CAN Bus Interface for the $\overline{\text{P}}$ANDA DCS

Florian Feldbauer

Johannes Gutenberg-Universität Mainz / Helmholtz Institut Mainz
Ruhr-Universität Bochum

XLIII. $\overline{\text{P}}$ANDA Collaboration Meeting
December 10, 2012

PANDA DCS and EPICS

Experimental Physics and Industrial Control System (EPICS) is used for the PANDA DCS

- Set of open source software tools, libraries and applications
- Network-based client/server model
- Decentralized architecture
- Freely scalable
- Supports many platforms (Linux, Unix, Windows, RTEMS, vxWorks, ...)
- Supports many architectures (x86, x64, ARM, ...)

Overview of the Slow Control for EMC

- Monitoring temperature, humidity, and pressure via CAN
  Temperature and Humidity Monitoring Board for $\overline{P}$ANDA (THM$\overline{P}$)
  Custom hardware developed at Bochum
- Controlling light pulser for monitoring radiation damages and
  transmittance of the PWO crystals via CAN
  Custom hardware developed at Bochum
- Controlling of VME crate via CAN
- Controlling power supplies via CAN
- Controlling chillers via RS232
- $\Rightarrow$ Other detectors will have similar requirements

For details on EMC Slow Control / EPICS see talks at DCS-TalksArchive on PANDA
wiki: `http://panda-wiki.gsi.de/cgi-bin/view/DCS/TalksArchive`

Requirements for CAN Bus Interface

- For the EMC we will have (per read out cycle):
    - 65 CAN frames per THM$\overline{P}$
      For EMC forward endcap ~10 THM$\overline{P}$s are forseen
    - 12 CAN frames per VME crate remote control
    - 8 CAN frames per HV channel (+6 per module)
      EMC forward endcap will have ~1000 HV channels (63 modules)
- Want to read out every 10 seconds
  $\Rightarrow$~900 CAN frames/s (EMC forward endcap)
  $\Rightarrow$ high data throughput needed
- Availability of hardware
- Easy maintainability of software
- Reliability
- Costs should be as low as possible
- Little space required
- Shielding
- Galvanic insulation?

HadCon CAN Bus Interface

- HadCon aka HadShoPoMo (Hades Shower Power Monitor) from HADES group (M. Traxler)

- HadCon is a general purpose IO module for SlowControl and small DAQ-systems
- ETRAX 100LX MCM 4+16 embedded CPU running EPICS
- CPU with 100 MHz and 16 MB RAM
- Microcontroller AT90CAN128 with integrated CAN interface connected via serial interface to CPU

HadCon CAN Bus Interface

Requirements fulfilled by HadCon

- Low cost
- Small
- RS232 support already exists for EPICS
  $\Rightarrow$ Easy maintainability of software

Requirements not fulfilled by HadCon

- Data throughput of CAN bus interface not sufficient for us
  (*Not developed for such purpose!*)
- CPU is discontinued

Other CAN Bus Interfaces

Alternatives to HadCon:

- CAN bus interfaces with high data throughput available from Kvaser and Peak Systems
- Many different types of interfaces supported (e.g. USB, PCIe, ...)
- But: very expensive ($\gtrsim 200 \,€$)
- Need PC for read out (also expensive, needs lots of space)
- Driver support from company?
- No support for EPICS systems available
- No easy maintainability of software!

Are there other solutions?

Other CAN Bus Interfaces

Alternatives to HadCon:

- CAN bus interfaces with high data throughput available from Kvaser and Peak Systems
- Many different types of interfaces supported (e.g. USB, PCIe, ...)
- But: very expensive ($\gtrsim 200 \, €$)
- Need PC for read out (also expensive, needs lots of space)
- Driver support from company?
- No support for EPICS systems available
- No easy maintainability of software!

Are there other solutions?

- All CAN interfaces from Kvaser and Peak Systems use SJA1000 stand-alone CAN Controller
- SJA1000 has parallel interface with 8 mulitplexed address/data lines, 5 control lines (Motorola and Intel protocol supported)
- Idea: Connect SJA1000 directly to an embedded CPU
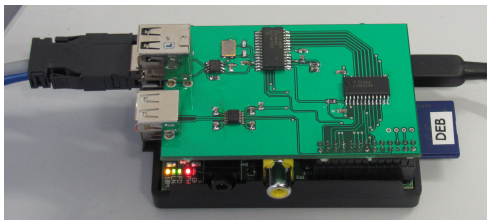
Raspberry Pi Computer

- Small, fully functional computer developed in UK
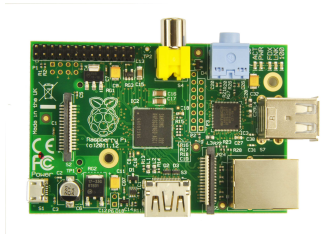
- Broadcom BCM2835 SoC with 700 MHz ARM CPU and 512 MB RAM (and full HD graphics)
- up to 17 GPIOs, UART, I^2C, SPI
- 10/100 MBit Ethernet-Controller
- Dimensions: $85.60\,\text{mm} \times 53.98\,\text{mm} \times 17\,\text{mm}$
- Power Consumption: $\leq 3.5\,\text{W}$
- Costs $\sim$30 €
- c.f. http://www.raspberrypi.org/ for further informations/details

CAN Bus Interface for Raspberry Pi Computer

- Developed extension board with CAN bus and RS232 interfaces

- Driver based on open-source linux driver from Peak Systems
- Raspberry Pi CAN interface has comparable data throughput/performance to Peak Systems' USB-CAN-Converter (~ 1000 CAN frames/s sending/receiving)
- EPICS device support implemented for all used devices with CAN interfaces (for prototype of EMC forward endcap)

Summary and Outlook

- $\overline{P}$ANDA DCS needs high-performance CAN bus interface
- Commercial solutions too expensive, need lots of space
- Developed a CAN bus interface for Raspberry Pi Computer
  Advantages:
    - Small, low cost
    - High data throughput
    - Each CAN bus interface has its own front-end computer
    - Generic design: Just need CPU board with GPIO header and ethernet
    - $\Rightarrow$ All requirements for CAN bus interface fulfilled!
- EPICS support implemented for this CAN bus interface and connected devices

- Further device support needed?
  Which devices with CAN bus interface are used by other groups?
- Add galvanic insulation for interfaces