Application Note 307

Example LogicTile Express 20MG design for a CoreTile Express A15x2_A7x3.

Document number: ARM DAI 0307

Issued: May 2013

ARM

Application Note 307
Example LogicTile Express 20MG design for a CoreTile Express A15x2_A7x3

Release information

The following changes have been made to this Application Note.

Change history

Date	Issue	Change
October 22, 2012	A	First release
February 24, 2013	B	Second release
May 15, 2013	C	Added information for V2F-1XV7 revision C

Proprietary notice

Confidentiality status

This document is Open Access. This document has no restriction on distribution.

Feedback on this Application Note

If you have any comments on this Application Note, please send email to errata@arm.com giving:

- the document title
- the document number
- the page number(s) to which your comments refer
- an explanation of your comments.

General suggestions for additions and improvements are also welcome.

ARM web address

http://www.arm.com/

Table of Contents

1 Introduction

1.1 Purpose of this application note

This application note discusses the operation of an example design for a LogicTile Express 20MG (V2F-1XV7) operating alongside a CoreTile Express A15x2_A7x3 (V2P-CA15-A7) processor board. It examines the contents of the V2F-1XV7, the system interconnect, the clock structure, and specifics of the programmer's model directly relevant to V2F-1XV7 operation.

Having read this application note, the user should be in a position to debug and analyze the operation of the provided example design. He also should be able to make changes to the example design, connect his own AXI or AHB based masters, and AXI, AHB or APB slaves.

1.2 Overview of hardware platform

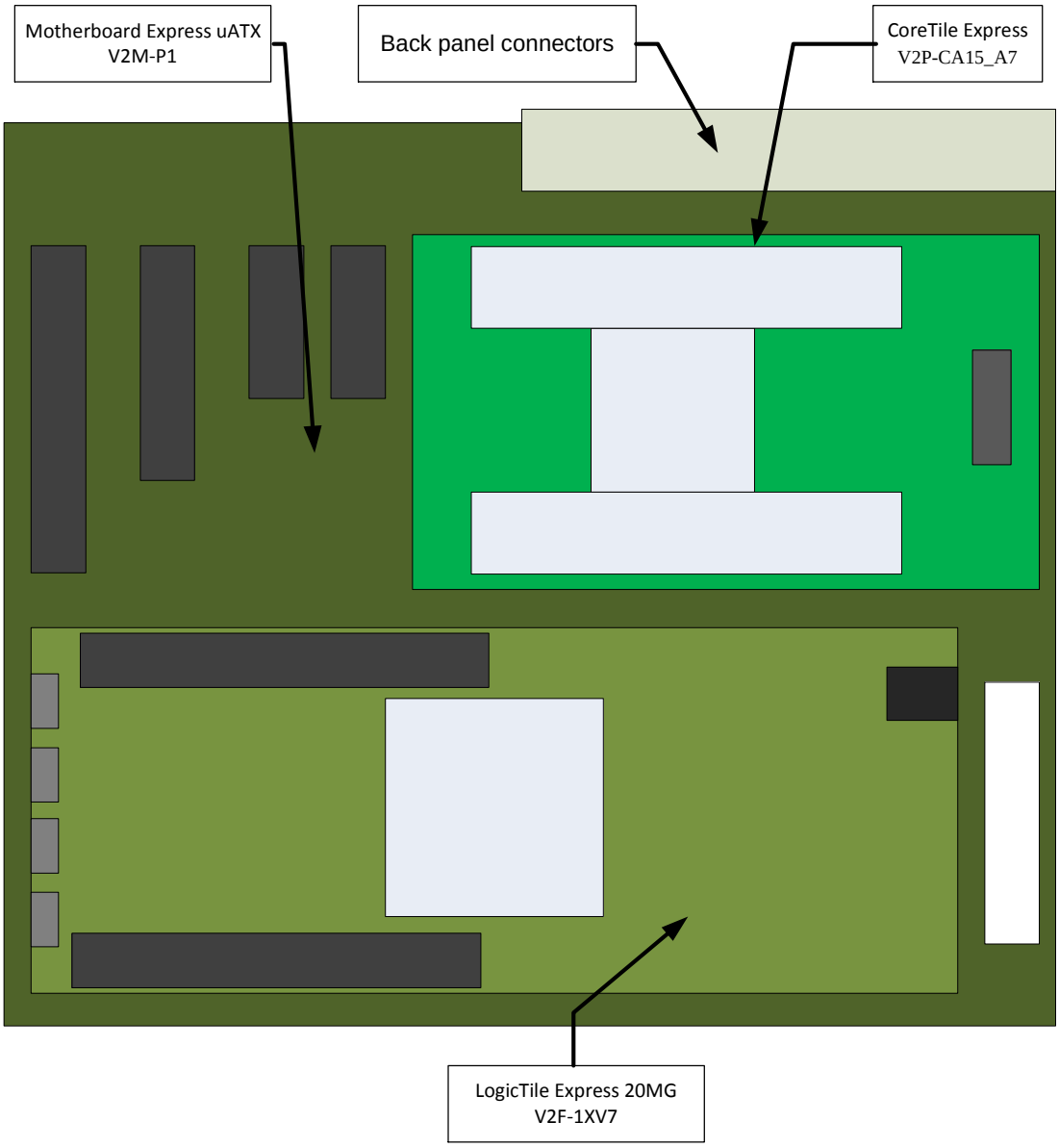

Figure 1-1 V2F-1XV7 and V2P-CA15-A7 daughterboards on a V2M-P1 motherboard

The example design is intended to work on a Motherboard Express uATX (V2M-P1) with a CoreTile Express A15x2_A7x3 (V2P-CA15-A7) mounted on Site 1 and a LogicTile Express 20MG (V2F-1XV7) mounted on Site 2.

2 Getting Started

The steps below show you how to set up the hardware platform and copy the necessary configuration files to the V2M-P1 motherboard USB Flash disk and program the LogicTile Express FPGA image.

1. Mount the V2P-CA15-A7 daughterboard on site 1 and the V2F-1XV7 daughterboard on site 2 of the V2M-P1 motherboard as described in:

Quick Start Guide for the Versatile Express Family - Adding Daughterboards.

2. Connect USB, UART0, and power cables, and power up the boards as described in:

Quick Start Guide for the Versatile Express Family - Powering up the System.

3. After the host computer has recognized the motherboard Micro-SD card as a USB mass storage device, you must copy the application note boardfiles on to the card. If this application note was obtained from a Versatile Express DVD, the easiest way to copy the boardfiles on to the card is to follow the system recovery procedure given in the release notes supplied with the DVD.

If you don't have the Versatile Express DVD, you must add the boardfiles to the existing contents of your motherboard Micro-SD card by copying the `boardfiles/SITE2/HBI0247*/AN307` directories from the application note into the `/SITE2/HBI0247*` directories on the motherboard Micro-SD card. For a revision B V2F-1XV7 board, copy the `boardfiles/SITE2/HBI0247B/AN307` directory from the application note into `/SITE2/HBI0247B` on the motherboard Micro-SD card. For a revision C V2F-1XV7 board, copy the `boardfiles/SITE2/HBI0247C/AN307` directory from the application note into `/SITE2/HBI0247C` on the motherboard Micro-SD card.

4. Edit the `board.txt` file located in `/SITE2/HBI0247B/` or `/SITE2/HBI0247C/` on the motherboard Micro-SD card. You must modify the APPNOTE field in board.txt to comment out the line containing ANxxx and uncomment the line containing AN307. (Note: The bitfiles for the revision B and the revision C boards are not identical).

For a revision B V2F-1XV7 board, the APPNOTE field should look like this after editing:

```
[APPLICATION NOTE]
;APPNOTE: ANxxx\axxxrxpx.txt   ;Please select the required application note
;APPNOTE: AN306\a306r0p1.txt
APPNOTE: AN307\a307r1p1.txt
```

For a revision C V2F-1XV7 board, the APPNOTE field should look like this after editing:

```
[APPLICATION NOTE]
;APPNOTE: ANxxx\axxxrxpx.txt   ;Please select the required application note
;APPNOTE: AN306\a306r1p0.txt
APPNOTE: AN307\a307r2p0.txt
```

5. Power cycle the boards by pressing the black button and then the red button. The application note bitfile will be uploaded to the V2F-1XV7 FPGA.

6. The system will now be fully configured and ready for use.

3 System architecture

This system is an AXI (AMBA 3.0) based system. The example V2F-1XV7 image exposes one muxed 64 bit AXI slave port EMM (External Muxed Master) to the V2P-CA15-A7 daughterboard and Static Memory Bus, Multimedia Bus, System Bus to the V2M-P1 motherboard.

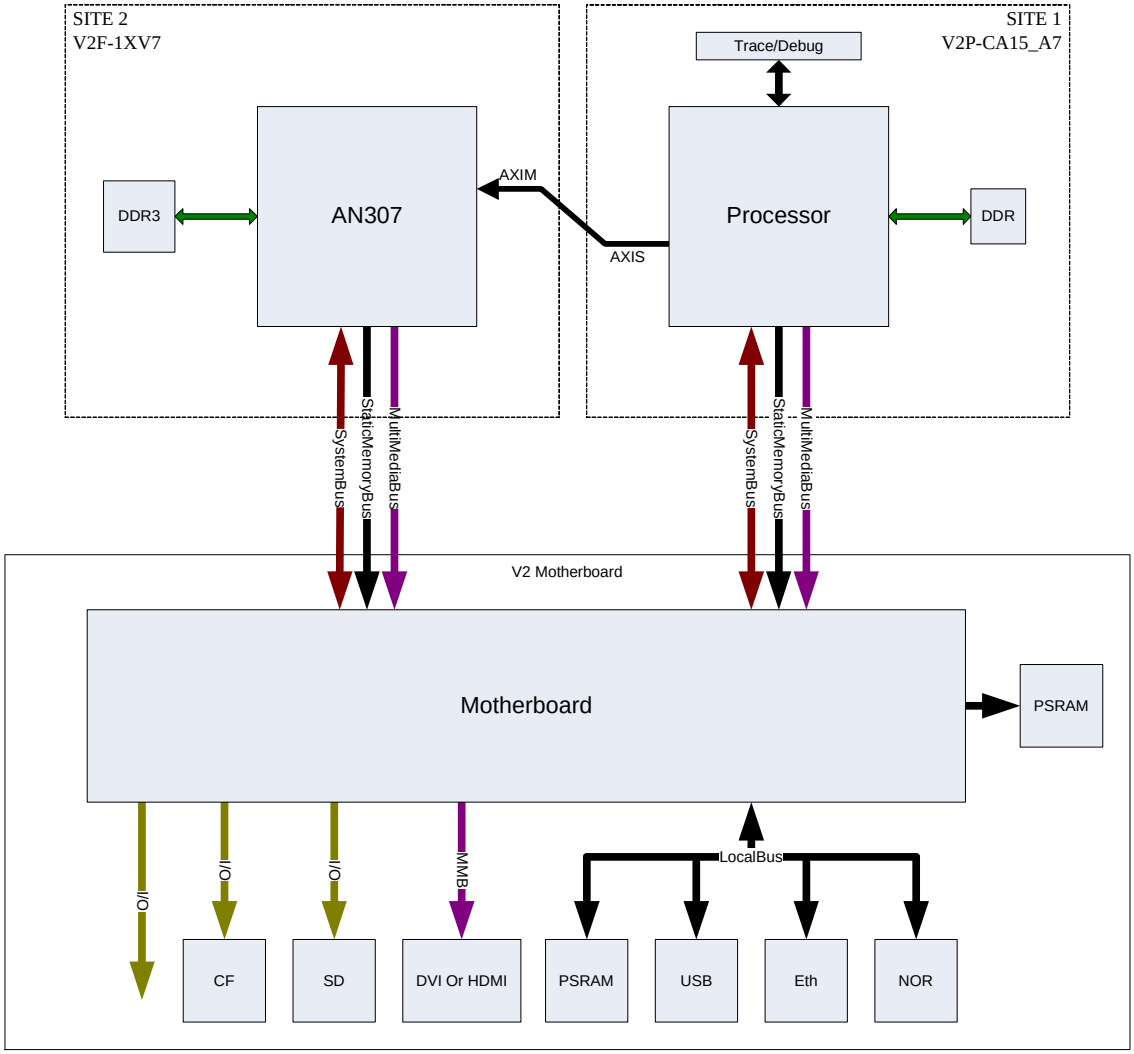

Figure 3-1 Block level architecture

Note that the direction of the arrows indicates the direction of control, i.e. it points from the Master to the Slave. An AXI bus contains signals going in both directions.

3.1 AN307 architecture

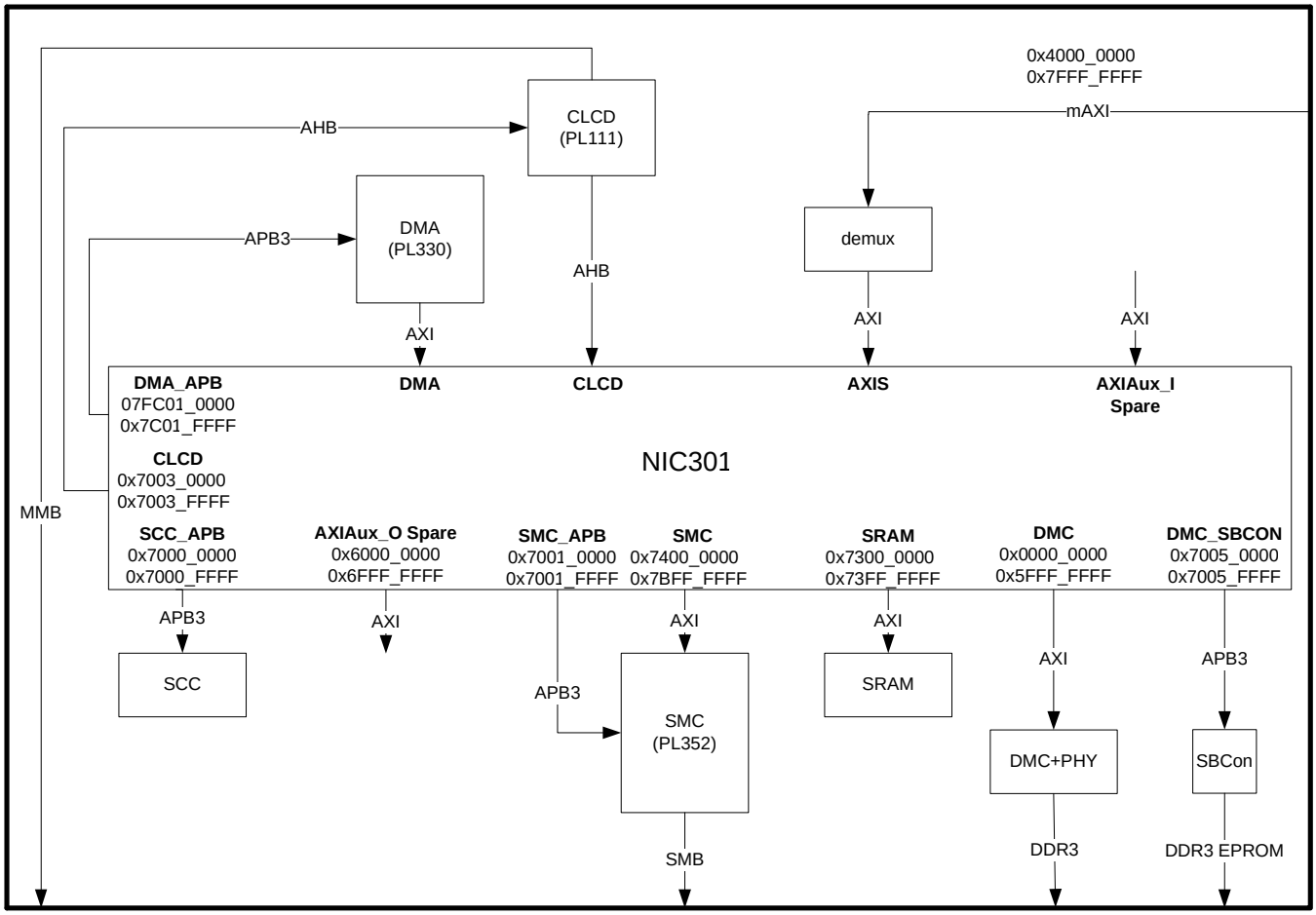

Figure 3-2 AN307 Architecture

3.2 Module functionality

The function of each of these blocks is as follows:

NIC - 301 Bus Matrix

This provides the interconnect structure. It allows any of the 3 AXI and 1 AHB slave ports to connect to one of the 4 AXI, 1 AHB and 4 APB master ports without blocking the other masters (unless they both try to access the same slave).

It also contains the decoder mapping to determine the address map, and a scheme to determine priority of competing masters to a single slave.

See the **AMBA Network Interconnect (NIC-301) Technical Reference Manual** for more information.

PL111 CLCD

This is a Color Liquid Crystal Display controller. The ARM PrimeCell PL111 is used in this design. See the **PrimeCell Color LCD Controller (PL111) Technical Reference Manual** for more information.

DxAsynchAxi_V2

This block provides an AXI demux block which demultiplexes the 2:1 muxed 64 bit EMM AXI bus and a Aysnchronous Bridge (BP132) to allow multiplexed bus to operate with best possible timing. The register slice increases the internal operating frequency and introduces an extra clock cycle of latency.

PL330 DMAC

This is a Direct Memory Access Component. The ARM PrimeCell PL330 is used in this design. For more information please refer to the PrimeCell documentation.

PL354 SMC

This is a Static Memory Controller. The ARM PrimeCell PL354 is used in this design. See the **PrimeCell Static Memory Controller (PL350 series) Technical Reference Manual** for more information.

DMC

The DMC (Dynamic Memory Controller) is a Xilinx core instantiated in the design. The edf is not included and must be regenerated by the user if rebuilding the design.

SCC

This is example of a Serial Configuration Controller (SCC). The SCC provides a standard serial interface to a V2F-1XV7 Daughterboard Configuration Controller (DCC). The DCC uses this interface by issuing commands to receive/transmit information from/to the SCC registers in the FPGA.

All DCC defined commands have been supported. The SCC enables identification and of systems, passes DCM lock status to V2M-P1 motherboard reset controller as well to light the V2F-1XV7 daughterboard LEDs and update an internal register according to the states of the V2F-1XV7 switches. The SCC also provides two general purpose user registers which value can be uploaded during power up sequence from configuration file.

For more information about DCC interface and commands please refer to **LogicTile Express 20MG TRM**.

The SCC registers are also accessible via APB bus which allows to access SCC registers via AN307 bus matrix. For programming model see 5.2 SCC registers

3.3 Modules revision

Module	Revision
NIC-301 Bus Matrix	r2p1
PL111 CLCD	r0p2
DxAsynchAxi_V2	r0p0
PL330 DMAC	r1p0
PL354 SMC	r2p1
SCC	r0p0

Table 1 Modules revision in AN307 r1p0

3.4 Clock architecture

The clock architecture is carefully designed to minimize the skew (difference) in the clock edge position between different components across the system. The DCMs and clock loops on V2F-1XV7 have been used to achieve that.

For the maximum and default frequencies of clocks please refer to **8.1 Default, minimum and maximum operating frequencies**.

For information how to set up and change OSC0-OSC5 clocks frequency on please refer to **Motherboard Express uATX TRM**.

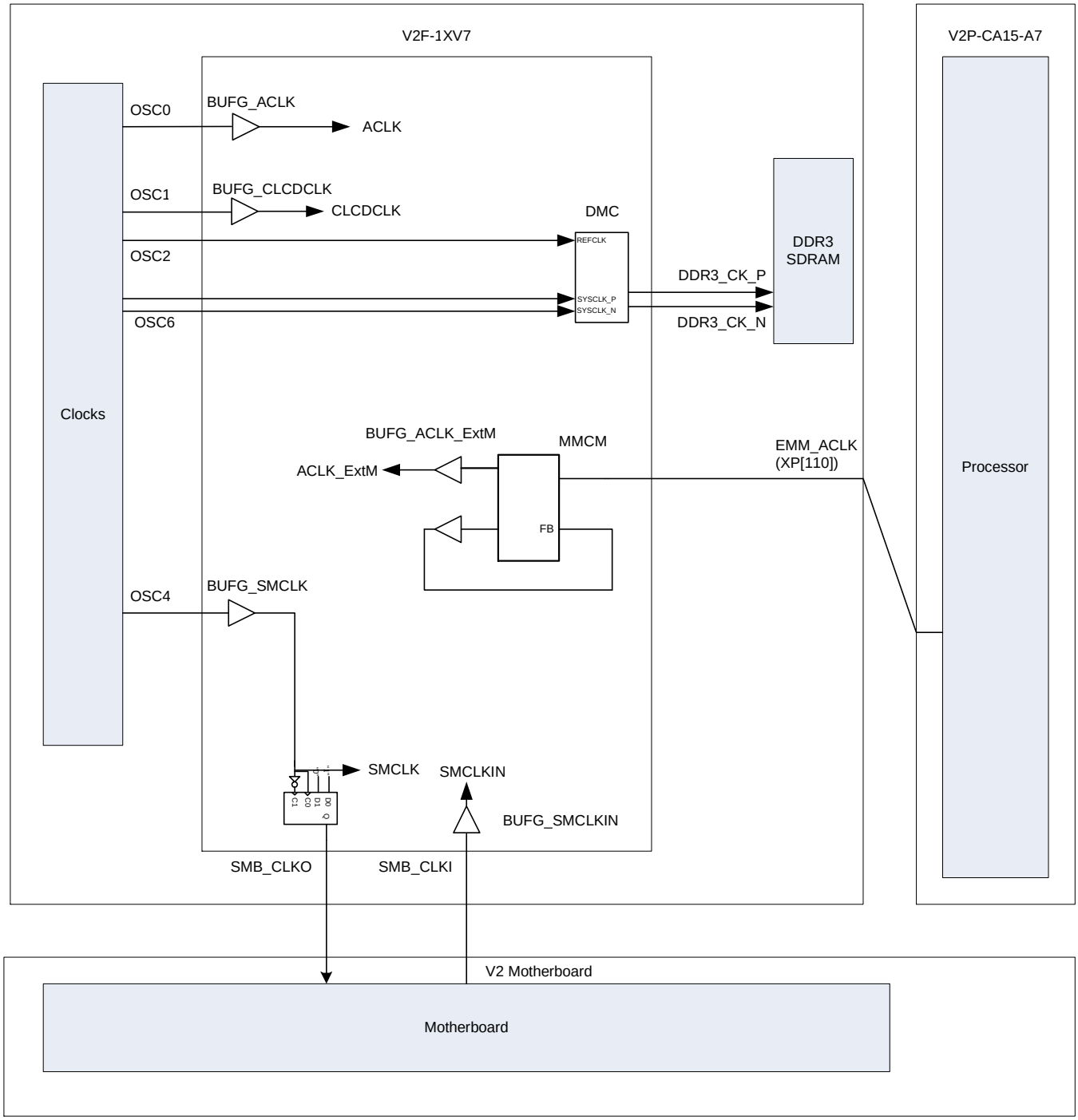

Figure 3-3 Clock architecture for AN307

There are 5 clock domains in this design:

AN307 AXI System Bus

The Example Internal AXI System Bus is clocked by ACLK. The ACLK is generated through a clock buffer from OSC0. The Internal AXI System Bus clock domain includes:

- Bus Matrix
- DMA Controller
- CLCD controller AHB buses
- SMC AXI bus
- De-multiplexing AXI master bus
- APB subsystem

DMC

OSC2 – as a DDRREFCLK for DDR3 Controller. And Differential OSC6 as DDRSYSCLK for DDR3 PHY.

CLCD

CLCDCLK is directly connected to OSC1. It is reference clock for the CLCD controller. The frequency of this clock must be adjusted to match target screen resolution.

Static Memory Bus

The SMCLK is used to drive Static Memory Controller and Static Memory bus. The SMCLK is directly connected to OSC4. The SMCLKIN feedback clock is used to register data from Static memory bus slave implemented on V2 Motherboard.

External AXI Master Bus

The External AXI Slave Bus is clocked by EMM_ACLK provided by an external daughterboard (V2F-1XV7).

3.5 Interrupt architecture

There are 4 interrupts sources generated by AN307 which are connected to the processor's daughterboard interrupt controller via the V2M-P1 motherboard. The interrupts signals have been inverted in AN307 top level and V2M-P1 motherboard to provide the correct polarity for the interrupt controller in the CoreTile Express A15x2_A7x3.

Signal	Source	V2M-P1 interrupts	V2P_CA15x2_A7x3 interrupts
Clcdintr	CLCD	SB_INT[0]	INTSOURCE[22]
smc_int	SMC	SB_INT[1]	INTSOURCE[23]
dma_int[0]	DMAC	SB_INT[2]	INTSOURCE[24]
dma_int[1]	DMAC	SB_INT[3]	INTSOURCE[25]

Table 2 Interrupts

3.6 Reset architecture

The reset signal CB_nRST from the motherboard is synchronized to the correct clock domains to reset all the peripherals in the Versatile Express FPGA Daughterboard.

The reset signal nPLLRESET from the DCC controller is used to reset MMCM during power-up sequence.

The reset signal CB_nPOR is not used in the design.

4 Hardware description

4.1 V2F-1XV7 wrapper (V2F_1XV7_wrapper)

This level defines the mapping from the V2F-1XV7 XN, XP and XS busses into their functional allocations. The wrapper contains also clock structure and ties off static pins.

4.2 AN307 top level (AN307_toplevel)

This level connects all the Application Note components together. This includes all the major modules as shown in **Figure 3-2 AN307 Architecture**. The top level RTL is provided so modules can be added and removed by the user.

All modules except SCC are high value AXI blocks, and are only provided as .NGC netlists.

Two constrain files (xcd) are included, for the application note which shows all FPGA pins used on the V2F-1XV7 daughterboard and timing constraints.

4.3 AXI demultiplexing scheme

DDR registers have been used to implement 2:1 demultiplexing to reduce the pin count for the two AXI buses for implementation on the X header.

The registers are part of the AXI regslice logic therefore multiplexing does introduce an extra cycle of latency to the AXI bus.

The xVALID and xREADY signals on AXI are not multiplexed in this way due to their timing requirements and are passed directly between devices.

For multiplexed AXI timing requirements please refer to **8.2 AXI timing requirements**.

The demultiplexing blocks are provided as NGC netlist.

4.4 Header HDRX pin allocation

The AXI bus connects to the HDRX as shown. Multiplexed AXI slave bus EMM connects via HDRX header to AXI External Master.

4.4.1 Multiplexed AXI slave bus

HDRX pin	X bus	Signal (Hi/Lo)	HDRX pin	X bus	Signal (Hi/Lo)
D50	XN153	EMM_WDATA0/32	E48	XP154	EMM_WSTRB3/7
D49	XP153	EMM_WDATA1/33	D26	XS13	EMM_WVALID
C49	XN152	EMM_WDATA2/34	C26	XS12	EMM_WREADY
C48	XP152	EMM_WDATA3/35	K41	XN129	EMM_AWADDR0/16
B50	XN151	EMM_WDATA4/36	K40	XP129	EMM_AWADDR1/17
B49	XP151	EMM_WDATA5/37	J40	XN128	EMM_AWADDR2/18
A49	XN150	EMM_WDATA6/38	J39	XP128	EMM_AWADDR3/19
A48	XP150	EMM_WDATA7/39	H41	XN127	EMM_AWADDR4/20
K47	XN149	EMM_WDATA8/40	H40	XP127	EMM_AWADDR5/21
K46	XP149	EMM_WDATA9/41	G40	XN126	EMM_AWADDR6/22
J46	XN148	EMM_WDATA10/42	G39	XP126	EMM_AWADDR7/23
J45	XP148	EMM_WDATA11/43	F41	XN125	EMM_AWADDR8/24
H47	XN147	EMM_WDATA12/44	F40	XP125	EMM_AWADDR9/25
H46	XP147	EMM_WDATA13/45	E40	XN124	EMM_AWADDR10/26
G46	XN146	EMM_WDATA14/46	E39	XP124	EMM_AWADDR11/27
G45	XP146	EMM_WDATA15/47	D41	XN123	EMM_AWADDR12/28
F47	XN145	EMM_WDATA16/48	D40	XP123	EMM_AWADDR13/29
F46	XP145	EMM_WDATA17/49	C40	XN122	EMM_AWADDR14/30
E46	XN144	EMM_WDATA18/50	C39	XP122	EMM_AWADDR15/31
E45	XP144	EMM_WDATA19/51	D44	XN133	EMM_AWID1/0
D47	XN143	EMM_WDATA20/52	D43	XP133	EMM_AWID3/2
D46	XP143	EMM_WDATA21/53	C43	XN132	EMM_AWID5/4
C46	XN142	EMM_WDATA22/54	C42	XP132	EMM_AWID7/6
C45	XP142	EMM_WDATA23/55	B44	XN131	EMM_AWID9/8
B47	XN141	EMM_WDATA24/56	B43	XP131	EMM_AWID11/10
B46	XP141	EMM_WDATA25/57	A43	XN130	EMM_AWID13/12
A46	XN140	EMM_WDATA26/58	H43	XP137	EMM_AWPROT2/AWID14
A45	XP140	EMM_WDATA27/59	E43	XN134	EMM_AWLEN0/2
K44	XN139	EMM_WDATA28/60	E42	XP134	EMM_AWLEN1/3
K43	XP139	EMM_WDATA29/61	G42	XP136	EMM_AWSIZE0/1
J43	XN138	EMM_WDATA30/62	H44	XN137	EMM_AWPROT0/1
J42	XP138	EMM_WDATA31/63	G43	XN136	EMM_AWBURST0/1
K50	XN159	EMM_WID1/0	A42	XP130	EMM_AWLOCK0/1
K49	XP159	EMM_WID3/2	F44	XN135	EMM_AWCACHE0/2
J49	XN158	EMM_WID5/4	F43	XP135	EMM_AWCACHE1/3
J48	XP158	EMM_WID7/6	F26	XS15	EMM_AWVALID
H50	XN157	EMM_WID9/8	E26	XS14	EMM_AWREADY
H49	XP157	EMM_WID11/10	A31	XN90	EMM_BID0/1
G49	XN156	EMM_WID13/12	A30	XP90	EMM_BID2/3
G48	XP156	EMM_WLAST/WID14	C28	XN82	EMM_BID4/5
F50	XN155	EMM_WSTRB0/4	C27	XP82	EMM_BID6/7
F49	XP155	EMM_WSTRB1/5	B29	XN81	EMM_BID8/9
E49	XN154	EMM_WSTRB2/6	B28	XP81	EMM_BID10/11

HDRX pin	X bus	Signal (Hi/Lo)	HDRX pin	X bus	Signal (Hi/Lo)
A28	XN80	EMM_BID12/13	F32	XN95	EMM_RDATA8/40
A27	XP80	EMM_BID14/1'b0	F31	XP95	EMM_RDATA9/41
A33	XP100	EMM_BRESP0/1	E31	XN94	EMM_RDATA10/42
A26	XS10	EMM_BVALID	E30	XP94	EMM_RDATA11/43
B26	XS11	EMM_BREADY	D32	XN93	EMM_RDATA12/44
D38	XN113	EMM_ARADDR0/16	D31	XP93	EMM_RDATA13/45
D37	XP113	EMM_ARADDR1/17	C31	XN92	EMM_RDATA14/46
C37	XN112	EMM_ARADDR2/18	C30	XP92	EMM_RDATA15/47
C36	XP112	EMM_ARADDR3/19	B32	XN91	EMM_RDATA16/48
B38	XN111	EMM_ARADDR4/20	B31	XP91	EMM_RDATA17/49
B37	XP111	EMM_ARADDR5/21	K29	XN89	EMM_RDATA18/50
K35	XN109	EMM_ARADDR6/22	K28	XP89	EMM_RDATA19/51
K34	XP109	EMM_ARADDR7/23	J28	XN88	EMM_RDATA20/52
J34	XN108	EMM_ARADDR8/24	J27	XP88	EMM_RDATA21/53
J33	XP108	EMM_ARADDR9/25	H29	XN87	EMM_RDATA22/54
H35	XN107	EMM_ARADDR10/26	H28	XP87	EMM_RDATA23/55
H34	XP107	EMM_ARADDR11/27	G28	XN86	EMM_RDATA24/56
G34	XN106	EMM_ARADDR12/28	G27	XP86	EMM_RDATA25/57
G33	XP106	EMM_ARADDR13/29	F29	XN85	EMM_RDATA26/58
F35	XN105	EMM_ARADDR14/30	F28	XP85	EMM_RDATA27/59
F34	XP105	EMM_ARADDR15/31	E28	XN84	EMM_RDATA28/60
H38	XN117	EMM_ARID1/0	E27	XP84	EMM_RDATA29/61
H37	XP117	EMM_ARID3/2	D29	XN83	EMM_RDATA30/62
G37	XN116	EMM_ARID5/4	D28	XP83	EMM_RDATA31/63
G36	XP116	EMM_ARID7/6	D35	XN103	EMM_RID0/1
F38	XN115	EMM_ARID9/8	D34	XP103	EMM_RID2/3
F37	XP115	EMM_ARID11/10	C34	XN102	EMM_RID4/5
E37	XN114	EMM_ARID13/12	C33	XP102	EMM_RID6/7
B40	XP121	EMM_ARPROT2/ARID14	B35	XN101	EMM_RID8/9
J37	XN118	EMM_ARLEN0/2	B34	XP101	EMM_RID10/11
J36	XP118	EMM_ARLEN1/3	A34	XN100	EMM_RID12/13
A39	XP120	EMM_ARSIZE0/1	E33	XP104	EMM_RID14/RLAST
B41	XN121	EMM_ARPROT0/1	E34	XN104	EMM_RRESP1/0
A40	XN120	EMM_ARBURST0/1	J26	XS18	EMM_RVALID
E36	XP114	EMM_ARLOCK0/1	K26	XS19	EMM_RREADY
K38	XN119	EMM_ARCACHE0/2	A37	XN110	NC
K37	XP119	EMM_ARCACHE1/3	A36	XP110	EMM_ACLK
G26	XS16	EMM_ARVALID			
H26	XS17	EMM_ARREADY			
K32	XN99	EMM_RDATA0/32			
K31	XP99	EMM_RDATA1/33			
J31	XN98	EMM_RDATA2/34			
J30	XP98	EMM_RDATA3/35			
H32	XN97	EMM_RDATA4/36			
H31	XP97	EMM_RDATA5/37			
G31	XN96	EMM_RDATA6/38			
G30	XP96	EMM_RDATA7/39			

Table 3 Header HRDX EMM bus pin allocation

4.5 Header HDRY pin allocation

4.5.1 Multimedia bus

HDRY pin	MMB bus	CLCD Signal	HDRY pin	MMB bus	CLCD Signal
A46	MMB_DATA0	CLD0	J45	MMB_DATA18	CLD18
B46	MMB_DATA1	CLD1	K45	MMB_DATA19	CLD19
C46	MMB_DATA2	CLD2	A43	MMB_DATA20	CLD20
D46	MMB_DATA3	CLD3	B43	MMB_DATA21	CLD21
E46	MMB_DATA4	CLD4	C43	MMB_DATA22	CLD22
F46	MMB_DATA5	CLD5	D43	MMB_DATA23	CLD23
G46	MMB_DATA6	CLD6	G43	MMB_DE	CLAC
H46	MMB_DATA7	CLD7	E43	MMB_HS	CLLP
J46	MMB_DATA8	CLD8	K43	MMB_IDCLK	CLCP
K46	MMB_DATA9	CLD9	F43	MMB_VS	CLFP
A45	MMB_DATA10	CLD10			
B45	MMB_DATA11	CLD11			
C45	MMB_DATA12	CLD12			
D45	MMB_DATA13	CLD13			
E45	MMB_DATA14	CLD14			
F45	MMB_DATA15	CLD15			
G45	MMB_DATA16	CLD16			
H45	MMB_DATA17	CLD17			

Table 4 Multimedia bus pin allocation

4.5.2 Static memory bus

HDRY pin	SMB bus	SMC Signal	HDRY pin	SMB bus	SMC Signal
C26	SMB_ADDR0	add_0	B29	SMB_DATA11	data_11
D26	SMB_ADDR1	add_1	C29	SMB_DATA12	data_12
E26	SMB_ADDR2	add_2	D29	SMB_DATA13	data_13
F26	SMB_ADDR3	add_3	E29	SMB_DATA14	data_14
G26	SMB_ADDR4	add_4	F29	SMB_DATA15	data_15
H26	SMB_ADDR5	add_5	G29	SMB_DATA16	data_16
J26	SMB_ADDR6	add_6	H29	SMB_DATA17	data_17
K26	SMB_ADDR7	add_7	J29	SMB_DATA18	data_18
A24	SMB_ADDR8	add_8	K29	SMB_DATA19	data_19
B24	SMB_ADDR9	add_9	A27	SMB_DATA20	data_20
C24	SMB_ADDR10	add_10	B27	SMB_DATA21	data_21
D24	SMB_ADDR11	add_11	C27	SMB_DATA22	data_22
E24	SMB_ADDR12	add_12	D27	SMB_DATA23	data_23
F24	SMB_ADDR13	add_13	E27	SMB_DATA24	data_24
G24	SMB_ADDR14	add_14	F27	SMB_DATA25	data_25
H24	SMB_ADDR15	add_15	G27	SMB_DATA26	data_26
J24	SMB_ADDR16	add_16	H27	SMB_DATA27	data_27
K24	SMB_ADDR17	add_17	J27	SMB_DATA28	data_28
A23	SMB_ADDR18	add_18	K27	SMB_DATA29	data_29

SMB bus	SMB bus	SMB bus	SMB bus	SMB bus	SMB bus
B23	SMB_ADDR19	add_19	A26	SMB_DATA30	data_30
C23	SMB_ADDR20	add_20	B26	SMB_DATA31	data_31
D23	SMB_ADDR21	add_21	F20	SMB_nADV	adv_n
E23	SMB_ADDR22	add_22	G20	SMB_nBAA	baa_n
F23	SMB_ADDR23	add_23	K23	SMB_nBLS0	bls_n_0
G23	SMB_ADDR24	add_24	A21	SMB_nBLS1	bls_n_1
H23	SMB_ADDR25	add_25	B21	SMB_nBLS2	bls_n_2
J23	SMB_ADDR26	add_26	C21	SMB_nBLS3	bls_n_3
A19	SMB_ALEN	1'b0	C19	SMB_nCEN	1'b1
B19	SMB_CLEN	1'b0	E21	SMB_nCS0	1'b1
K21	SMB_CLKI	fbclk_in_0	F21	SMB_nCS1	1'b1
K19	SMB_CLKO	SMCLK	G21	SMB_nCS2	1'b1
H20	SMB_CRE	cre	H21	SMB_nCS3	cs_n_1
A30	SMB_DATA0	data_0	J21	SMB_nCS4	1'b1
B30	SMB_DATA1	data_1	A20	SMB_nCS5	1'b1
C30	SMB_DATA2	data_2	B20	SMB_nCS6	1'b1
D30	SMB_DATA3	data_3	C20	SMB_nCS7	cs_n_0
E30	SMB_DATA4	data_4	D20	SMB_nOE	oe_n
F30	SMB_DATA5	data_5	D19	SMB_nREN	1'b1
G30	SMB_DATA6	data_6	F19	SMB_nREQ	1'b0
H30	SMB_DATA7	data_7	E20	SMB_nWAIT	Wait
J30	SMB_DATA8	data_8	D21	SMB_nWE	we_n
K30	SMB_DATA9	data_9	E19	SMB_nWEN	1'b1
A29	SMB_DATA10	data_10			

Table 5 Static Memory bus pin allocation

4.5.3 Interrupts

HDRY pin	SB bus	Signal
J11	SB_INT[0]	clcdintr
K11	SB_INT[1]	smc_int
A10	SB_INT[2]	dma_int[0]
B10	SB_INT[3]	dma_int[1]

Table 6 Interrupts outputs pin allocation

5 Programmer's model

The example design for a V2F-1XV7 provides

- SCC memory mapped registers,
- FPGA SRAM,
- Static memory bus (SMB) to V2M-P1 motherboard
- Multimedia bus (MMB) to V2M-P1 motherboard
- AXI master bus to V2F-1XV7 daughterboard slave port.

5.1 Example AXI memory map

Memory Start	Memory End	Size	Bus	AN307
0x0000_0000	0x3FFF_FFFF	1 GB	N/C	Internal to V2P_CA15x2_A7x3
0x4000_0000	0x5FFF_FFFF	512 MB	AXI	External SODIMM
0x6000_0000	0x6FFF_FFFF	256 MB	AXI	AXIM (spare)
0x7000_0000	0x7000_FFFF	64 KB	APB	SCC
0x7001_0000	0x7001_FFFF	64 KB	APB	PL352 config
0x7002_0000	0x7002_FFFF	64 KB	-	Reserved
0x7003_0000	0x7003_FFFF	64 KB	AXI	PL111 config
0x7004_0000	0x7004_FFFF	64 KB	-	Reserved
0x7005_0000	0x7005_FFFF	64 KB	APB	SBCon
0x7006_0000	0x700F_FFFF	640 KB	-	Reserved
0x7010_0000	0x701F_FFFF	1MB	APB	NIC control
0x7020_0000	0x72FF_FFFF	46 MB	-	Reserved
0x7300_0000	0x73FF_FFFF	16 MB	AXI	SRAM
0x7400_0000	0x77FF_FFFF	64 MB	SMB	MB Peripherals (nCS3)
0x7800_0000	0x7BFF_FFFF	64 KB	SMB	MB Video RAM (nCS2)
0x7C00_0000	0x7C00_FFFF	64 KB	-	Reserved
0x7C01_0000	0x7C01_FFFF	64 KB	APB	PL330 config
0x7C02_0000	0x7FFF_FFFF	~64 MB	-	Reserved
0x8000_0000	0xFFFF_FFFF	2 GB	N/C	Internal to V2P_CA15x2_A7x3

Table 7 Memory Map

5.2 SCC registers

Table 8 shows the location of the SCC registers in the example design. The addresses shown are on APB bus offsets from the SCC base address 0x7000_0000.

Offset address	Name	Reset value	SIF Type	APB Type	Size	Function
0x000	SCC_USER0	0xXXXXXXXX	R/W	R/W	32	R/W register
0x004	SCC_USER1	0xXXXXXXXX	R/W	R/W	32	R/W register
0x100	SCC_DLLLOCK	0xFFXX000X	R/W	R	32	DLL locked
0x104	SCC_LED	0x0000000F	R	R/W	8	User LEDs control register
0x108	SCC_SW	0x000000XX	R/W	R	8	User Switches register
0xFF8	SCC_AID	0xXXXX0302	R/W	R	32	Auxiliary ID
0xFFC	SCC_ID	0x41X0307X	R/W	R	32	System ID

Table 8 Serial Configuration Control registers

5.2.1 SCC_USERx registers

The registers SCC_USER0 and SCC_USER1 (at offset 0x000-0x004) are general purpose user registers initialized during power up sequence by values from the daughter board configuration file.

In an existing AN307 build these registers can be used for any purpose by software.

Bits	Name	Access	Function	Default
[31:0]	SCC_USERx[31:0]	Read/write	General purpose registers configured during power up from configuration file	hXXXXXXXX

Table 9 SCC_USERx bit pattern

5.2.2 DLL lock register

The lock register SCC_DLLLOCK (at offset 0x100) indicated if all DLLs in system have been locked.

Bits	Name	Access	Function	Default
[31:24]	DLL LOCK MASK[7:0]	Read	These bits indicate if the DLL locked is masked.	8'b11111111
[23:16]	DLL LOCK[7:0]	Read	These bits indicate if the DLLs are locked or unlocked: b0 = unlocked b1 = locked	8'b111xxxxx
[15:1]	Reserved	Reserved	Reserved	15'b0
[0]	LOCKED	Read	This bit indicates if all enabled DLLs are locked: b0 = unlocked b1 = locked	1'bx

Table 10 SCC_DLLLOCK bit pattern

5.2.3 User LEDs control register

The SCC_LED register (at offset 0x104) controls the 8 of user LEDs on the V2F-1XV7 daughterboard.

Writing the value b11111111 will light all 8 LEDs. LEDs can be lit individually, for example writing b00000011 will light only the LED0 and LED1.

Bits	Name	Access	Function	Default
[31:8]	Reserved	Read	Reserved	hXXXXXX
[7:0]	LED[7:0]	Read/Write	These bits control LEDs	h0f

Table 11 SCC_LED bit pattern

5.2.4 User switches register

The SCC_SW register (at offset 0x108) indicates state of the 8 of user switches on the V2F-1XV7 daughterboard.

Bits	Name	Access	Function	Default
[31:8]	Reserved	Read	Reserved	hXXXXXX
[7:0]	SW[7:0]	Read	These bits indicate state of user switches	hXX

Table 12 SCC_SW bit pattern

5.2.5 SCC_AID register

The SCC_AID register (at offset 0xff8) includes the 16-bit SCC registers description.

Bits	Name	Access	Function	Default
[31:24]	Build	Read	FPGA build number	hXX
[23:16]	Reserved	Read	Reserved	hXX
[15:11]	Reserved	Read	Reserved	5'b00000
[10]	SWREGP	Read	These bits indicate if SCC_SW register have been implemented	1'b1
[9]	LEDREGP	Read	These bits indicate if SCC_LED register have been implemented	1'b1
[8]	DLLREGP	Read	These bits indicate if DLL lock register have been implemented	1'b1
[7:0]	USERREGN	Read	These bits indicate number of SCC_USERx register	h02

Table 13 SCC_AID bit pattern

5.2.6 SCC_ID registers

The SCC_ID register (at offset 0xffc) includes the 32-bit AN identification.

Bits	Name	Access	Function	Default
[31:24]	Implementer	Read	Implementer ID	h41
[23:20]	Variant	Read	Variant Number	hX
[19:16]	Architecture	Read	Architecture. Have to be 0x0 for Application Notes.	h00
[15:4]	AN	Read	Application Note number	h307
[3:0]	Revision	Read	Revision number	hX

Table 14 SCC_ID bit pattern

5.3 Reserved and undefined memory

If reserved memory is accessed, it will be caught by the AXI bus matrix and return a decode error ('DECERR') which generates a data abort. The only exception to this is the APB subsystem, which has 13 spare PSEL signals for expansion. If this address range is accessed the AXIToAPB bridge will return an 'OKAY' response.

6 RTL

Only the top level and SCC RTL files are included. AXI components are supplied as netlists. However, Amba Designer XML configuration files are provided for those components to allow them to be rebuilt.

Example files are provided to allow building the system with Xilinx tools.

6.1 Directory structure

The application note has several directories. They are:

- boardfiles: The files are required to program the design into a V2F-1XV7.
- docs: Related documents including this document.
- logical: Verilog RTL for this design.
- physical: Synthesis and place and route (P&R) scripts and builds for target board.
- software: ARM code to exercise the AN307 application note.

6.2 Logical

The logical directory contains all the verilog supplied with this application note. It also contains Amba Designer XML configuration files which can be used to regenerate verilog for ARM PrimeCell used in the example design.

The top level for this system is in V2F_1XV7_wrapper.

6.3 Physical

The physical directory contains pre-synthesised components. The function of each block is shown earlier in **3.2 Module functionality**.

Each PrimeCell or other large IP block has its own directory (for example pl111_clcd_r0p2).

For tool revision used to build App Note please refer to **/doc/readme.txt** file.

6.4 Building the App Note using Unix

The Xilinx DDR controller is not supplied and needs to be rebuilt if modifications are made to the design. For V2F-1XV7 revision C, the DDR controller can be rebuilt from the IP catalog in Vivado 2012.04. For V2F-1XV7 revision B, please contact ARM support for assistance with rebuilding the DDR controller.

The following table details the settings required by CoreGen.

Refer to the run_vivado_201*n.m*.tcl[1] script in AN307/physical/an307_toplevel/xilinx_rev*x*/scripts[2] directory for the path to the .edf.

[1] Where *n.m* is Xilinx Vivado version and revision.
[2] Where *x* is the V2F-1XV7 daughterboard revision.

DDR controller for revision C board - Vivado 2012.04 (MIG 1.8)	
CORE Generator Options:	
Target Device	xc7v2000t-flg1925
Speed Grade	-1
HDL	verilog
Synthesis Tool	VIVADO
MIG Output Options:	
Module Name	mig_7series_v1_8_a_0
No of Controllers	1
FPGA Options:	
System Clock Type	Single-Ended
Reference Clock Type	Differential
Debug Port	OFF
Internal Vref	disabled
IO Power Reduction	ON
XADC instantiation in MIG	Enabled
Extended FPGA Options:	
DCI for DQ,DQS/DQS#,DM	enabled
Internal Termination (HR Banks)	50 Ohms
Controller Options	
Memory	DDR3_SDRAM
Interface	AXI
Design Clock Frequency	2500 ps (400.0 MHz)
Phy to Controller Clock Ratio	4:1
Input Clock Period	10000 ps
VCC_AUX IO	1.8V
Memory Type	SODIMMs
Memory Part	MT8KTF51264HZ-1G6
Memory Voltage	1.5V
Data Width	128
ECC	Disabled
Data Mask	Enabled
ORDERING	Strict
Memory Options:	
Burst Length (MR0[1:0])	8 - Fixed
Read Burst Type (MR0[3])	Sequential
CAS Latency (MR0[6:4])	7
Output Drive Strength (MR1[5,1])	RZQ/7
Rtt_NOM - ODT (MR1[9,6,2])	RZQ/4
Rtt_WR - Dynamic ODT (MR2[10:9])	Dynamic ODT off
Memory Address Mapping	BANK_ROW_COLUMN

Bank Selections:	
Reference_Clock:	
SignalName: clk_ref_p/n	PadLocation: AT15/AU15(CC_P/N) Bank: 32
System_Clock:	
SignalName: sys_clk_i	PadLocation: AU12/(MRCC_P) Bank: 31
System_Control:	
SignalName: sys_rst	PadLocation: No connect
SignalName: init_calib_complete	PadLocation: No connect
SignalName: tg_compare_error	PadLocation: No connect

Table 15 Xilinx DDR Coregen settings

To build the App Note using Unix run the make_vivado.scr batch file in the following directory: /physical/an307_toplevel/xilinx_rev*x*/scripts[1].

This synthesizes the design and runs place and route on the design pulling in pre synthesized components.

A programmable bit file is generated under /physical/an307_toplevel/xilinx_rev*x*/netlist[1] called a307r*y*p0.bit[2]

6.5 Using the new bitfile

To use the new a307r*y*p0.bit[2] bit file, the file must be copied to the SITE2/HBI0247x/AN307[1] on V2M-P1 USB Flash drive directory.

The board.txt text file located at /SITE2/HBI0247x/[1] in V2M-P1 USB Flash disk have to be edited and APPNOTE field must be modified to AN307/a307r*y*p0.txt[2].

[1] Where *x* is the V2F-1XV7 daughterboard revision.
[2] Where *y* is AN307 design revision.

7 Example software

Example software (AN307_test) is provided to verify the example design and the V2F-1XV7 daughterboard hardware.

C and assembly language source files are included.

After the Versatile Express system is configured you can upload and execute the example software using a debugger connected to the V2P-CA15-A7 processor board.

The example code communicates with the user via the debugger's console window. It operates as follows:

1. Reads the identification register to ensure that the software is being executed on the correct system.
2. Initialises SMC.
3. Detects presence of DDR3 SODIMM (serial presence-detect (SPD) EEPROM).
4. Tests the FPGA SRAM for sequential write/read and random write/read for word, half-word and byte accesses.
5. Tests the Static Memory bus by writing reading to V2M-P1 Motherboard VRAM.
6. Tests the DDR3 SDRAM for sequential write/read and random write/read for word, half-word and byte accesses.
7. Tests the DMA controller and interrupt signal from DMA.
8. Tests the CLCD and Multimedia Bus.

Note that to test VGA, a DVI-D or HDMI monitor that supports VGA resolution must be connected to the V2M-P1 DVI-I connector.

8 I/O Timing Requirements

All of these specific timing requirements refer to the r1p0 revision of the AN307. All units are in nano-seconds “ns” and have been rounded to a worst case value.

Signals with setup, hold and clock to data values are bidirectional signals or have been grouped by function in the table.

8.1 Default, minimum and maximum operating frequencies

Clock source	Clock signal	Clock domain	Default Freq	Min Freq	Max Freq
OSC0	ACLK	AN307 AXI System Bus	80MHz	30MHz	100MHz
OSC1	CLCDCLK	CLCD	23.75MHz	23MHz	33MHz
OSC2 (rev B)	DDRSYSCLK	DDR System clock	77.6MHz	77.6MHz	77.6MHz
OSC2 (rev C)	DDRSYSCLK	DDR System clock	100MHz	100MHz	100MHz
OSC3	Reserved	Reserved	50MHz	50MHz	50MHz
OSC4	SMCLK/SMCLKIN	Static Memory Bus	40MHz	32MHz	50MHz
OSC5	Reserved	Reserved	50MHz	50MHz	50MHz
OSC6	DDRREF	DDR REF CLK	200MHz	200MHz	200MHz

Table 16 Default and maximum operating frequencies

8.2 AXI timing requirements

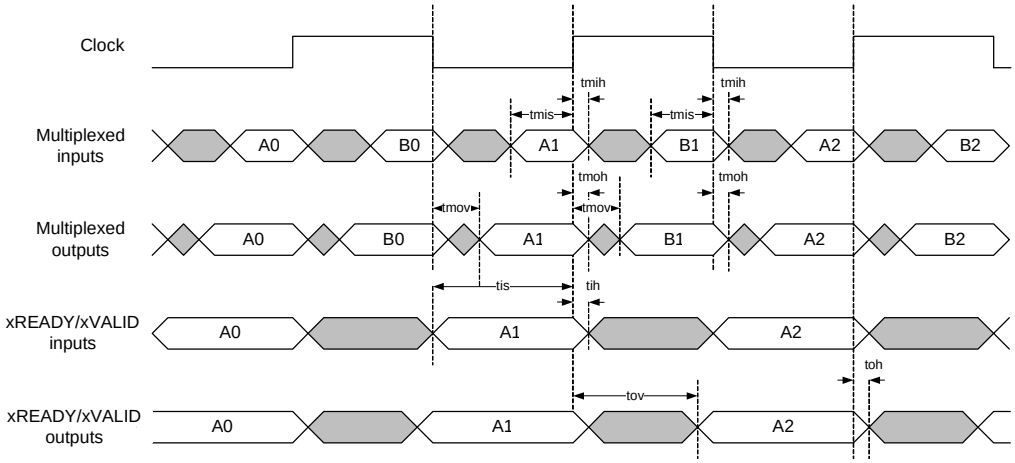

Figure 8-1 AXI timing requirements

Dir	Name	Description	Value
Inputs	tmis [ns]	Max multiplexed inputs setup to clock	7.0
	tmih [ns]	Max multiplexed inputs hold to clock	2.75
	tis [ns]	Max AWREADY, ARREADY, WREADY, RVALID, BVALID inputs setup to clock	20.0
	tih [ns]	Max AWREADY, ARREADY, WREADY, RVALID, BVALID input hold to clock	2.75
Outputs	tmov [ns]	Max clock to multiplexed data valid	7.5
	tmoh [ns]	Min clock to multiplexed data invalid	2.75
	tov [ns]	Max clock to WREADY, BREADY, AWVALID, ARVALID, WVALID outputs valid	20.0
	toh [ns]	Min clock to WREADY, BREADY, AWVALID, ARVALID, WVALID outputs invalid	2.75

Table 17 AXI Slave Input/Output timing to EMM_ACLK clock input

8.3 MMB signals timing.

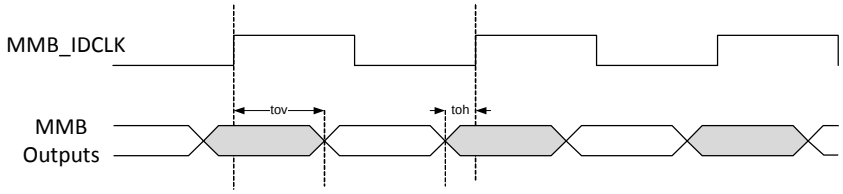

Figure 8-2 MMB Outputs Timing

Dir	Name	Description	Value
Outputs	tov [ns]	Max rising edge of MMB_IDCLK to data valid	9
	toh [ns]	Min rising edge of MMB_IDCLK to data invalid	-1.0

Table 18 MMB Output timing

8.4 SMB signals timing.

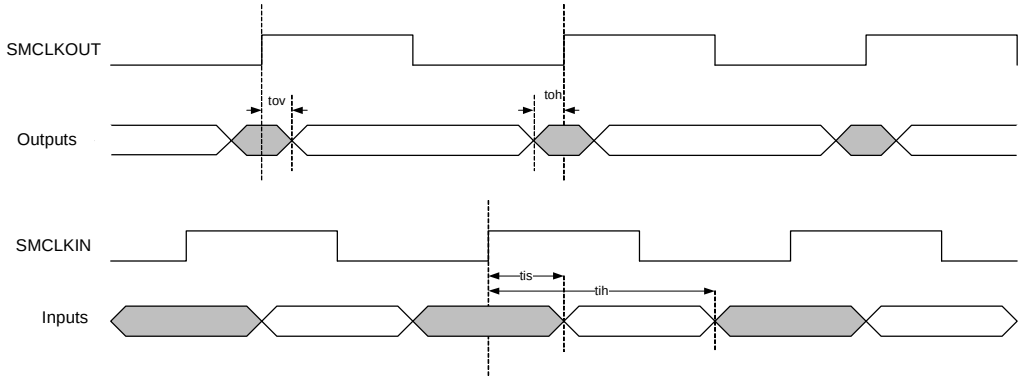

Figure 8-3 SMB Inputs and Outputs Timing

Dir	Name	Description	Value
Inputs	tis [ns]	Max inputs setup to rising edge of SMCLKIN	7.5
	tih [ns]	Max inputs hold to rising edge of SMCLKIN	12.5
Outputs	tov [ns]	Max rising edge of SMCLKOUT to data valid	16
	toh [ns]	Min rising edge of SMCLKOUT to data invalid	-2.5

Table 19 SMB Input/Output timing