Application Note

PRU Ultrafast Broadside (RTOS) - AM243x LP EVM / AM64x GP EVM

Rev. <ref>
<date>

Revision History

Version	Date	Author	Description
0.3	Oct 24 - 2022		Added broadside accelerator example

Table of Contents

1 Ultrafast_Broadside

Learning goal:

- Example 1 uses BSWAP widget on internal register only
- 2nd Example which data transfer widget to send 64 bytes to memory
- Examples uses include file for broadside definitions and memory definitions

Broadside extension of PRU core can operate data transfers and data processing functions using 1024 bit data bus. In this chapter there are two examples to demo the advantage of ultrafast broadside accelerator. An overview of all broadside accelerator is shown in appendix 2.3 It shows the register allocation of the accelerator. There is a collision with C compiler which uses R2, R14, R15 for stack pointer and function call arguments. Therefore these examples are using assembler coding. For C code implementation using broadside accelerator a register context save and restore is required.

1.1.1 Byte Swap accelerator

The byte swap accelerator has three different modes which are described in TRM chapter "6.4.6.2.7 PRU_ICSSG Byte Swap".

- ID 160: bytes swap inside register which can be applied to R0-R29, in steps of one byte
- ID 161: 4_8 function swaps the contents of R2-R5 <-> R6-R9
- ID 162: 4_16 function swaps the contents of R2-R5 <-> R14-R17 and R6-R9 <-> R10-R13

All modes take a single cycle to execute. Figure 1 shows assembly source for all three modes. With single step (F5) through the source code and one can observe the operation using register view when PRU core is selected.

1.1.2 Xfer2vbus DMA widget

The burst commands for memory transfer are variable in execution time and depend on the length of the operation for write transfers. For example a 64 byte burst to any memory location takes 17 PRU cycles. For write transfers it does not matter whether target is inside ICSS or any system memory as the interconnect has bridges with buffers to decouple the transfer from bus arbitration and access time. With xfr2vbus write widget the PRU does not see additional cycles for burst operation. The xout instruction contains both, target address and number of bytes to execute. The destination address is stored in r18 (32 bit address) and r19 (48 bit address, bit 33-48).

```
;********
;* MAIN *
;********

main:
    zero    &r0, 120

; fill r0-r29 with incremental bytes from 0 to 119
; loop instruction saves two instructions per iteration
; pointer increment and eof loop check
; each iteration takes two cycles instead of 3 cycles.
    ldi     r1.b0, 120
    loop    endloop, r1.b0
    mvib    *--r1.b0, r1.b0
endloop:

; restore value in r1.b0 which is the pointer
    ldi     r1.b0, 4

; perform bytes swaps with different length
    xin     160, &r0, 4
    xin     160, &r1, 120-4
    xin     160, &r0.b3, 2

; perform 4_8 register swap
    xin     161, &r2, 8*4
    xin     161, &r2, 8*4

; perform 4_16 register swap
    xin     162, &r2, 16*4
    xin     162, &r2, 16*4

; init xfr2vbus write dma
    ldi32 xfr2vbus_addr, MSRAM_BK0 ; write to MSRAM
    ldi   xfr2vbus_addr_high, 0    ; 48 bit address extension

; execute xfr2vbus_write - 64 byte non blocking write operation
    xout  0x62, &r2, 68

; same memory transfer with sbco instruction takes 17 cycles
    sbco  &r2, c16, 0x40, 64
```

Figure 1 - PRU broadside accelerator example

2 Appendix

2.1 References

PRU Optimizing C/C++ Compiler User's Guide

PRU Assembly Language Tools User's Guide

PRU assembly instructions

AM64x/AM243x Technical Reference Manual

AM243x Datasheet

2.2 PRU IO poster

PRU R30/31 Input/Output modes

- Each PRU processor implements fast GPO through R30 register and GPI through R31 register
  - PRU has full input and output control on all interfaces
  - RTU_PRU and TX_PRU see R31 input and can process receive in parallel

General-Purpose Input modes (R31)

Direct Input	**GPI [19:0]** feeds directly into the PRU R31 in Default state There are **80 general-purpose inputs** in total.
16-bit parallel capture	**DATAIN[0:15]** is captured by the pos_edge or neg_edge of **CLOCKIN R31[16]** programmable through the PRU_ICSSG CFG register → If clocking is configured to be **positive/negative**, then it will **equal PRUn_Clock/PRUn_Clock inverted**
28-bit Serial shift in mode	**DATAIN is sampled** and **shifted into a 28-bit shift register** on an internal clock pulse. The **shift rate** is controlled by the **effective divisor of two cascaded dividers applied to the ICSSG_CORE_CLK clock** (150MHz/ 200MHz/ 225MHz) → It also supports **Start Bit Detection (SB), Shift Counter (Cnt_16)** and **Stop/Freeze** current shift operation and disable search for new Start Bit
MII_RT	mii_rt_r31_status [29:0] internally driven by the MII_RT module Enabled by ICSSG_GPCFG0_REG[1-0] PRU0_GPI_MODE register (value: 3h), where n = 0 or 1
9x Sigma Delta	Integrator counts the number of 1's per clock event. **Each channel has three cascaded counters**, which are the accumulators for the **Sinc3 filter.** Each **counter and accumulator are 28 bits.** This sample counter **updates the count value** on the effective clock event for that channel. Each channel also **contains a programmable counter compare block.**

General-Purpose Output modes (R30)

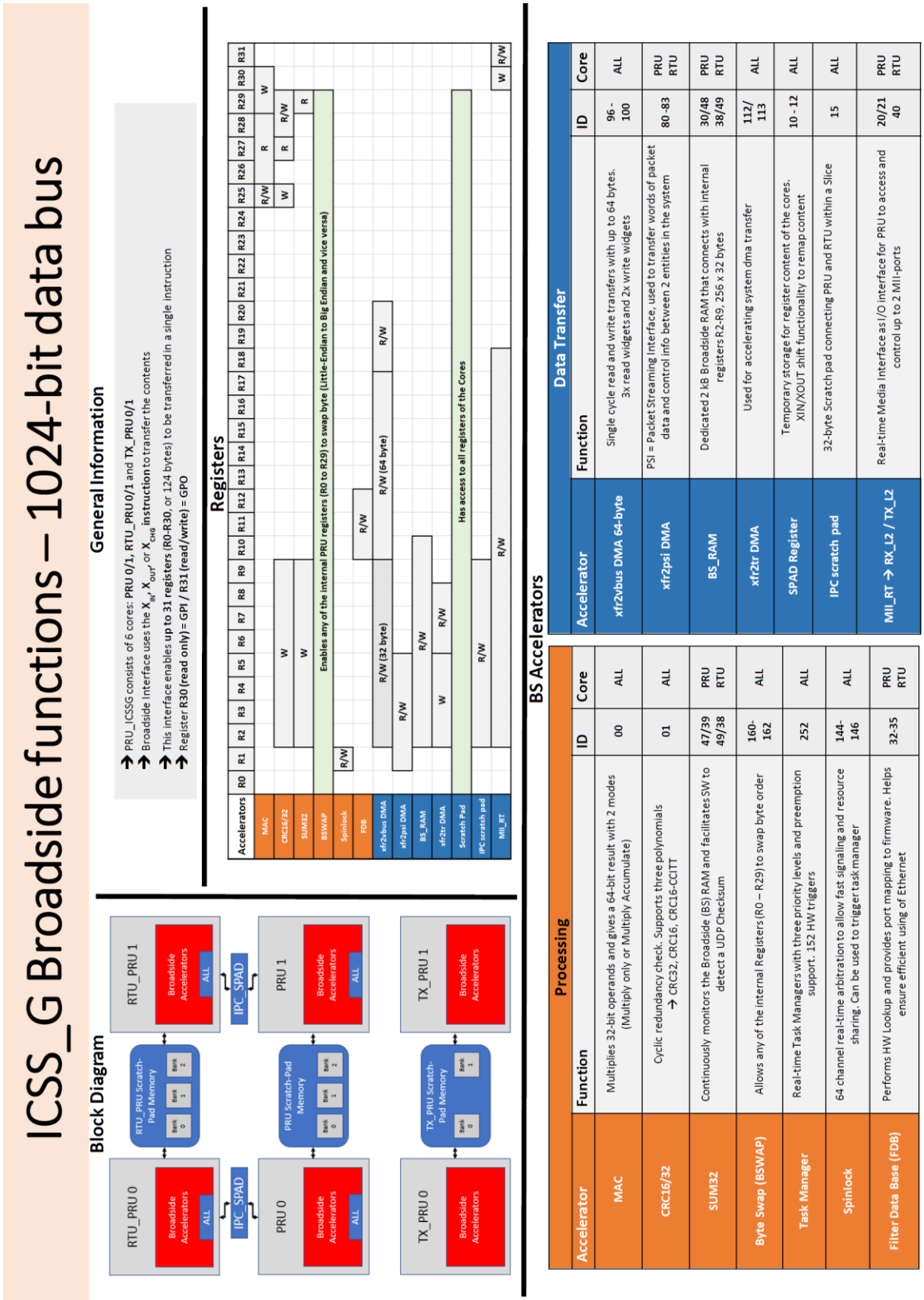
Direct Output	**PRU0_r30[0:19]** feeds directly into PRU0_GPO[0:19] There are **40 total general-purpose outputs**
Serial shift out mode	Data is **shifted out of PRU0_r30[0]** (PRU0_DATAOUT) on **every rising edge** of PRU0_r30[1] (PRU0_CLOCK) The **shift rate** is controlled by the **effective divisor of two cascaded dividers applied to the ICSSG_CORE_CLK clock** (150MHz/ 200MHz/ 225MHz) → Supports **2 modes**: Free Running Clock Mode (default) and Fixed Clock Count Mode

GPIO (R30/R31)

INTC (R31)	Interrupt controller (INTC) **maps interrupts** coming from different parts of the device to a reduced set of PRU_ICSSG interrupt channels, uses **bit 30, 31** Capturing **up to 160 Events,** supports **20 output interrupt channels,** each **event/host can be enabled/disabled**
3x Peripheral interface	**3 channels** with baud range from 100 kHz to 16 MHz, **The Peripheral Interface's I/O**s are **multiplexed** with the PRU GPI/GPO signals **TX FIFO** size of 32 bits, **RX FIFO** size of 4 bits

I/O	GPI/O0	GPI/O1	GPI/O2	GPI/O3	GPI/O4	GPI/O5	GPI/O6	GPI/O7	GPI/O8	GPI/O9	GPI/O10	GPI/O11	GPI/O12	GPI/O13	GPI/O14	GPI/O15	GPI/O16	GPI/O17	GPI/O18	GPI/O19
Direct Input	GPI0	GPI1	GPI2	GPI3	GPI4	GPI5	GPI6	GPI7	GPI8	GPI9	GPI10	GPI11	GPI12	GPI13	GPI14	GPI15	GPI16	GPI17	GPI18	GPI19
Parallel Capture	DATAIN0	DATAIN1	DATAIN2	DATAIN3	DATAIN4	DATAIN5	DATAIN6	DATAIN7	DATAIN8	DATAIN9	DATAIN10	DATAIN11	DATAIN12	DATAIN13	DATAIN14	DATAIN15	Clocking			
28-bit Shift In	DATAIN																			
9x Sigma Delta	SD0_CLK	SD0_D	SD1_CLK	SD1_D	SD2_CLK	SD2_D	SD3_CLK	SD3_D	SD4_CLK	SD4_D	SD5_CLK	SD5_D	SD6_CLK	SD6_D	SD7_CLK	SD7_D	SD8_CLK	SD8_D		
Direct Output	GPO0	GPO1	GPO2	GPO3	GPO4	GPO5	GPO6	GPO7	GPO8	GPO9	GPO10	GPO11	GPO12	GPO13	GPO14	GPO15	GPO16	GPO17	GPO18	GPO19
Serial shift out mode	DATAOUT	CLOCKOUT																		
3x Peripheral interface	PERIF0_CLK	PERIF0_Out	PERIF0_OUT_EN	PERIF1_Out	PERIF1_CLK	PERIF1_OUT_EN	PERIF2_CLK	PERIF2_Out	PERIF2_OUT_EN	PERIF0_IN	PERIF1_IN	PERIF2_IN								

2.3 PRU broadside poster



IMPORTANT NOTICE AND DISCLAIMER