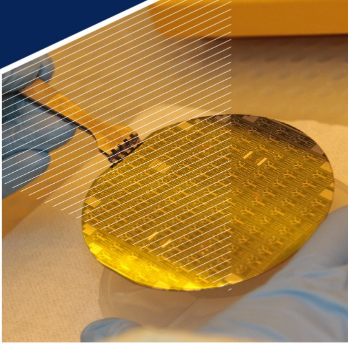


Study of the data exchange between PL and PS of Zynq-7000 devices

Rodrigo A. Melo, Bruno Valinoti (INTI)
Marie Baly Amador, Luis G. García (ICTP)
Andres Cicuttin, Maria Liz Crespo (ICTP)

Motivation

FPGA SoC:

- In 2010 Actel (later Microsemi, now Microchip) introduced SmartFusion (ARM Cortex-M3).
- In 2011 Xilinx introduced Zynq-7000 and Altera (now Intel Programmable Solutions Group) some variants of Cyclone/Arria (2 x ARM Cortex-A9).

Previous attempts:

- Excalibur from Altera (ARM 9 and MIPS microcontrollers)
- Virtex-II and Virtex-4 Pro from Xilinx (embedded PowerPC from IBM)

The uP approach has a lowest integration level and lack of peripherals. The FPGA SoC solution integrates the software programmability of state of the art processors, capable of run an operating system, with a huge variety of general purpose and high speed peripherals, and several memory controllers, with the flexibility and scalability of programmable hardware into a single device.

Advanced Microcontroller Bus Architecture

An open standard for the connection and management of functional blocks in a SoC.

ARM

- **AMBA 1 (1996):** Advanced Peripheral Bus (**APB**)
- **AMBA 2 (1999):** AMBA High-performance Bus (**AHB**)
- **AMBA 3 (2003):** Advanced Extensible Interface (**AXI3**)
- **AMBA 4 (2010): AXI4**

Xilinx was one of the thirty-five companies that contributed with the AMBA 4 specification and an early adopter.

Source: ARM AMBA 4 Specification maximizes performance and power efficiency (press release)

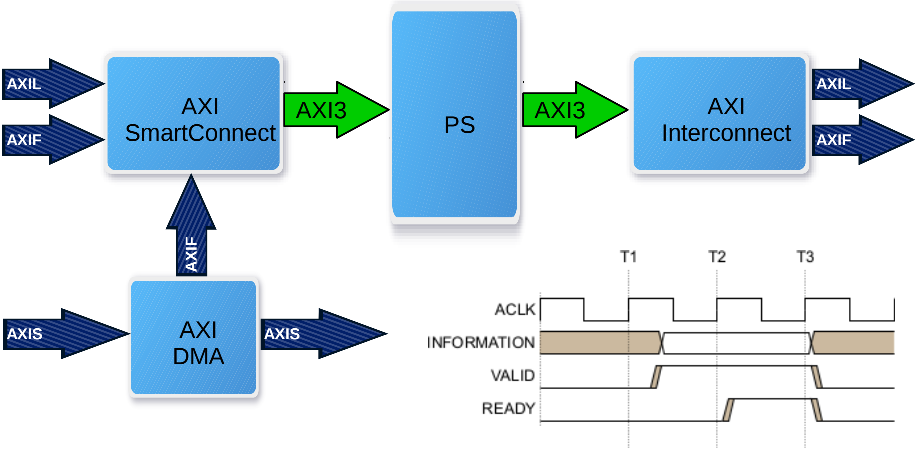
AXI 3 vs 4

Masters and slaves in the PS are AXI 3, but hardware in the PL is suggested to be AXI 4.

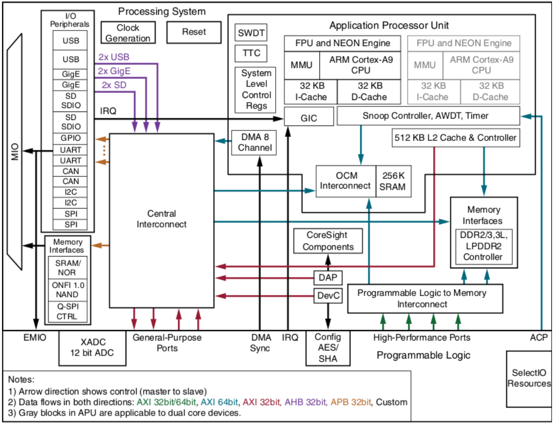
The maximum burst length was extended from 16 to 256 beats (INCR type). Additionally, AXI 4 defines three interfaces:

- **AXI4** (also known as **AXI4-Full**) for high-performance memory-mapped requirements.
- **AXI4-Lite** for simple, low-throughput memory-mapped communication (such as control and status registers).
- **AXI4-Stream** for high-speed streaming data (removes address phase and allows unlimited data burst size).

Vivado AXI Infrastructure



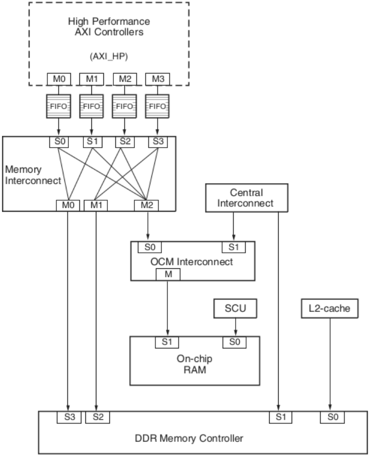
Zynq-7000 All Programmable SoC Overview

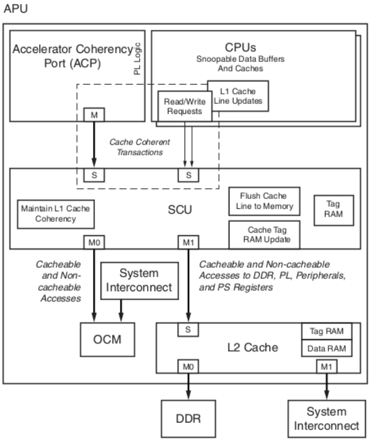


Source: Zynq-7000 All Programmable SoC Technical Reference Manual (UG585)

- Cortex-A9 MPCore (r3p0)
- 2 x 32b General Purpose masters (M_AXI_GP[1:0])
- 2 x 32b General Purpose slaves (S_AXI_GP[1:0])
- 4 x 32/64b High Performance slaves (S_AXI_HP[3:0])
- 1 x 64b Accelerator Coherency Port slave (S_AXI_ACP)

More about AXI ACP and HP





Source: Zynq-7000 All Programmable SoC Technical Reference Manual (UG585)

Data Movement Method Comparison Summary

Method	Benefits	Drawbacks	Suggested Uses	Estimated Throughput
CPU Programmed I/O	• Simple Software • Least PL Resources • Simple PL Slaves	• Lowest Throughput	• Control Functions	<25 MB/s
PS DMAC	• Least PL Resources • Medium Throughput • Multiple Channels • Simple PL Slaves	• Somewhat complex DMA programming	• Limited PL Resource DMAs	600 MB/s
PL AXI_HP DMA	• Highest Throughput • Multiple Interfaces • Command/Data FIFOs	• OCM/DDR access only • More complex PL Master design	• High Performance DMA for large datasets	1,200 MB/s (per interface)
PL AXI_ACP DMA	• Highest Throughput • Lowest Latency • Optional Cache Coherency	• Large burst might cause cache thrashing • Shares CPU Interconnect bandwidth • More complex PL Master design	• High Performance DMA for smaller, coherent datasets • Medium granularity CPU offload	1,200 MB/s
PL AXI_GP DMA	• Medium Throughput	• More complex PL Master design	• PL to PS Control Functions • PS I/O Peripheral Access	600 MB/s

*

$$MB/s = MHz * \frac{bits}{8}$$

* PL Freq. is 150 MHz
* Data width is 32/64 bits

Where is the protocol overhead?

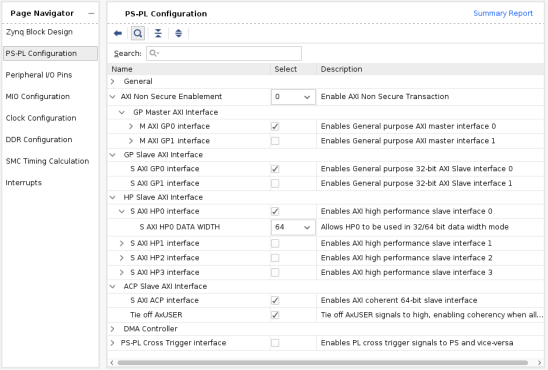
Source: Zynq-7000 All Programmable SoC Technical Reference Manual (UG585)

System-Level Address Map

Address Range	CPUs and ACP	AXI_HP	Other Bus Masters[1]	Notes
0000_0000 to 0003_FFFF [2]	OCM	OCM	OCM	Address not filtered by SCU and OCM is mapped low
	DDR	OCM	OCM	Address filtered by SCU and OCM is mapped low
	DDR			Address filtered by SCU and OCM is not mapped low
				Address not filtered by SCU and OCM is not mapped low
0004_0000 to 0007_FFFF	DDR			Address filtered by SCU
				Address not filtered by SCU
0008_0000 to 000F_FFFF	DDR	DDR	DDR	Address filtered by SCU
		DDR	DDR	Address not filtered by SCU[3]
0010_0000 to 3FFF_FFFF	DDR	DDR	DDR	Accessible to all interconnect masters
4000_0000 to 7FFF_FFFF	PL		PL	General Purpose Port #0 to the PL, M_AXI_GP0
8000_0000 to BFFF_FFFF	PL		PL	General Purpose Port #1 to the PL, M_AXI_GP1
E000_0000 to E02F_FFFF	IOP		IOP	I/O Peripheral registers, see Table 4-6
E100_0000 to E5FF_FFFF	SMC		SMC	SMC Memories, see Table 4-5
F800_0000 to F800_0BFF	SLCR		SLCR	SLCR registers, see Table 4-3
F800_1000 to F880_FFFF	PS		PS	PS System registers, see Table 4-7
F890_0000 to F8F0_2FFF	CPU			CPU Private registers, see Table 4-4
FC00_0000 to FDFF_FFFF [4]	Quad-SPI		Quad-SPI	Quad-SPI linear address for linear mode
FFFC_0000 to FFFF_FFFF [2]	OCM	OCM	OCM	OCM is mapped high
				OCM is not mapped high

Source: Zynq-7000 All Programmable SoC Technical Reference Manual (UG585)

Zynq AXI Configurations

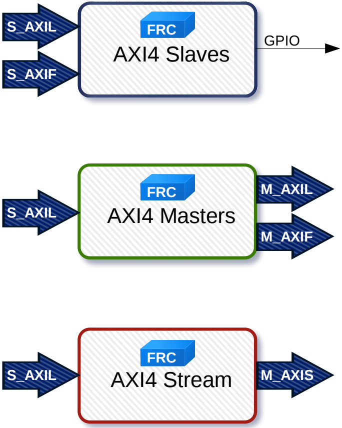


To enable cache coherency with ACP, the AXI signals AxCACHE must be **XX11** and AxUSER must have all its bits tie high.

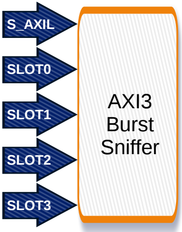
Developed IPs

Free Running Counter

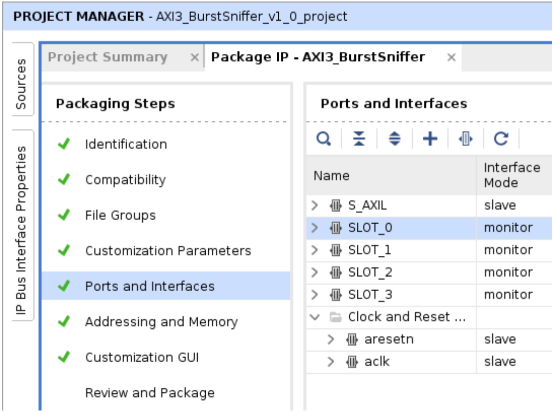
```
counter_proc: process(aclk)
begin
  if (rising_edge (aclk)) then
    if (aresetn = '0') then
      counter <= (others => '0');
    else
      if enable = '1' then
        counter <= counter + 1;
      else
        counter <= (others => '0');
      end if;
    end if;
  end if;
end process counter_proc;
```



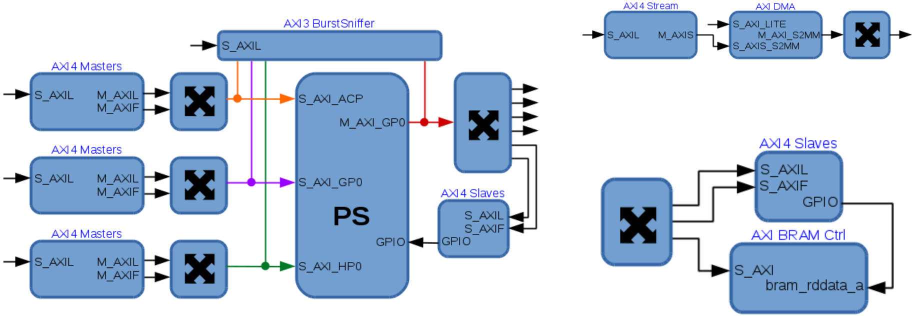
AXI3 Burst Sniffer



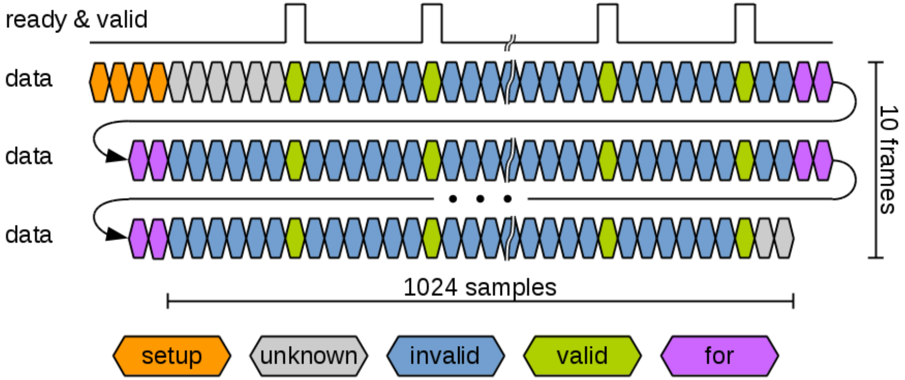
SLOT0-3 are AXI3 interfaces in monitor mode, which have only INPUT ports.



Block Designs



Cycles measurement in the PS



Cycles measurement in the PS

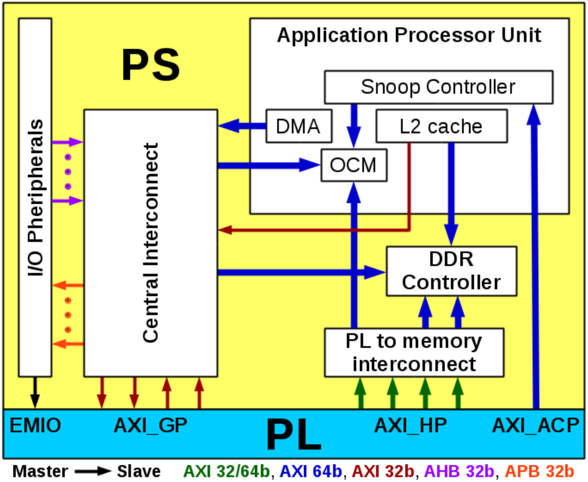
```
int data[ROWS][COLS] __attribute__ ((aligned (32)));
...
int row, col;
```

```
pl_cycles = data[row][COLS-1]-data[row][0]
```

```
#include "xtime_l.h"
...
XTime tStart[ROWS], tEnd[ROWS];
...
XTime_GetTime(&tStart[row]);
... // do something to be measured here
XTime_GetTime(&tEnd[row]);
...
ps_cycles = 2 * (tEnd[0]-tStart[0]);
```

$$MB/s = \frac{FREQUENCY * SAMPLES * BYTES}{CYCLES}$$

Zynq Interfaces Summary

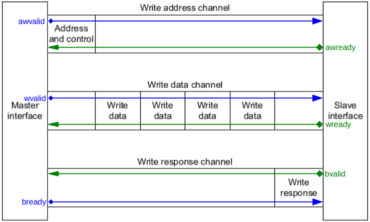


Measured cycles

Interface	Test Case Variant	Burst	Between Data min	typ	max	Per Frame PS (MB/s)	PL (MB/s)	PS/PL
EMIO	GPIO (XGpioPs_Read)	No	20	21	29	96954 (27.46)	22358 (27.48)	4.33
EMIO	GPIO (*Xil_In32*)	No	20	20	31	92502 (28.78)	21330 (28.80)	4.33
M_AXI_GP	AXI Lite (*Xil_In32*)	No	28	28	33	124386 (21.40)	28689 (21.41)	4.33
M_AXI_GP	AXI Full (*Xil_In32*)	No	24	24	26	106588 (24.97)	24581 (24.99)	4.33
M_AXI_GP	AXI Lite (*memcpy*)	No	19	20	31	90973 (29.26)	20974 (29.29)	4.33
M_AXI_GP	AXI Full (*memcpy*)	No	15	16	25	73336 (36.30)	16910 (36.33)	4.33
S_AXI_GP	AXI Lite	No	44	44	45	200229 (13.29)	46075 (13.33)	4.34
S_AXI_HP	AXI Lite	No	36	36	37	160386 (16.59)	36865 (16.66)	4.35
S_AXI_ACP	AXI Lite	No	36	36	36	160389 (16.59)	36864 (16.66)	4.35
S_AXI_GP	AXI Full	Yes	1	4	59	21962 (121.22)	4868 (126.21)	4.51
S_AXI_HP	AXI Full	Yes	1	3	40	16669 (159.72)	3675 (167.18)	4.53
S_AXI_ACP	AXI Full	Yes	1	3	37	15506 (171.70)	3409 (180.22)	4.54
M_AXI_GP	AXI Full with PS DMA	Yes	1	1	4	11425 (233.3)	1213 (506.51)	9.41
S_AXI_GP	AXI Full with AXI DMA	Yes	1	1	571	7245 (367.48)	1654 (371.46)	4.38
S_AXI_HP	AXI Full with AXI DMA	Yes	1	1	381	6048 (440.21)	1397 (439.79)	4.32
S_AXI_ACP	AXI Full with AXI DMA	Yes	1	1	422	6154 (432.62)	1418 (433.28)	4.33

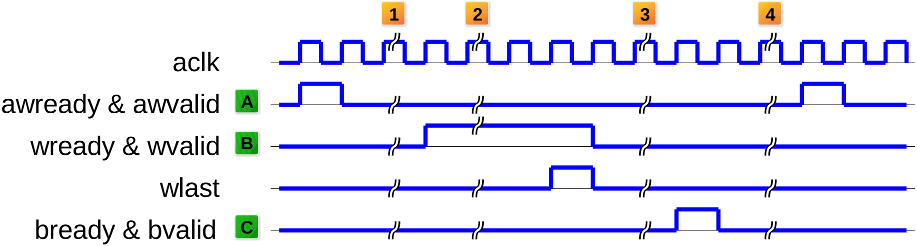
The ideal PS/PL relation is 650 MHz/150 MHz = 4.33

Custom AXI master vs AXI DMA



Custom AXI master (GP example)

- 3 cycles between A and B
- 16 cycles in B
- 36 cycles between B and C
- 21 cycles between C and a new A



Custom AXI Master improvment

Test Case		Between Data			Per Frame		
Interface	Variant	min	typ	max	PS (MB/s)	PL (MB/s)	PS/PL
S_AXI_GP	AXI Lite	44	44	45	200229 (13.29)	46075 (13.33)	4.34
S_AXI_HP	AXI Lite	36	36	37	160386 (16.59)	36865 (16.66)	4.35
S_AXI_ACP	AXI Lite	36	36	36	160389 (16.59)	36864 (16.66)	4.35
S_AXI_GP	AXI Full	1	4	59	21962 (121.22)	4868 (126.21)	4.51
S_AXI_HP	AXI Full	1	3	40	16669 (159.72)	3675 (167.18)	4.53
S_AXI_ACP	AXI Full	1	3	37	15506 (171.70)	3409 (180.22)	4.54
S_AXI_GP	AXI Full with AXI DMA	1	1	571	7245 (367.48)	1654 (371.46)	4.38
S_AXI_HP	AXI Full with AXI DMA	1	1	381	6048 (440.21)	1397 (439.79)	4.32
S_AXI_ACP	AXI Full with AXI DMA	1	1	422	6154 (432.62)	1418 (433.28)	4.33

↓↓↓↓↓

Test Case		Between Data			Per Frame		
Interface	Variant	min	typ	max	PS	PL	PS/PL
S_AXI_GP	AXI Lite	3	3	4	14382 (185.12 MB/s)	3187 (192.78 MB/s)	4.51
S_AXI_HP	AXI Lite	3	3	3	13952 (190.82 MB/s)	3072 (200. 0 MB/s)	4.54
S_AXI_ACP	AXI Lite	3	5	8	26769 (99.45 MB/s)	5963 (103. 3 MB/s)	4.48
S_AXI_GP	AXI Full	1	1	5	6677 (398.74 MB/s)	1406 (436.98 MB/s)	4.74
S_AXI_HP	AXI Full	1	1	4	6456 (412.39 MB/s)	1342 (457.82 MB/s)	4.81
S_AXI_ACP	AXI Full	1	1	5	6684 (398.32 MB/s)	1406 (436.98 MB/s)	4.75

Conclusions

- If burst transactions will not be used (neither DMA or cache) use AXI Lite interfaces (they are simpler and less PL resources are consumed).
- The AXI interfaces provided by the IP packager could/must be improved:
  - AXI Lite interfaces consume an extra cycle per operation.
  - AXI Full slave do not work with burst.
  - The address phase of AXI Full master can be changed to be at same time that TLAST (is what AXI DMA does).
  - The write response channel can be ignored to improve the data rate (is what AXI DMA does but IS NOT COMPLIANT WITH THE AMBA AXI SPEC).
- When 32-bit data is used in 64-bit interfaces, the burst transactions involves 64-bit transfer with one cycle between them.
- The PS DMA driver seems that could be improved to obtain very high data rates.
- The main disadvantage in GP interfaces is the 32-bit data width, due that slightly lower data rates are observed compared with HP/ACP.

INTI-CMNB-FPGA

Rodrigo A. Melo
rmelo@inti.gob.ar
rodrigoalejandromelo
@rodrigomelo9ok
rodrigomelo9

Bruno Valinoti
valinoti@inti.gob.ar
bruno-valinoti

Thanks!