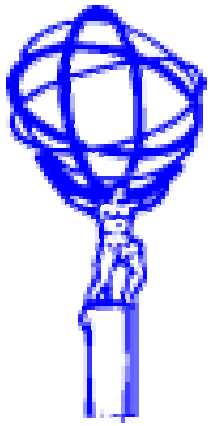


ATLAS TGC-DCS ELMB Software Status

Ronen Lifshitz

TGC-DCS Development Team

Physics Department

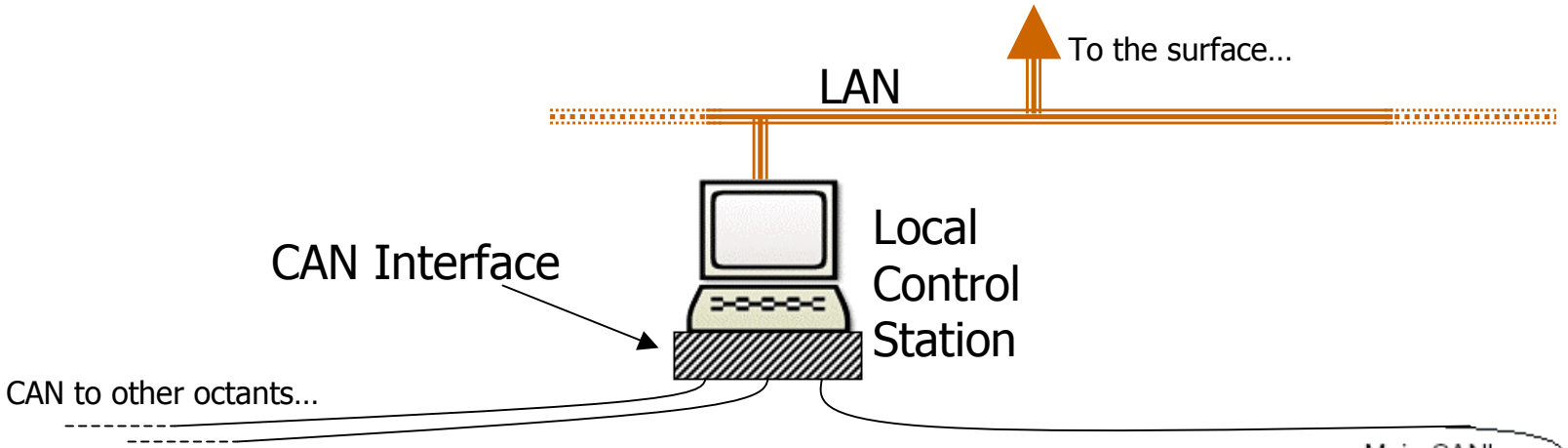
Technion, ISRAEL

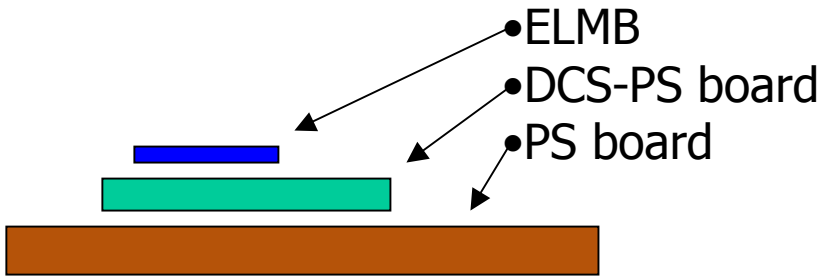
DCS Workshop - NIKHEF Amsterdam - Oct. 2001

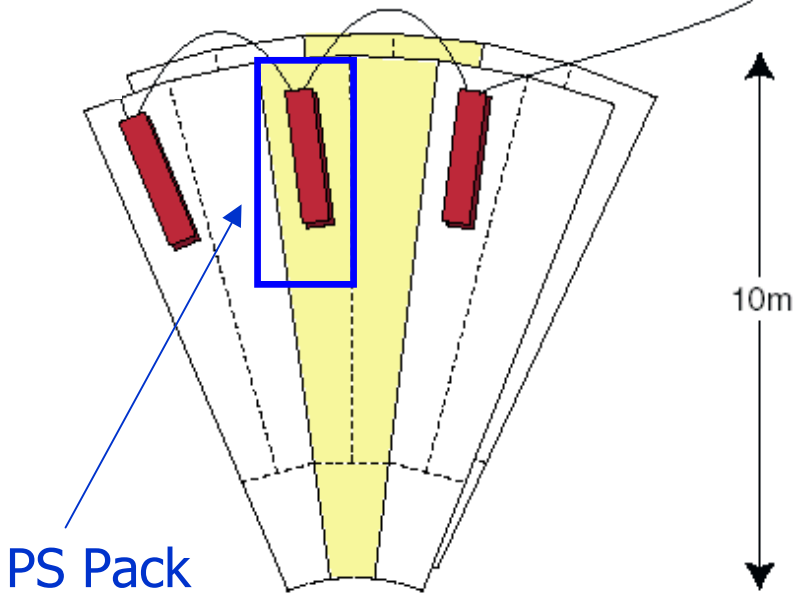
TGC-DCS: ELMB SW Status Report

- Introduction to the TGC-DCS Layout
- **ELMB SW Integration**
- Interaction with the **DCS-PS board**
- Interaction with the **PS board**
- Interaction with the **Local Control Station**
- Summary and Plans

Introduction to TGC DCS layout







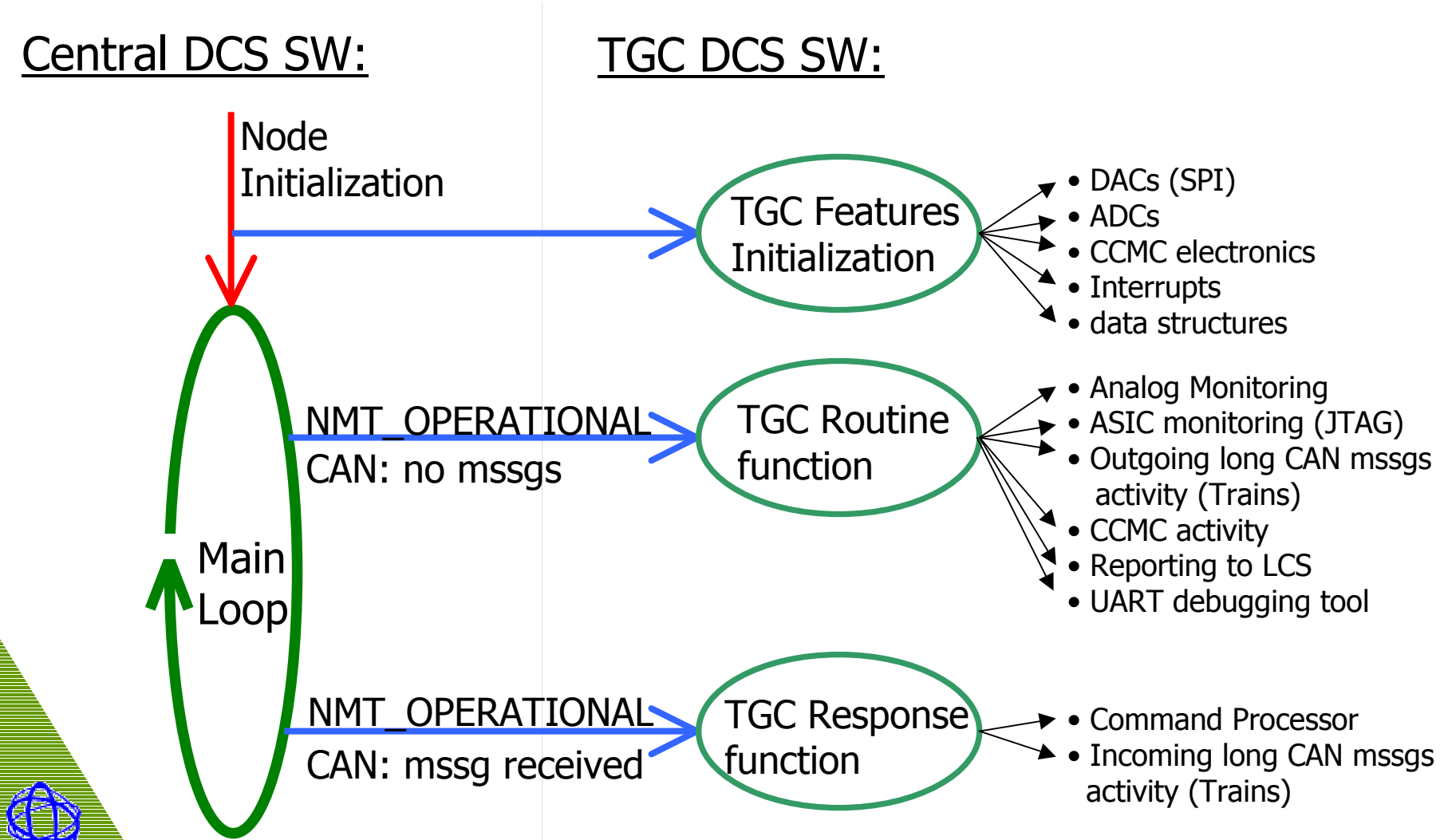
Thin Gap Chambers

LL08V01

ELMB SW Integration Concept

- **Minimal interface between Central DCS code and dedicated TGC features.**
- Current contact points:
  - CAN object declarations (2 PDOs)
  - TGC **Initialization** function
  - TGC **Routine** function
  - TGC **CAN Response** function
  - Toggling Real-Time clock tick flag
- All changes/additions to Central DCS code are logged, for simplifying future SW upgrading processes.

ELMB SW Structure



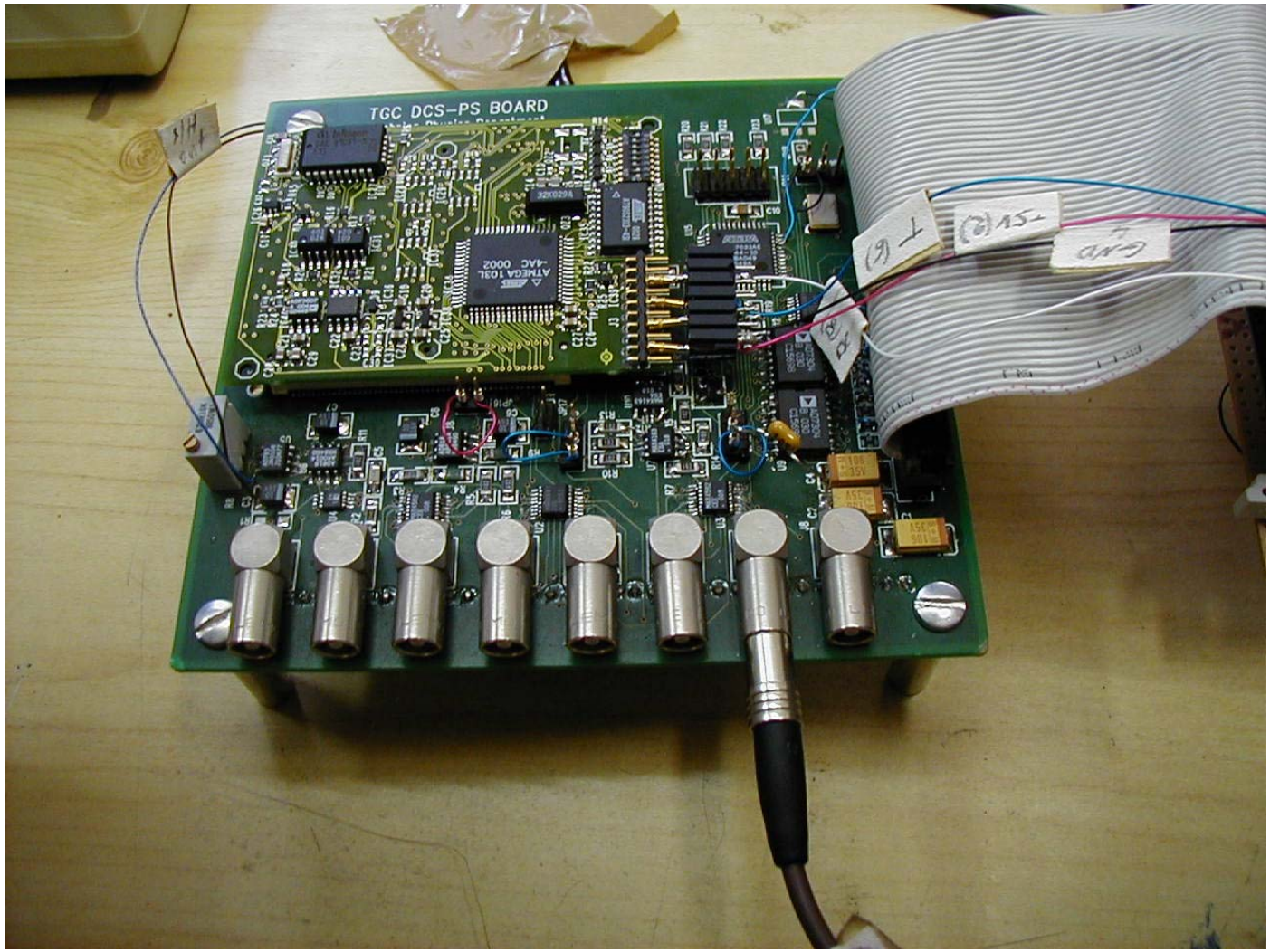
Interaction with DCS-PS board

- **Analog channels (Temp/Alignment/LV)** are multiplexed on DCS-PS board and monitored via ATmega103 ADC
- **ASD thresholds** are set using SPI controlled DACs
  - Built-in SPI features of ATmega103 were used, without conflicting with Central-DCS SPI activity
  - Threshold monitoring includes matching of read to set value, since DAC/ADC references are not equal
- Tests in lab are possible thanks to a temporary interface board, replacing the PS board (E.Hadash)

Interaction with DCS-PS board (2)

- Interaction with **Camber Charge Measuring Circuit** (N.Lupu) is fully operational:
  - Interrupt driven handshake protocol was implemented
  - Interrupt nesting is enabled for ADC operation
  - Automatic scan of input channels was implemented
  - Data is histogrammed and transmitted to LCS via CAN
  - Extensive monitoring features were already proven useful in debug stage

ELMB on DCS-PS board



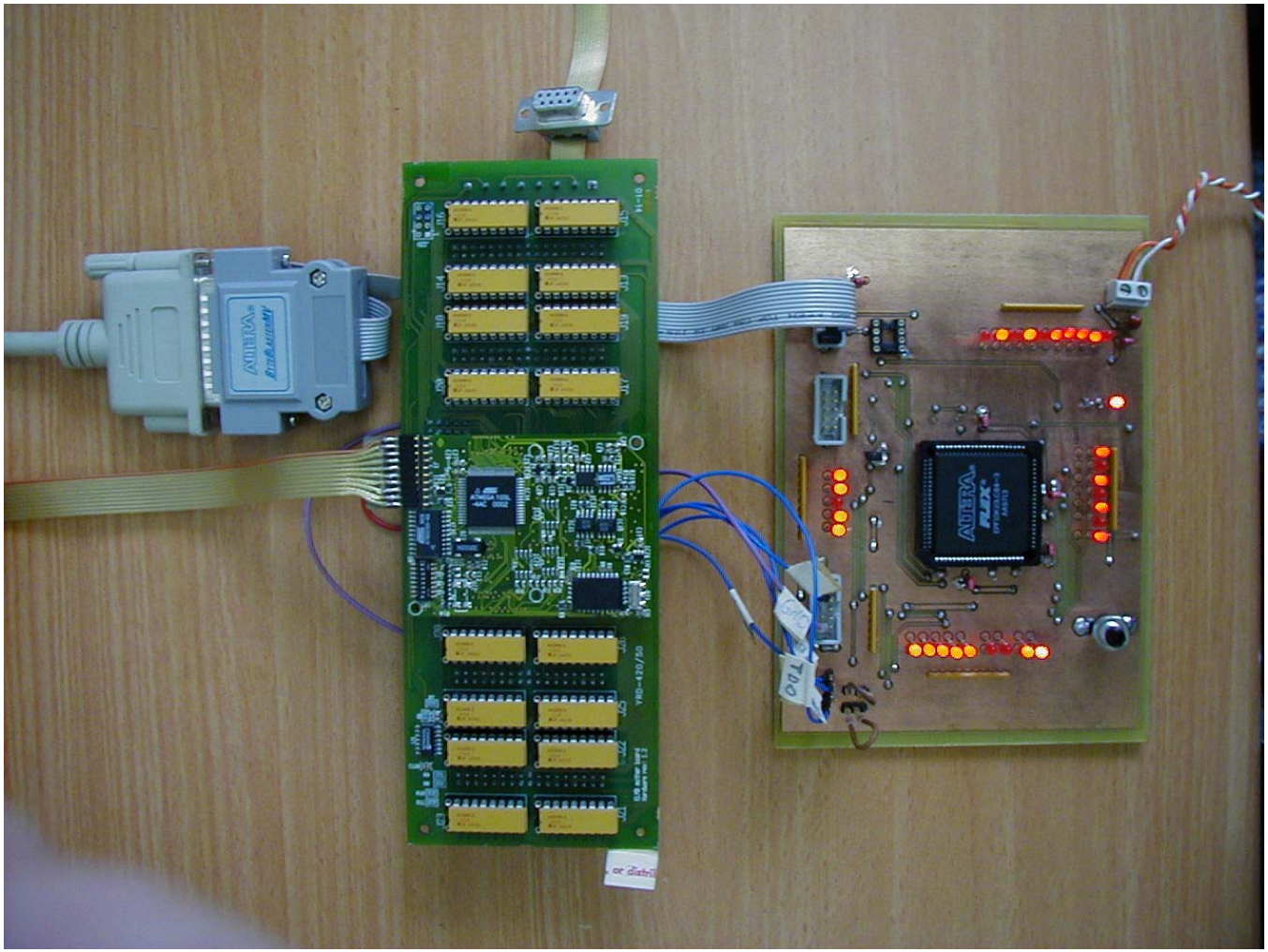
Interaction with PS board

- Main interaction with PS board is **ASIC configuration and monitoring (JTAG)**
- **JTAG toolbox is operational**
  - First approach was a generalized high-level JTAG control utility (exists & tested)
  - Although proven useful, a simpler and faster lower-level approach was taken.
  - Current JTAG data-transfer rates are approx 400KHz

Interaction with PS board (2)

- **ASIC configuration utilities were tested**
  - Without ASICs to configure,
    a **PPIC JTAG Control Emulator Board** was built in **Weizmann Institute**, hosting an FPGA (ALTERA FLEX10K), containing executable compiled from PPIC Verilog files.
  - JTAG register contents are reflected via LEDs
  - "Dialectical" JTAG was implemented
  - Existing tools can set/monitor a single register or all of them, in a specific PPIC or a whole PPIC chain

PPIC JTAG Control Emulator Board



Interaction with Local Control Station

- ELMB CANopen capabilities are used,
  as supplied by **Central DCS**.
- Local Control Station (LCS) runs the ATLAS-wide standard SCADA system – PVSS II
- LCS-ELMB CAN communication was established and tested.
- As an addition to the supplied OPC server,
  a dedicated PVSS-CAN driver was developed (A.Harel)
  for efficient & flexible communication
- LCS-ELMB CAN command syntax was established,
  use is expanding according to DCS features and use-cases.

Interaction with Local Control Station (2)

- **Dedicated long-message-protocol** (via CAN) was developed and implemented using CANopen PDOs (i.e. **Message Trains**)
- Message Trains are used instead of CANopen SDOs, yet compatible with CANopen (PDOs)
- **Trains use advantages**:
  - Customizable acknowledgement cycle allows high efficiency
  - Interface between TGC-dedicated and Central-DCS SW is kept small and easy to maintain/upgrade
  - SDO server may be used in future, without extra effort

Interaction with Local Control Station (3)

- **Message Trains protocol principle**:
  - Every CAN message starts with a command identifier
  - Train sender issues a command, specifying 8bit train ID
  - Train ID range is predefined, all IDs are illegal as command identifiers
  - CAN frames, starting with an active train ID, are assigned (7 bytes of data) according to the train initiator command contents
  - Acknowledgement messages are sent after reception N-messages and/or after last train-item is received.
  - Error-handling, Train abortions and re-transmissions treatment exists, yet untested

Summary

- **Success in implementing JTAG, SPI, CCMC-handshake**
- **Success in integrating TGC dedicated features into Central DCS SW**
- **Existing monitoring features have already proven useful (ASD thresholds, CCMC results)**
- **We DO have a DCS running…**

To do:

- **Establish LCS→ELMB SW download (via CAN)**
- **Master-Slave interactions not yet checked/used**
- **Test ASIC control with a real PS board (after ATLAS week)**
- **Use external memory for configuration data (now waiting for chip…)**
- **Implement I^2C**
- **Implement TGC Test Pulses**