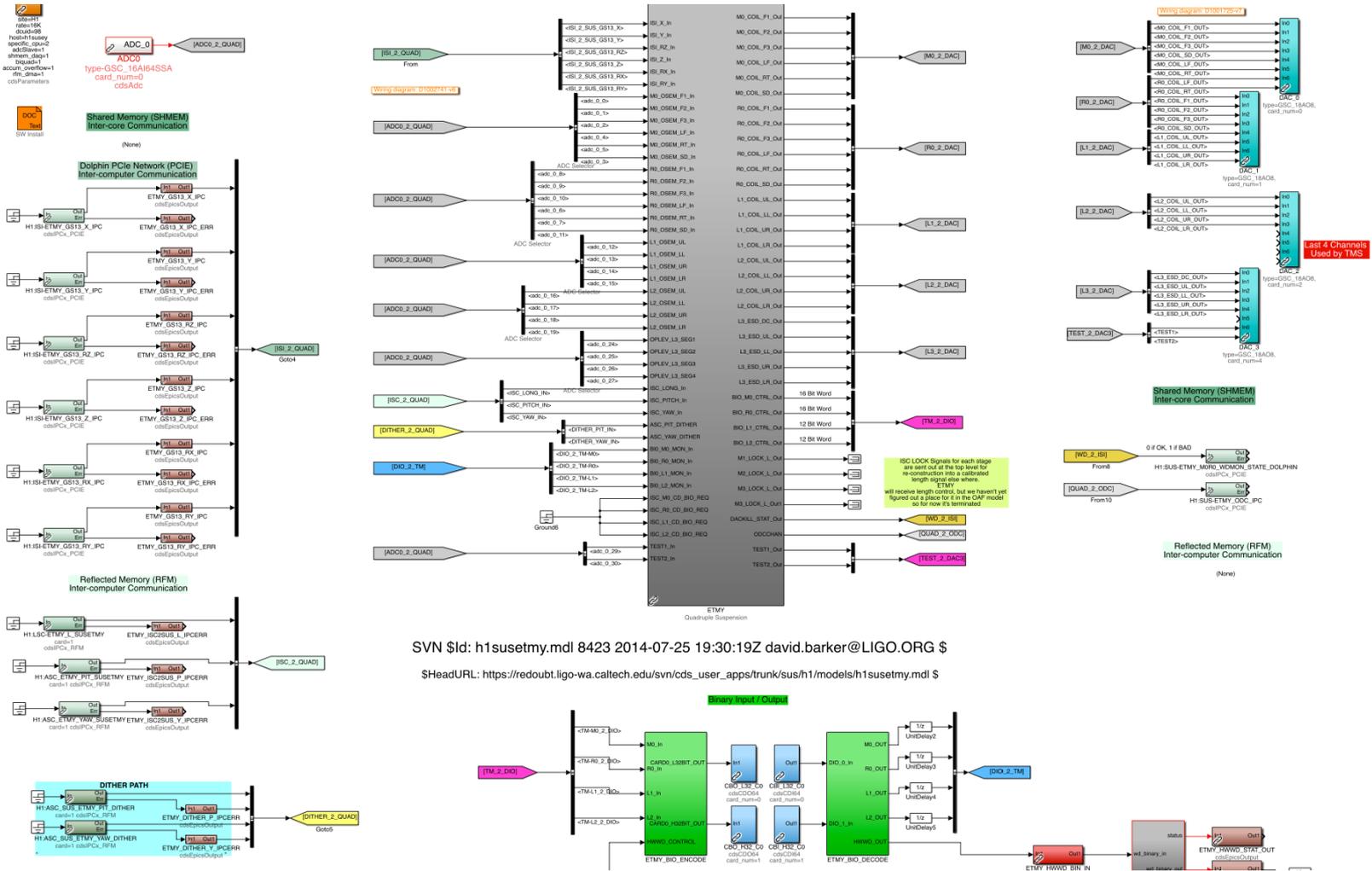
PCIe Real-time Network



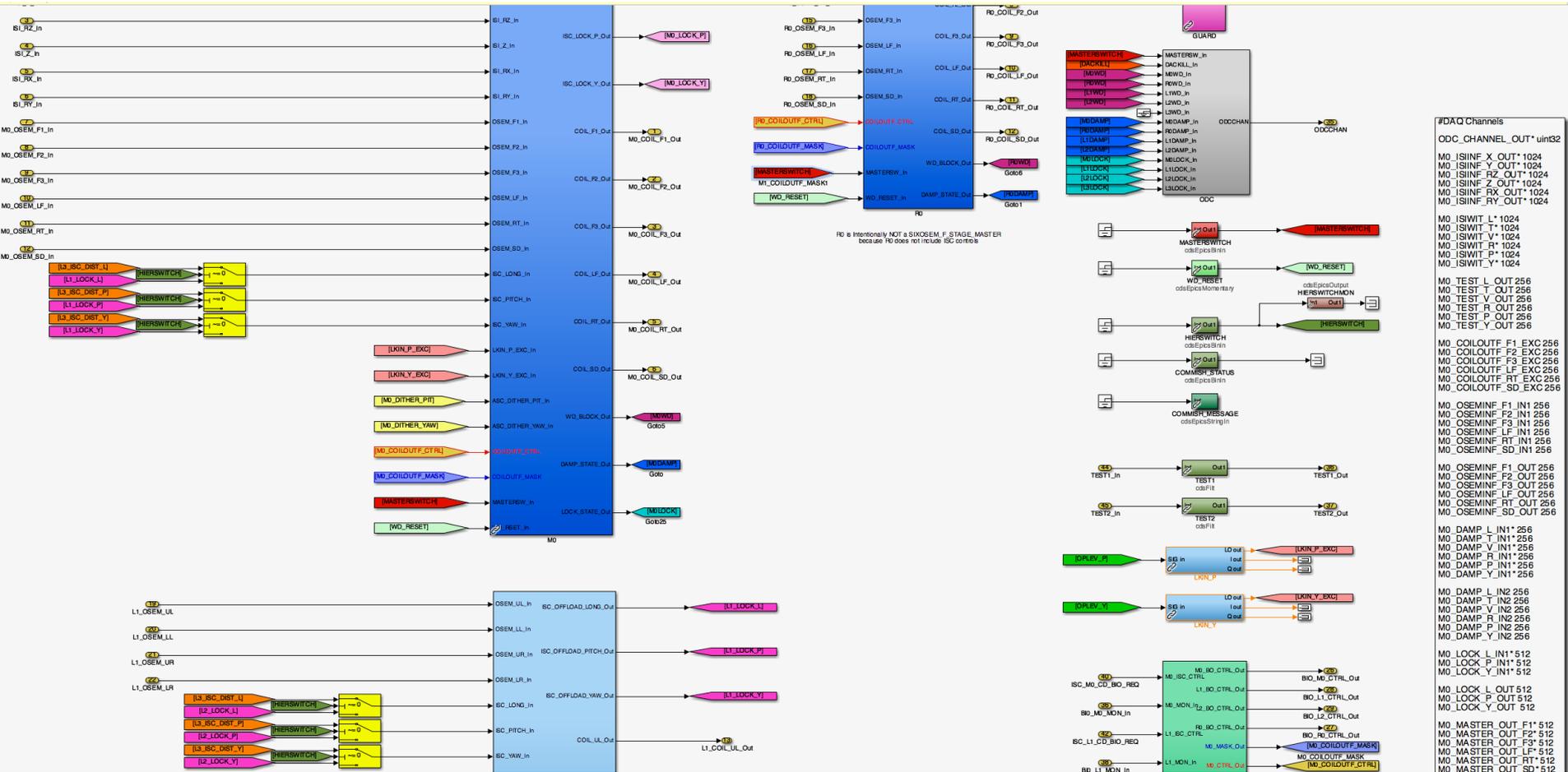
CDS Software

- Pros
 - » Designed as common tool set with standard interfaces
 - » DAQ software
 - Stand-alone to fully distributed
 - Network Data Server (NDS1 and NDS2)
 - Client libraries to support C/C++, python and Matlab
 - » Many Diagnostic tools
 - DTT, awgstream, Dataviewer, etc.
 - » Remote Data Access
 - Via NDS2 and MEDM web access
 - » Real-time Code Generator (RCG)
 - » Custom patch to GPL Linux to provide real-time capabilities
 - Saved >\$200K/year in licensing fees

RCG Example Quad Suspension (Top Level)



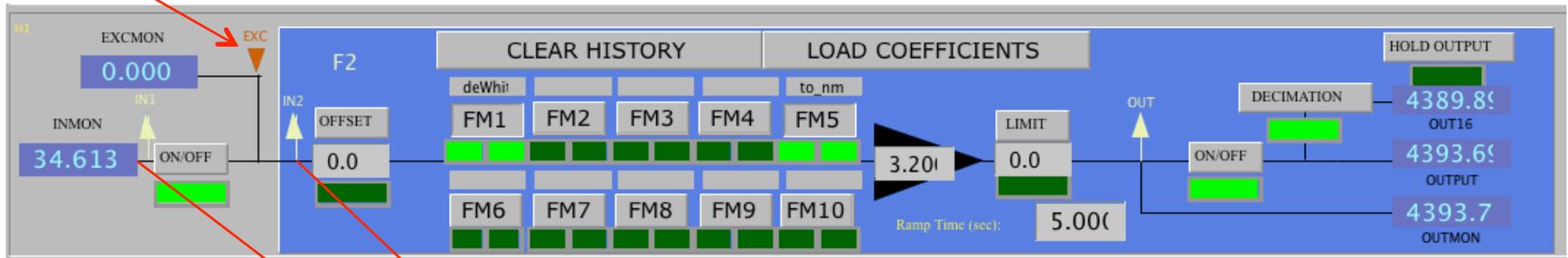
Quad Suspension Control Model (Level 2, Partial View)



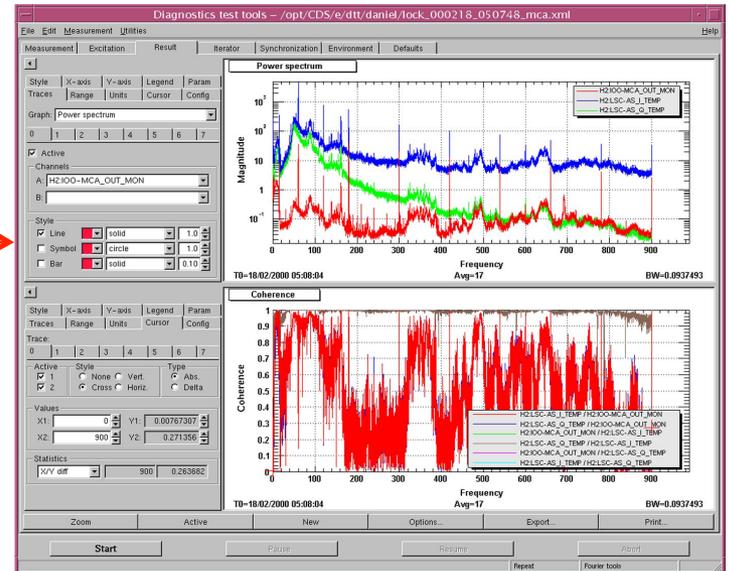
CDS Standard Filter Module With Diagnostics



AWG Input



DAQ System



Built-in Support for New Features

- RCG accepts user provided C code modules (Fast/Sync)
 - » Code must run within the real-time code time constraints
 - » Can be turned into RCG supported parts
- NDS / EPICS channel access client software to access control settings and diagnostics information. (Slow/Asynchronous)
 - » Support for C/C++, Python, Matlab
- Scheduled control (Slow to Fast/Synchronous)
 - » User task running on a real-time computer
 - » Could be expanded to run on a separate computer via a real-time data link

- Asynchronous communications between supervisory (EPICS) layer and real-time applications.
 - Not found to be a particular problem to date, but various synchronization schemes are being explored, as time permits.
- Exposes a lot of control settings, adding to overall complexity from the operations perspective.
 - Over 100,000 settings in aLIGO
 - Added SDF to monitor channels which should not change on a regular basis
- No built-in 'control intelligence' provided
 - Can we add some standard adaptive filtering algorithms?

Overall Assessment

- Not without growing pains, but hardware/software systems seem reliable and robust.
- Early deployment a big plus
 - » Users could become accustomed to the tools early
 - » Controls developed for early stand-alone SUS/SEI prototyping and test systems could be applied directly in the final aLIGO control system.
 - » CDS software developers were provided with plenty of feedback on problems, new feature requests at an early stage.
- Ready for O1