DSP/BIOS PALSYS VLYNQ Device Driver

Release version 1.10.02

Release Notes

November 15, 2007

The product release notes in this document are for DSP/BIOS PALSYS VLYNQ Device Driver, which is part of the PAL SYS package.

TABLE OF CONTENTS

1 New In this release

None.

2 System Requirements

Refer system level release notes for tools and BIOS versions.

3 Installation and Usage

1. Install DSP/BIOS PSP package as per instructions provided along with the package.
2. PAL SYS VLYNQ Device driver sources are available in <root>/pal_sys/vlynq folder.
3. Build the vlynq project file in build directory to build the debug/release library.
4. Sample application / test code is provided in system/<SOC_NAME>/bios/<PLATFORM_NAME>/Vlynq package.

Note 1: To use VLYNQ module on evmDM648 and evm6452 platforms, pin muxing has to be enabled for the daughter card using the I2C bus. For this, one should write to the slave address 0x70u a value of 0x28u to the offset 0x06u in Alpha and Beta versions of EVMs. For other versions of EVMs, please refer the firmware documentation available with EVM. One can modify either the I2C sample application (just as a generic I2C driver software) or the Gel file released by EVM manufacturers, for this configuration. No such pin muxing is required on the evm6424 and evmDM6437 EVMs.

Note 2: Following jumper settings should be used in the DM648/C6452 Vlynq daughter card:

P1: Master, P5: Slave
Jumper settings for Master: Short P2.4 - P2.6 and P2.12 – P2.14
Jumper settings for Slave: Short P2.2 - P2.4 and P2.10 – P2.12

For more information on the jumper settings, refer to the DM648 Daughter Card Manual.

Note 3: Following jumper settings should be used in the DM6437/C6424 Vlynq Mini PCI daughter card:

P2: Master, P3: Slave
Jumper settings for Master: Short the following jumpers:
a) J8: 2-3
b) J7: 1-2
c) J6: 1-2
d) J5: 1-2

Jumper settings for Slave: Short the following jumpers:

a) J8: 2-3
b) J7: 1-2
c) J6: 2-3
d) J5: 2-3

For more information on the jumper settings, refer to the DM6437/C6424 Daughter Card Manual.

Note 4: The Vlynq release follows the PAL SYS architecture.

4 Uninstallation

1. Un-install the PAL SYS package as per instructions provided with the package.

5 Known Issues

1	**Chaining is not tested in this release.**
Workaround None	

2	**Backward compatibility to 1.x version is not tested.**
Workaround None	

6 Hardware limitations for DM648/C6452

1	**Memory offset of 0x100u is observed.**
VLYNQ sample app tries to access slave's (DM6437) memory from master side (DM648). It was observed that the actual write is happening on the slave side at an address of CONF_ADDRESS + 0x100u, where CONF_ADDRESS is the value which is written in the Remote Rx Address Map Offset. For e.g., if one programs the Remote Rx Address Map Offset1 register as 0x84000000 from the master side, the actual write at the slave side will happen at 0x84000100. This is because of the hardware problem or hardware design only. **Workaround** None	

2	**Link not stable for DM648-DM648 connection.**
When DM648 is connected with another DM648 board, with one working as Master and the other one working as slave, the link establishment is unstable (once in few trials). We suspect this to be hardware problem. When DM648 connected with DM6437, it worked fine for both the configurations: Master as well as slave. **Workaround** None	

3	VLYNQ clock divisor can not have all the possible values
DM648 running at XXX MHz will supply XXX/6 MHz clock to the Vlynq IP. Vlynq IP alos has one divisor value which can further reduce this clock. Vlynq IP spec recommends maximum Vlynq clock to be 100 MHz. So the Vlynq divisor value should be chosen in such a way such that only permissible clock drives the Vlynq IP. For e.g. DM648 running at 891 MHz will give 891/6 = ~150 MHz to the Vlynq IP. After this, one can choose the Vlynq divisor values from 2 to 7 (both inclusive) so that the Vlynq clock will range from 150/7 (=21MHz) to 150/2 (=75MHz), which has been tested. Since the recommended clock value is met, Vlynq works fine. But if the user tries the divisor as 1, then the clock will become 150/1 = 150 MHz, which can cause unpredictable results and possibly hang. User should do a hardware reset then. So user should always use the 'valid' clock divisors only. Also, please note that depending upon the signal integrity/Vlynq cable quality, one may face communication/link problems at higher frequencies (i.e. lower clock divisors). **Workaround** None	

7 Known Issues for DM6437/C6424

1	Mini-PCI connectors that are used for VLYNQ connection don't support change in the direction of clock dynamically.
Workaround None	

2	PAL_VLYNQ_IOCTL_PREP_LINK_DOWN and PAL_VLYNQ_IOCTL _PREP_LINK_UP IOCTL commands can only be used to tear down and re-setup the link from LOCAL vlynq side.
Workaround None	

8 Known Issues for DM648/C6452

NA

9 Revision history

Date	Author	Comments	Version
December 1, 2006	Rinkal Shah	BFT release 0.3.0	1.1
January 18, 2007	JP	BFT release 0.3.1	1.2
March 24, 2007	Anuj Aggarwal	BFT release 0.4.0	1.3
June 15, 2007	Anuj Aggarwal	Updated for release 1.10.00.00 (formerly release 0.6)	1.4
July 12, 2007	Anuj Aggarwal	Updated for release 1.10.00.03	1.5
November 15, 2007	Nagarjuna K	Updated for DM6437/C6424/DM648/C6452 package	1.6
May 28, 2008	M Sriram	Updated for release	1.7