Exploring Embedded Systems' Dedicated Cores for Real-Time Applications

LNLS Brazilian Synchrotron Light Laboratory · sirius

CNPEM Brazilian Center for Research in Energy and Materials

P. H. Nallin*, R. C. Ito, J. G. R. S. Franco, A. R. D. Rodrigues

Brazilian Synchrotron Light Laboratory – LNLS

**patricia.nallin@lnls.br*

Introduction

Sirius, the new 3 GeV Brazilian Light Source

- Technological challenges
- Controlling and monitoring a large variety of equipment
- Embedded solution for general and critical controls:

Beaglebone Black and its Programmable Real-Time Units (PRUs)

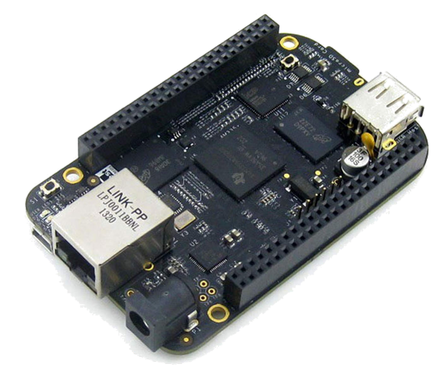

Figure 1: Beaglebone Black

Beaglebone Black

- In use since 2016 (UVX facility) – Debian distribution
- Inexpensive open hardware
- Chosen as distributed cores for Sirius Controls System
- Dual Programmable Real-Time Units as SoC subsystems
- High level tools can be installed on it

PRU: an embedded real-time core

- 32-bit RISC core running at 200 MHz
- 8 KB instruction memory
- Interface with userspace environment
- Access to Beaglebone hardware
- More than 8 KB for data storage

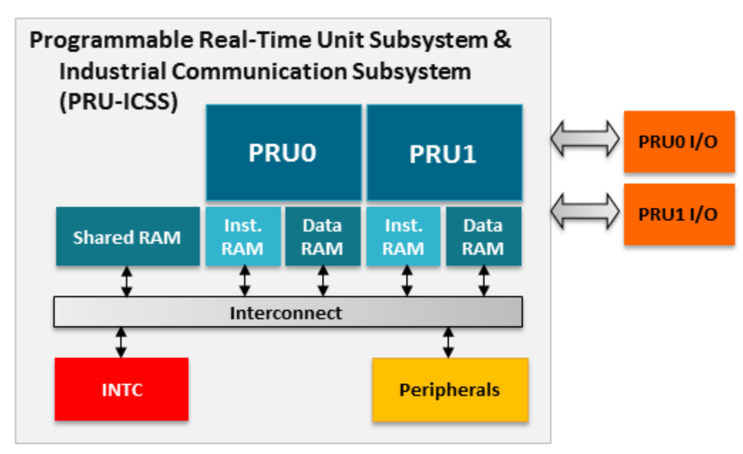

Figure 2: Programmable Real-Time Units subsystems, from Texas Instruments (http://processors.wiki.ti.com/index.php/PRU-ICSS)

Coding

Either Assembly or C/C++
(compilers provided by Texas Instruments)

Carrier Library

Built binary can be loaded with *libprussdrv* C library, from embedded Linux userspace

GPIO access

Registers directly accessed by PRUs

Rise time: 3.08 ns – Fall time: 2.64 ns

Applications in Sirius Light Source

Three different tasks so far, described below

High-Performance Serial Interface

Purpose: fast serial communication interface to critical device (PRUserial485).

Aspects:

- External UART, configurable up to 15 Mbps
- Single package large data transfers
- Python or C interface to PRUs
- Synchronous and conventional mode
- RS-485 standard

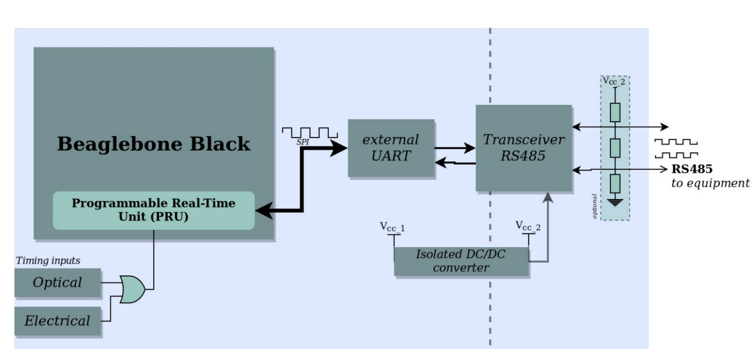

Figure 3: System architecture (SERIALxxCON) for PRUserial485

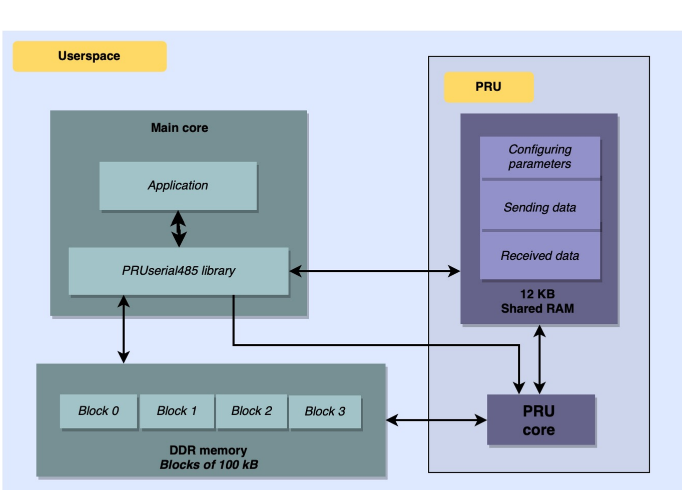

Figure 4: Memory mapping

Data transfers (userspace ↔ PRUs)

- Memory mapping:
- Shared RAM (faster) and external DDR for different modes

Sync operation

- Serial messages after a timing pulse
- **Modes**: general broadcast message
  single curve via setpoints
  multiple curves via setpoints
- Recovery time: 3.1µs

Sync Message	Message [bytes]	Latency [µs]	Jitter [ns]
Broadcast	6	1.01	13.44
Single setpoint	10	1.43	52.73
Four setpoints	22	2.19	82.43

Figure 5: Time characterization for PRU in synchronous mode, polling for trigger input

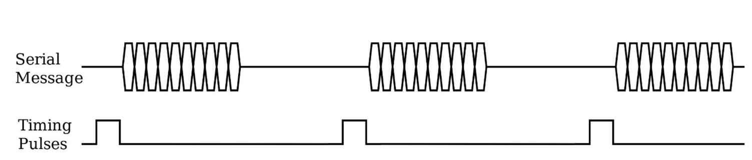

Figure 6: Synchronous operation example

Figure 7: Project GitHub github.com/lnls-sirius/pru-serial485

Status: Units currently in use for Power Supplies interface

Multi-Purpose Counting System

Main purpose: Diagnostics with Bergoz Beam Loss Monitor (Diff BLM) and in-house developed gamma detectors.

Board configuration:

- PoE powered
- 2 Bergoz driver and input channels
- 6 digital single-ended channels

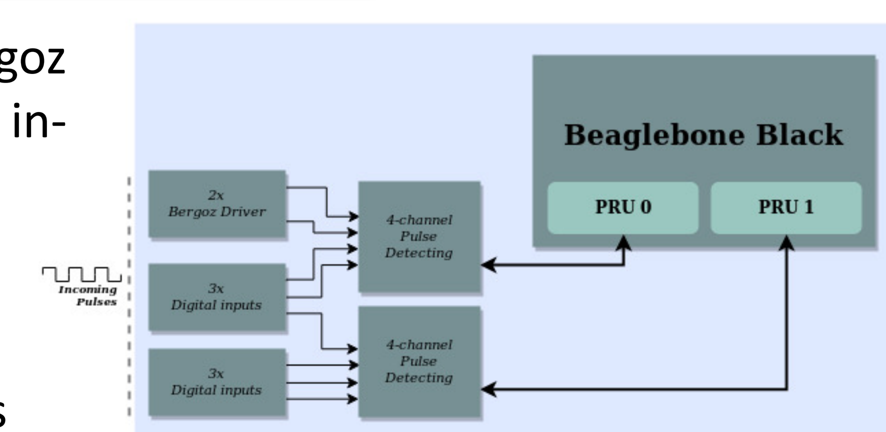

Figure 8: System architecture for Counting System

Maximum counting rates

Active Channels	Assembly [MHz]	C [MHz]
1	14.3	4.0
2	12.5	-
3	11.1	-
4	10.0	-

Figure 9: Counting rates

Figure 10: Hardware integration

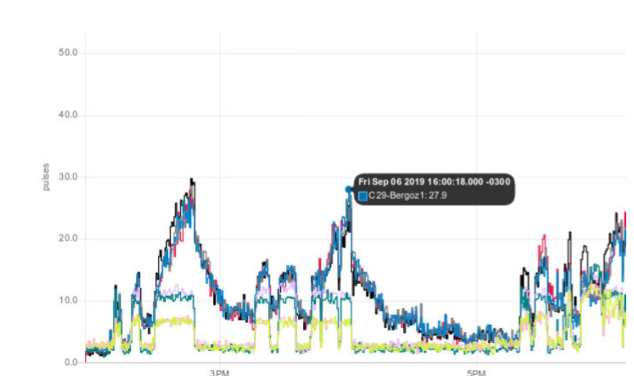

Figure 11: Application in Sirius site, interfacing Bergoz BLM positioned at first Booster sectors. Data acquired during machine commissioning tests

Status: Installation in progress in Sirius Storage Ring

Water Leak Monitor

Purpose: detect minimal water leaks in tunnel area with capacitive sensors and a coaxial line in Sirius girders.

Distance calculation: signal and reflected signal travel time. One-way max error: 0.61 m

Status: final developments

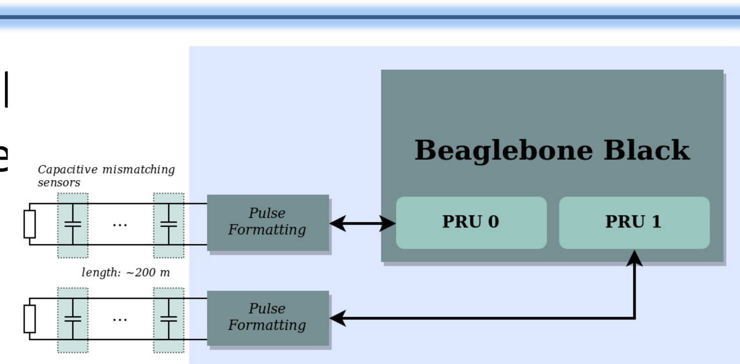

Figure 12: Interface for detecting water leak position inside Sirius tunnel.

EPICS and Redis Integration

Beaglebone Black:

Running real-time applications inside an embedded Linux

- Easier and cleaner system integration
- Python or C interface to PRUs
- Plenty of possibilities on sharing data

EPICS

- Sirius Controls System is based on EPICS
- EPICS base and modules can be installed on Beaglebone Black
- Possibility of running IOCs directly from single board computers

- Largely used worldwide and recently integrated to some systems at CNPEM/Sirius
- Redis Server can be installed on Beaglebone Black

redis

Conclusion

Having both embedded Linux and a dedicated core in the same SoC reduces costs and system overall complexity, allowing developers to design time-critical applications with the advantage of sharing data with largely used tools, such as EPICS and Redis. Demonstrated applications using this architecture are vital to ensure Sirius operation, once low jitter, low latency and determinism are important for real-time applications.