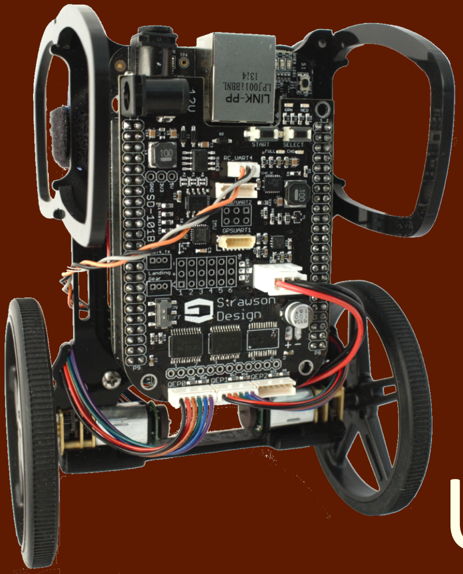
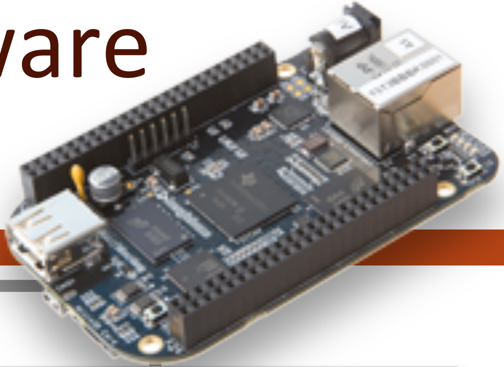


USING THE BEAGLEBONE REAL-TIME MICROCONTROLLERS

beagleboard.org

Jason Kridner, jkridner@beagleboard.org

BeagleBone Black: Open hardware computer for makers



Truly flexible open hardware and software development platform

All you need is in the box

Proven ecosystem from prototype to product

BeagleBone Black

- Ready to use: ~$50
- 1 GHz performance
- On-board HDMI to connect directly to TVs and monitors
- More and faster memory now with 512MB DDR3
- On-board flash storage frees up the microSD card slot
- Support for existing Cape plug-in boards

Most affordable and proven open hardware Linux platform available

beagleboard.org

What are PRUs

- “Programmable Real-time Units”
- 32-bit RISC processors at 200MHz with single-cycle pin access for hard real-time
- Optimized for packet processing/switching and software implementations of peripherals
- Part of the PRU-ICSS, “Industrial Communications SubSystem”

Why and when to use PRUs

- Free from running on an operating system, so can be dedicated to a function
- Real-time because it can't be interrupted from its given task by other tasks
  - Interrupts are simply registered into an event register
  - Operations scheduled in an event loop
- Low, low, low latency from input to output
  - Zero-depth pipeline
- You can't interface an external MCU to DDR memory so fast!

Examples usage

- Tight control loops
  - Driving motors in a mobile robot, CNC machine or 3D printer
- Custom protocols
  - WS28x LEDs, DMX512, ...
  - EtherCAT, ProfiBUS, ProfiNET, ...
- Soft peripherals
  - PWM, UART (LEGO), ...

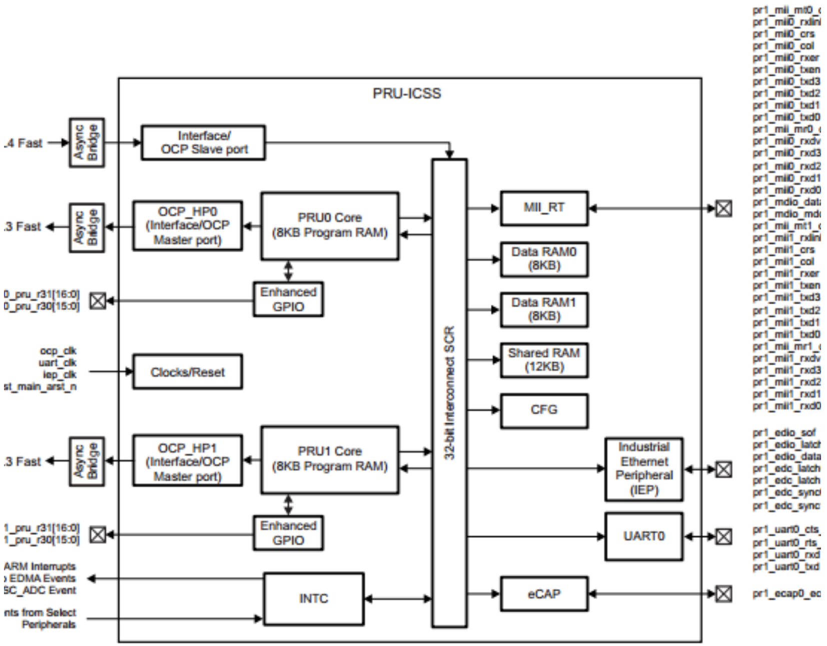
Example projects (see wiki page)

- 6502 memory slave
- DMX512
- WS28xx LEDs (OLA, LEDscape)
- MachineKit for 3D printing or CNC
- GSoC: pruspeak, BeagleLogic
- GCC, Forth, ...

PRUSS architecture details

- 2 cores at 200MHz each
- Memory
  - 8kB program each
  - 8kB data each
  - 12kB data shared
  - Access to external memory and peripherals
- Parallel/serial capture/ send

Figure 2. PRU-ICSS Integration



25 PRU low-latency I/Os

P9

DGND	1	2	DGND
VDD_3V3	3	4	VDD_3V3
VDD_5V	5	6	VDD_5V
SYS_5V	7	8	SYS_5V
PWR_BUT	9	10	SYS_RESETN
GPIO_30	11	12	GPIO_60
GPIO_31	13	14	GPIO_50
GPIO_48	15	16	GPIO_51
GPIO_5	17	18	GPIO_4
I2C2_SCL	19	20	I2C2_SDA
GPIO_3	21	22	GPIO_2
GPIO_49	23	24	GPIO_15
PRU0_7	25	26	PRU1_16 IN
PRU0_5	27	28	PRU0_3
PRU0_1	29	30	PRU0_2
PRU0_0	31	32	VDD_ADC
AIN4	33	34	GNDA_ADC
AIN6	35	36	AIN5
AIN2	37	38	AIN3
AIN0	39	40	AIN1
PRU0_6	41	42	PRU0_4
DGND	43	44	DGND
DGND	45	46	DGND

P8

DGND	1	2	DGND
GPIO_38	3	4	GPIO_39
GPIO_34	5	6	GPIO_35
GPIO_66	7	8	GPIO_67
GPIO_69	9	10	GPIO_68
PRU0_15 OUT	11	12	PRU0_14 OUT
GPIO_23	13	14	GPIO_26
GPIO_47	15	16	GPIO_46
GPIO_27	17	18	GPIO_65
GPIO_22	19	20	PRU1_13
PRU1_12	21	22	GPIO_37
GPIO_36	23	24	GPIO_33
GPIO_32	25	26	GPIO_61
PRU1_8	27	28	PRU1_10
PRU1_9	29	30	PRU1_11
GPIO_10	31	32	GPIO_11
GPIO_9	33	34	GPIO_81
GPIO_8	35	36	GPIO_80
GPIO_78	37	38	GPIO_79
PRU1_6	39	40	PRU1_7
PRU1_4	41	42	PRU1_5
PRU1_2	43	44	PRU1_3
PRU1_0	45	46	PRU1_1

Accessing the other peripherals

- Yes, you can!
- The “L3” bus is exposed, so you can directly poke all of the peripheral registers
- Be careful! --- be sure the main CPU isn’t trying to access them at the same time, so you need to manually disable access to them on the main CPU

PRU tools – a work in progress

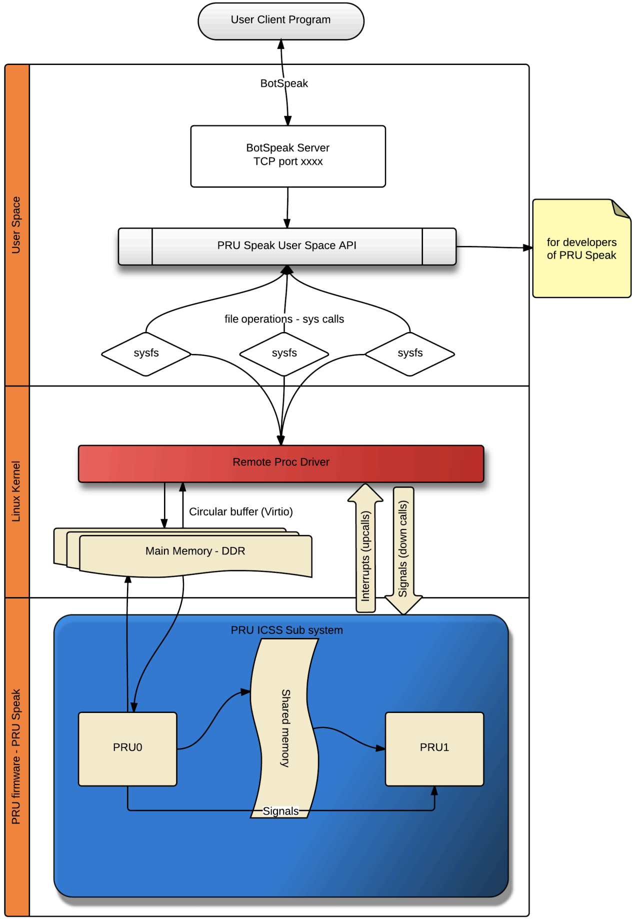
- TI C compiler
- GCC
- Forth
- PRU Speak
- TI StarterWare library

Typical components to talk to PRUs

- Linux kernel driver, either uio_pruss or pru_rproc
- Userspace loader or kernel driver for application
- Device tree entries to configure pins and driver
- Your actual PRU firmware source written in C or assembly

PRU Speak

- □ Complete firmware ready-to-run
- □ Included in upcoming software release
- □ Implementation of "BotSpeak" from Tufts



What is BotSpeak?

□ Runs on Arduino, LEGO, Rasberry Pi, BeagleBone Black (JavaScript), ...

□ Integrates with Labview

□ Interpreted with variables, conditions and scripts

□ Assembly language-like

□ Interfaced via serial, network or web

□ Arduino-like operations

BotSpeak

Universal Robotics Programming Language

http://botspeak.org/

Questions!

- http://elinux.org/Ti_AM33XX_PRUSSv2
- jkridner@beagleboard.org (but, I don’t answer questions if beagleboard@googlegroups.com isn’t in copy)
- Follow @jadon and @beagleboardorg