VME-CHIP

of

PSB-9U_V2-card

H. Bergauer, K. Kastner, B. Neuherz, M. Padrta, T. Schreiner, A. Taurok

Feb-07

Version 0x1009

Table of contents

1 Introduction

The VME-CHIP-PSB works with the VME64x chip PSB as controller for the VME-bus of the PSB-9U_V2-card. There are VME-registers on it as the Registers for Programmable-chips-configuration, General registers, Chip ID / version registers and JTAG registers. The VME-accesses to the PSB-chip are made via this chip too.

2 VME chip of PSB-9U_V2-card

2.1 Versionshistory

- V1000: first design. Error at RESET_MODE. **DO NOT USE!!** (HB110805).
- V1001: based on V1000, but RESET_MODE error corrected. **DO NOT USE!!** (HB110805).
- V1002: based on V1001, but PSB/MEM access (ENPSB, ENPSBMEM) made with DSSYNC for read and write accesses. (HB120805).
- V1003: based on V1002, but write/read for general-register and configuration-register implemented. (HB160805).
- V1004: based on V1003, but lines ASCYC, ASSYNC, ASPULS and D08_E from VME64x-chip represent the CARD_NR[3:0] – from jumpers S27-S24 on board.
  DSSYNC with two clock delays (ADDR_DEC_PSB_V2_2 and output-FF) used for read/write of PSB-chip.
  [VIEWDRAW: library P:\Lab3Lib\..\vme_chip_lib used] (HB121005).
- V1005: based on V1004, but DTACK_EXT and BERR_EXT are used as negative active signals to VME64x-chip (because at power-up configuration of VME64X-CHIP is faster than configuration of VME-CHIP and therefore wrong DTACK and BERR signals are generated after configuration, which causes LEDs=”on” of CAEN-controller). (HB201005)
  Because of changes in the VME64x-chip version, the CARD_NR[3:0] comes from jumpers S31-S28 on board (HB071105).
- V1006: based on V1005, but error-counters for serial-link-chips implemented. (HB071105).
- V1007: based on V1006, but JTAG_CTRL V1.5 implemented, because of Quartus 5.1 and EN_JTAG is delayed to have proper setup-time for addresses, because of syncronous version of VME64x-chip. (HB050106).
- V1008: based on V1007, but JTAG_CTRL V1.7 (with JTAGController V1.11) implemented, to get proper JTAGoverVME access. Addressdecoder changed to ADDR_DEC_PSB_V2 V2.6 and signals DTACK and BERR are send as pos. active signals to VME64x-chip (HB041206). **DO NOT USE!!**
- **V1009:** based on V1008, but ADDR_DEC_ PSB_V2 V2.7 used to combine EN_JTAG with SELECT (HB121206).

2.2 Hardware

The VME-CHIP-PSB is an Altera EP1K100QC208-3.

2.3 Firmware

chip_id: `0x0001Cn21` `(n = CARD_NR from jumpers S31 – S28)`
version: `0x00001009`

2.4 Features of the VME-CHIP-PSB (V1009)

- Cardnumber from VME64x-chip (from jumpers S31-S28 on board).
- Register for chip_ID and version.

- Register for commands and status (see VME-CHIP-PSB address-table).
- Register for modes of serial-link-chips.
- Register to enable technical- and/or totem-trigger-bits.
- VME-access to PSB-chip with DSSYNC for read and write (two clock delays).
- "JTAG over VME" for configuration of FPGAs.
- Configuration of PSB-chip via VME (changes in hardware of mezz957 necessary – not implemented in software).
- DTACK and BERR generation from PSB-chip.
- Error-counters for serial-link-chips.

2.5 VME access

2.5.1 Base address

Base address of all GT-slaves is encoded on A31-A25 (A24 not used), because of address space of GTL-6U-card. See definition in VME64x-chip for PSB-9U_V2.

2.5.2 AM and datatransfer

AM=0x0D and 0x09 „extended data access" - for single access.
AM=0x0F and 0x0B „extended block transfer" - for block transfer access.
D16 „word access" - for all accesses.
See definitions in VME64x-chip for PSB-9U_V2.

2.6 Chip selection on PSB-9U_V2-card

With the VME addresses A23-A20 the chip selection is done on the PSB-9U_V2-card.

A23	A22	A21	A20	Chip-name
0	0	0	0	VME-CHIP-PSB
0	0	0	1	PSB-chip-registers
0	0	1	0	PSB-chip-memories

2.7 VME-CHIP-PSB register

Registeraddresses:			
A31..A24	A23..A20	A19..A05	A06..A01,(00)
8 bits		13bits	6bits
Base address	0000	XXXX	Registers

2.7.1 VME-CHIP-PSB address-table

The address-table lists the address-offset which has to be combined with the base-address of the card.

A23-A00 mask w/r => Register-name

Register for Programmable-chips-configuration:

```
0x000000 0x00000001 w/r => CMD_ENPROG-register
0x000002 0x00000001 w/r => CMD_NPROG-register
0x000004 0x00000001 w/r => CMD_INIT-register
0x000006 0x00000001   r => STAT_INIT-register
0x000008 0x00000001   r => STAT_DONE-register
0x00000A 0x00000001 w   => Configuration register PSB-chip
```

General pulse registers:

```
0x000010 0x0000000F w   => Command pulse register
0x000012 0x0000000E   r => Status pulse register
```

General registers:

```
0x000014 0x0000001F w/r => Command register
0x000016 0x00000025   r => Status register
```

Chip ID and version registers:

```
0x000020 0x000000FF   r => chip_id_register_3
0x000022 0x000000FF   r => chip_id_register_2
0x000024 0x000000FF   r => chip_id_register_1
0x000026 0x000000FF   r => chip_id_register_0
0x000028 0x000000FF   r => version_register_3
0x00002A 0x000000FF   r => version_register_2
0x00002C 0x000000FF   r => version_register_1
0x00002E 0x000000FF   r => version_register_0
```

JTAG registers:

```
0x000030 0x000000FF w/r => tdo_register
0x000032 0x000000FF w/r => tdi_register
0x000034 0x000000FF w/r => tms0_register
0x000036 0x000000FF w/r => tms1_register
0x000038 0x000000FF w/r => cnt32_register
0x00003A 0x000000FF w/r => mode0_register
0x00003C 0x000000FF w/r => mode1_register
0x00003E 0x000000FF w/r => mode2_register
```

Serial link mode registers:

```
0x000040 0x0000FFFF w/r => SERLINK0-register
0x000042 0x0000FFFF w/r => SERLINK1-register
0x000044 0x000000FF w/r => SERLINK2-register
0x000046 0x000000FF   r => LOCKED-register
```

EN_TTIN Register to enable Technical or Totem Trigger bits:

```
0x000050 0x0000FFFF w/r => EN_TTIN -register
```

Error-counter Register for serial-link-chips (SERLINK):

```
0x000060 0x000000FF    r  => ERR_CNT_SERLINK_0-register
0x000062 0x000000FF    r  => ERR_CNT_SERLINK_1-register
0x000064 0x000000FF    r  => ERR_CNT_SERLINK_2-register
0x000066 0x000000FF    r  => ERR_CNT_SERLINK_3-register
0x000068 0x000000FF    r  => ERR_CNT_SERLINK_4-register
0x00006A 0x000000FF    r  => ERR_CNT_SERLINK_5-register
0x00006C 0x000000FF    r  => ERR_CNT_SERLINK_6-register
0x00006E 0x000000FF    r  => ERR_CNT_SERLINK_7-register
```

Access to/from PSB-chip:

```
0x1XXXXX                  => see PSB-chip-registers
0x2XXXXX                  => see PSB-chip-memories
```

2.7.2 Register for Programmable-chips-configuration

The PSB-chip (Virtex-II) is configurable by configuration device and by VMEbus instructions. The selection is made by jumpers. The register-definition for configuration by VMEbus shall be a standard. See P:\Lab3Lib\Altera\Lab3_altera\sch\xilinx_conf.

Register names	**D7..D1**	**D0**
CMD_ENPROG	-	ENPROG_PSB
CMD_NPROG	-	NPROG_PSB
CMD_INIT	-	INIT_PSB
STAT_INIT	-	INIT_PSB
STAT_DONE	-	DONE_PSB
CONF_PSB	-	configuration data

2.7.2.1 CMD_ENPROG-register

`0x000000` => CMD_ENPROG-register (write/read)

Bit 0 of the CMD_ENPROG-register allows sending the configuration bits via VME-bus to the PSB-chip.

2.7.2.2 CMD_NPROG-register

`0x000002` => CMD_NPROG-register (write/read)

Data-bit 0 = 1 of this register set the NPROG-signal of PSB-chip active. Then it should be reset to '0'. Then the PSB-chip enters into the configuration procedure. The FPGA either waits for configuration data (slave mode) sent via VME or starts to read configuration bits from a serial PROM (master mode).

2.7.2.3 CMD_INIT-register

`0x000004` => CMD_INIT-register (write/read)

Data-bit 0 = 1 of this register set the NINIT-signal of PSB-chip active.

2.7.2.4 STAT_INIT-register

`0x000006` => STAT_INIT-register (read)

Read the status of the NINIT-signal of PSB-chip (data-bit 0)

2.7.2.5 STAT_DONE-register

`0x000008` => STAT_DONE-register (read)

Read the status of the DONE-signal of PSB-chip (data-bit 0). After a successful configuration the PSB-chip sets DONE = 1.

2.7.2.6 Configuration register PSB-chip

`0x00000A` => Configuration-register PSB-chip (write)

The register is used to load the configuration bits into the PSB-chip (Virtex-II).
A write access to this register generates a CCLK and sends the data-bit 0 as DIN-signal to the PSB-chip, if the CMD_ENPROG-register bit has been set before. The VME accesses are repeated until the last bit has been loaded into the PSB-chip.

2.7.3 General pulse registers

Register names	**D3**	**D2**	**D1**	**D0**
Command_Pulse _Reg	SET_RUNNING (pulse)	RESET_PSB (pulse*)	RES_DCM_ PSB (pulse)	PWRDWN_ PSB (pulse)
Status_Pulse_Reg	RUNNING	LOCKED_ LED	CLK_LOCKED _PSB	not used

*) also generated by RESET_MODE

Register names	**D7**	**D6**	**D5**	**D4**
Command_Pulse _Reg	not used	not used	not used	not used
Status_Pulse_Reg	not used	not used	not used	not used

2.7.3.1 Command pulse register

`0x000010` => Command-pulse-register (write)

D0: **PWRDWN_PSB = 1** sends a low active pulse to the PSB-chip setting it into power down mode. NPWRDWN_B is sent as an open drain signal from the VME-PSB-chip to the PSB-chip.
Remark from data sheet:
The power-down sequence enables a designer to set the device into a low-power, inactive state. The sequence is initiated by pulling the PWRDWN_B pin Low. To monitor power-down status, observe the PWRDWN_B pin. When asserted, power-down has completed. After a successful wake-up, the status pin de-asserts. While powered down, the only active pins are the PWRDWN_B and DONE. All inputs are off and all outputs are 3-stated. While in the POWERDOWN state, the Power On Reset (POR) circuit is still active, but it does not reset the device if V_{CCINT} , V_{CCO} , or V_{CCAUX} falls below its minimum value. The POR circuit waits until the PWRDWN_B pin is released before resetting the device. Also, the PROG_B pin is not sampled while the device is in the POWERDOWN state. The PROG_B pin becomes active when the PWRDWN_B pin is released. Therefore, the device cannot be reset while in the POWERDOWN state. The wake-up sequence is the reverse of the power-down sequence.

D1: **RES_DCM_PSB = 1** sends a high active pulse to the PSB-chip, to forces the DCM module to lock.

D2: **RESET_PSB = 1** sends a high active pulse to the PSB-chip for reset activities. (RESET_MODE is another source for RESET_PSB.)

D3: **SET_RUNNING = 1** sends a high active pulse to set board in RUNNING mode.

2.7.3.2 Status pulse register

`0x000012` => **Status-pulse-register (read)**

D0: not used.

D1: **CLK_LOCKED_PSB = 1** indicates, that the DCM module of the PSB chip are locked to the 40 MHz clock.
This status bit has to be checked immediately after the configuration of the PSB chip and before any other actions. If the chips do not lock then either the clock signal from the TIM board or the on-board oscillator are bad.

D2: **LOCKED_LED** is the status of an AND of all LOCKED-signals.
The LOCKED_LED signal will illuminate the front-panel LED only if all enabled Serial Receiver Chips are locked to the clock of incoming serial data and if the PSB Chip has locked to the system clock.

D3: **RUNNING = 1** board is active.
If RUNNING = 0, send a SET_RUNNING command via VME.

2.7.4 General registers

Register names	**D3**	**D2**	**D1**	**D0**
Command_Reg	V_SEL_ CABLES	VME_CONF	EN_ROBUS	EN_CHLINK
Status_Reg	not used	STATUS_SEL_ VME	not used	EN_CHLINK

Register names	**D7**	**D6**	**D5**	**D4**
Command_Reg	not used	not used	not used	V_SEL_ BACKPL
Status_Reg	not used	not used	JTAG_JUMPER	not used

2.7.4.1 Command register

`0x000014` => **Command-register (write/read)**

D0: **EN_CHLINK = 1** enables channel-link-chips, if CLK_LOCKED_PSB is active (signal NEN_CHLINK active).

D1: **EN_ROBUS = 1** enables ROBUS (signal NEN_ROBUS active).

D2: **VME_CONF = 1** enables configuration of PSB-chip via VME and switches external mux from PROM to VME.

D3: **V_SEL_CABLES = 1** switches JTAG-chains to cables (MasterBlaster and Parallel-Cable-IV).

D4: **V_SEL_BACKPL = 1** switches JTAG-chains to backplane connection via SCANPSC110 (if V_SEL_CABLES = 0).

Truthtable for D4 and D3:

D4	**D3**	
0	0	JTAG-chains via VME
X	1	JTAG-chains via cables
1	0	JTAG-chains via backplane

2.7.4.2 Status register

`0x000016` => **Status-register (read)**

D0: **EN_CHLINK** is the inverted status of signal NEN_CHLINK.

D1: not used.

D2: **STATUS_SEL_VME = 1** indicates, that configuration of PSB-chip via VME is selected.
For configuration of PSB-chip via VME, set VME_CONF = 1 in the Command-register.

D3: not used.

D4: not used.

D5: **JTAG_JUMPER = 1** indicates, that SEL_CABLE_JTAG-jumper (JP50) is inserted. Therefore JTAG-chains are connected to cables (MasterBlaster and Parallel-Cable-IV).
For changing the sources of JTAG-chains, remove the jumper and make the selection with V_SEL_CABLES and V_SEL_BACKPL in the Command-register.

2.7.5 Chip_ID and version registers

2.7.5.1 Definitions

Chip_id_register and version_register have fixed values in the hardware. These registers have read access only.
The versions `0x00000000 - 0x00000FFF` are used for tests.
The versions `0x00001000 - 0xFFFFFFFF` are used for runs in CMS.

2.7.5.2 Settings

chip_id: `0x0001Cn21` `(n = CARD_NR from jumpers S31 – S28)`
version: `0x00001009`

2.7.5.3 Chip_ID and version registers addresses

`0x000020` => chip_id_register_3 (read)

D7	D6	D5	D4	D3	D2	D1	D0
chip_ID [31..24]							

`0x000022` => chip_id_register_2 (read)

D7	D6	D5	D4	D3	D2	D1	D0
chip_ID [23..16]							

`0x000024` => chip_id_register_1 (read)

D7	D6	D5	D4	D3	D2	D1	D0
chip_ID [15..08]							

`0x000026` => chip_id_register_0 (read)

D7	D6	D5	D4	D3	D2	D1	D0
chip_ID [07..00]							

`0x000028` => version_register_3 (read)

D7	D6	D5	D4	D3	D2	D1	D0
version [31..24]							

`0x00002A` => version_register_2 (read)

D7	D6	D5	D4	D3	D2	D1	D0
version [23..16]							

0x00002C => version_register_1 (read)

D7	D6	D5	D4	D3	D2	D1	D0
version [15..08]							

0x00002E => version_register_0 (read)

D7	D6	D5	D4	D3	D2	D1	D0
version [07..00]							

2.7.6 JTAG-registers

2.7.6.1 Definitions

JTAG registers are used to control JTAG-chains via VME-bus.
For details see JTAGController.vhd from Hannes Sakulin.

2.7.7 Serial link mode registers

Register names	**D15..D12**	**D11..D8**	**D7..D4**	**D3..D0**
SERLINK0	SEND_SYNC_ PATTERN [3..0]	LINE_ LOOPBACK [3..0]	TPWRDWN [3..0]	ENTR [3..0]
SERLINK1	RPWRDWN [7..0]		ENREC [7..0]	
SERLINK2	not used		LOCAL_LOOPBACK [7..0]	
LOCKED	not used		LOCKED [7..0]	

Remarks:
23.8.05 AT corrected in table above NTPWRDWN to TPWRDWN, NRPWRDWN to RPWRDWN.
Index [7..0] means DS92LV16 chip number = channel number.

SEND_SYNC_PATTERN = 1 => transmitter sends SYNC patterns so that a receiver can synchronize to the incoming data stream.
LINE_LOOPBACK = 1 => the serial received data are returned via the serial transmission line.
TPWRDWN = 1 => powers down the transmitter part of the chip (signal NTPWRDWN=0).
ENTR = 1 => enables the transmitter circuits of the chip.
LOCAL_LOOPBACK = 1 => returns parallel Transmit-data to parallel Receiver lines.
RPWRDWN = 1 => powers down the receiver part of the chip (signal NRPWRDWN=0).
ENREC = 1 => enables the receiver circuits of the DS92LV16 chip.

0x000040 => SERLINK0-register (write/read)
0x000042 => SERLINK1-register (write/read)
0x000044 => SERLINK2-register (write/read)
0x000046 => LOCKED-register (read)

2.7.8 EN_TTIN Register to enable Technical or Totem Trigger bits

Enables the LVDS receivers for Technical Trigger resp. Totem Trigger signals.
Disabled Receivers send bits 1111 = 'F'.

See also registers in PSB9U chip to switch between Parallel LVDS and Serial input channels.

Register names	**D15..D0**
EN_TTIN	EN_R[15..0]

EN_Rxx = 1 => enables parallel LVDS receivers for Parallel cable xx (0 = default).

0x000050 => EN_TTIN-register (write/read)

2.7.9 Error-counter Register for serial-link-chips (SERLINK)

The Error-counter Register for serial-link-chips contains the number of missed LOCKED-signals of the serial-link-chip. If there is an overflow in the counter, the value remains at 0xFF. Reading the register causes a clear on the counter.

Register names	**D7..D0**
ERR_CNT_SERLINK_x	Errors caused by missing LOCKED-signal of serial-link-chip

0x000060 => ERR_CNT_SERLINK_0-register (read)
0x000062 => ERR_CNT_SERLINK_1-register (read)
0x000064 => ERR_CNT_SERLINK_2-register (read)
0x000066 => ERR_CNT_SERLINK_3-register (read)
0x000068 => ERR_CNT_SERLINK_4-register (read)
0x00006A => ERR_CNT_SERLINK_5-register (read)
0x00006C => ERR_CNT_SERLINK_6-register (read)
0x00006E => ERR_CNT_SERLINK_7-register (read)

2.8 DTACK/BERR-generation

Writing to writeable registers and reading from readable registers generates a DTACK signal. Access to/from PSB-chip generates a DTACK or a BERR signal.