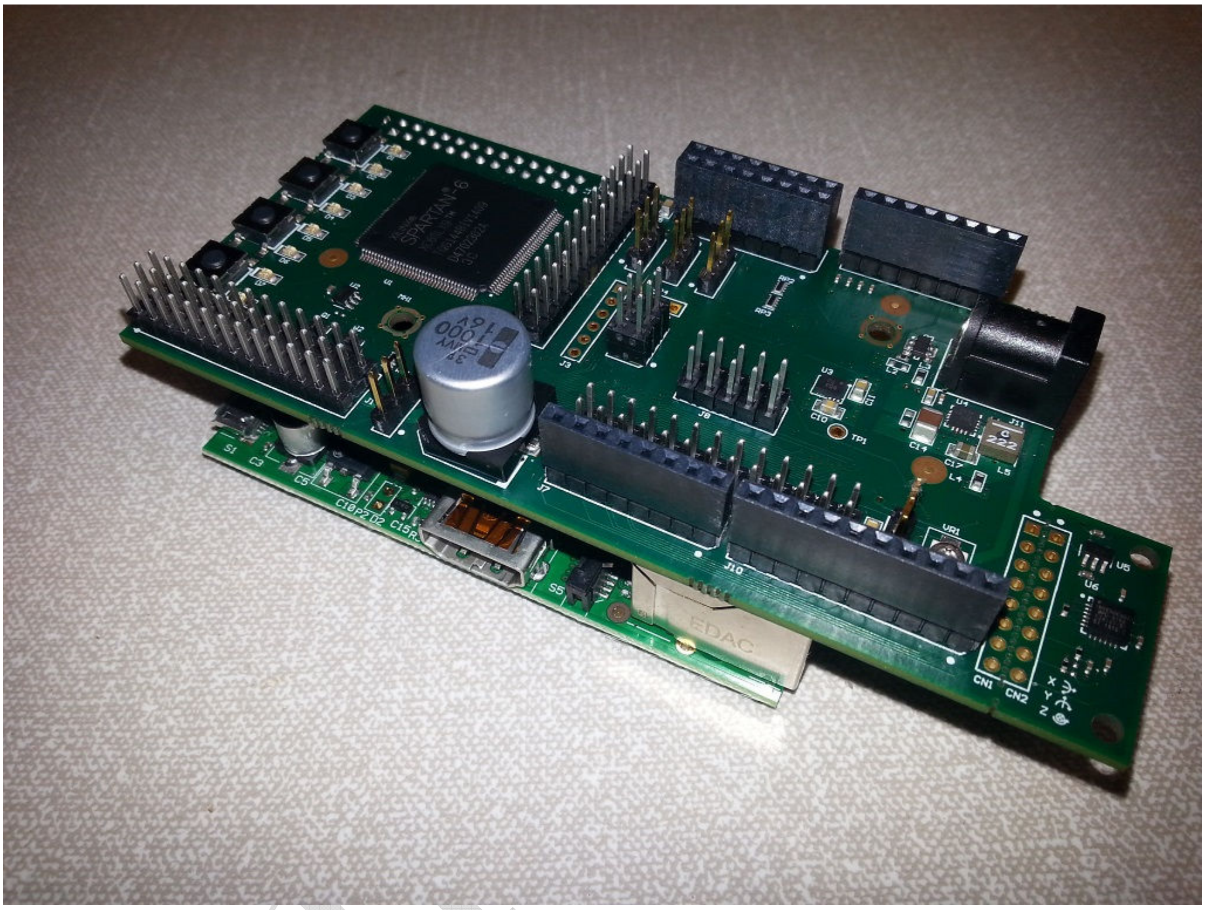
Application Note AN-025 (v1.1b)

PiXi SPI & I2C Register Map



Summary

The PiXi add-on board is designed to expand the general-purpose I/O capabilities of the Raspberry Pi and provide a low cost means of introducing the user to the world of digital electronics and FPGA technology as well as giving the 'Pi Enthusiast' a few more features to play with. The low product cost and feature-packed specification of the PiXi-200 makes it ideal for applications in computing, hobby-electronics, education, training and product development.

This application note describes the full SPI & I2C register map of the PiXi FPGA functions. At the time of writing this document, only the SPI interface is enabled however the I2C interface will be enabled at a later date and will support a similar register map to the one available over the SPI interface.

Changes

Revision	Date	Changes
1.0	5/07/2014	Preliminary release
1.1a	TBD	TBD
1.1b	17/07/2014	Added details of the Pi GPIO Configuration Register

SPI Interface

The SPI interface in the PiXi FPGA is a three to six byte interface where the first two bytes carry the register address, read/write instruction and word-length controls and the following sequence of bytes byte carry the data. Data can be 8-bit, 16-bit or 32-bit in length. All register locations are 32-bit wide, regardless of the word-length control. Most registers are just 8-bit or 16-bit wide with un-used bits ignored or set to ‘0’.

I^{2}C Interface

The I2C interface provides a simple I2C slave port that provides access to the entire register map plus an I2C switch to allow other devices to be seamlessly connected to the the I2C bus.

More information on the I2C interface will be provided later…

PiXi-Tools

PiXi-Tools provides a convenient set of applications and libraries for accessing the PiXi functions and FPGA registers over SPI & I2C. For more information on installing and using PiXi-Tools, please refer to application note AN-020.

Register Map Summary

The standard FPGA on the PiXi provides register-mapped control & status registers for a host of functions in the FPGA which can be accessed through the SPI interface on the Raspberry Pi.

Build-time (Build ID) & Test Registers		
Address:	**Read / Write**	**Register Function**
0x00 [0]	R (16b)	REG_BUILD_TIME0 Build Time Register (0xMMYY)
0x01 [1]	R (16b)	REG_BUILD_TIME0 Build Time Register (0xssDD)
0x02 [2]	R (16b)	REG_BUILD_TIME0 Build Time Register (0xhhmm)
0x03 [3]	R (16b)	REG_TEST3 Test register 3: Fixed read value (0x3210)
0x04 [4]	R (16b)	REG_TEST4 Test register 4: Fixed read value (0x7654)
0x05 [5]	R (16b)	REG_TEST5 Test register 5: Fixed read value (0x5555)
0x06 [6]	R (16b)	REG_TEST6 Test register 6: Fixed read value (0xAAAA)
0x07 [7]	R/W (16b)	REG_TEST7 Test register 7: Write / Read Testing.

SPI & I2C Configuration Registers		
Address:	**Read / Write**	**Register Function**
0x08 [8]	W	REG_I2C_CFG(0) I2C Configuration Register
0x09 [9]	W	REG_SPI_CFG(0) SPI Configuration Register

Serial Port Configuration Register		
Address:	**Read / Write**	**Register Function**
0x0A [10]	W (16b)	REG_SERIAL_CFG(0) Serial Port Configuration Register

Board Information Register		
Address:	**Read / Write**	**Register Function**
0x0E [14]	W (16b)	REG_BOARD_INFO Board Information Register. Miscellaneous status.

Raspberry Pi GPIO Configuration Register **Defines how the Raspberry Pi GPIO connector is used**		
Address:	**Read / Write**	**Register Function**
0x10 [16]	W (16b)	PI_GPIO(3:0) Mode
0x11 [17]	W (16b)	PI_GPIO(7:4) Mode
0x12 [18]	W (16b)	PI_GPIO(11:8) Mode
0x13 [19]	W (16b)	PI_GPIO(15:12) Mode
0x14 [20]	W (16b)	PI_GPIO(19:16) Mode

GPIO Input Output Registers		
Address:	**Read / Write**	**Register Function**
0x20 [32]	R/W (8b)	GPIO1(7:0) Input / Output
0x21 [33]	R/W (8b)	GPIO1(15:8) Input / Output
0x22 [34]	R/W (8b)	GPIO1(23:16) Input / Output
0x23 [35]	R/W (8b)	GPIO2(7:0) Input / Output
0x24 [36]	R/W (8b)	GPIO2(15:8) Input / Output
0x25 [37]	R/W (8b)	GPIO3(7:0) Input / Output
0x26 [38]	R/W (8b)	GPIO3(8:0) Input / Output

GPIO Mode Control Registers		
Address:	**Read / Write**	**Register Function**
0x28 [40]	R/W (16b)	GPIO1(3:0) Mode
0x29 [41]	R/W (16b)	GPIO1(7:4) Mode
0x2A [42]	R/W (16b)	GPIO1(11:8) Mode
0x2B [43]	R/W (16b)	GPIO1(15:12) Mode
0x2C [44]	R/W (16b)	GPIO1(19:16) Mode
0x2D [45]	R/W (16b)	GPIO1(23:20) Mode
0x2E [46]	R/W (16b)	GPIO2(3:0) Mode
0x2F [47]	R/W (16b)	GPIO2(7:4) Mode
0x30 [48]	R/W (16b)	GPIO2(11:8) Mode
0x31 [49]	R/W (16b)	GPIO2(15:12) Mode
0x32 [50]	R/W (16b)	GPIO3(3:0) Mode
0x33 [51]	R/W (16b)	GPIO3(7:4) Mode
0x34 [52]	R/W (16b)	GPIO3(11:8) Mode
0x35 [53]	R/W (16b)	GPIO3(15:12) Mode

LED Output & Configuration Registers		
Address:	**Read / Write**	**Register Function**
0x36 [54]	W (16b)	REG_LEDS LED Output Register
0x37 [55]	W (16b)	REG_LED_CTRL LED Function Control Register

LCD / VFD Output & Configuration Registers		
Address:	**Read / Write**	**Register Function**
0x38 [56]	W (16b)	REG_VFD LCD/VFD Output Register
0x39 [57]	W (16b)	REG_VFD_CTRL LCD/VFD Control Register

Switch Input Registers		
Address:	**Read / Write**	**Register Function**
0x3A [58]	R (8b)	REG_SWITCHES Switch Input Register

Keypad Input & Configuration Registers		
Address:	**Read / Write**	**Register Function**
0x3B [59]	R (16b)	REG_KEYPAD Matrix Keypad Input Register

Buzzer Control Register Registers		
Address:	**Read / Write**	**Register Function**
0x3C [60]	W (16b)	REG_BUZ0 Frequency Control Register (bits 15:0)
0x3D [61]	W (16b)	REG_BUZ1 Frequency Control Register (bits 23:16) Buzzer Enable

PWM Output & Configuration Registers		
Address:	**Read / Write**	**Register Function**
0x40 [64]	W (16b)	REG_PWM0 PWM Ch0 Pulse Width Output Register
0x41 [65]	W (16b)	REG_PWM1 PWM Ch1 Pulse Width Output Register
0x42 [66]	W (16b)	REG_PWM2 PWM Ch2 Pulse Width Output Register
0x43 [67]	W (16b)	REG_PWM3 PWM Ch3 Pulse Width Output Register
0x44 [68]	W (16b)	REG_PWM4 PWM Ch4 Pulse Width Output Register
0x45 [69]	W (16b)	REG_PWM5 PWM Ch5 Pulse Width Output Register
0x46 [70]	W (16b)	REG_PWM6 PWM Ch6 Pulse Width Output Register
0x47 [71]	W (16b)	REG_PWM7 PWM Ch7 Pulse Width Output Register
0x48 [72]	W (16b)	REG_PWM8 PWM Ch8 Pulse Width Output Register
0x49 [73]	W (16b)	REG_PWM9 PWM Ch9 Pulse Width Output Register
0x4A [74]	W (16b)	REG_PWM10 PWM Ch9 Pulse Width Output Register
0x4B [75]	W (16b)	REG_PWM11 PWM Ch9 Pulse Width Output Register
0x40 [64]	R (16b)	REG_PWM_IN0 PWM Ch0 Pulse Width Input Register
0x41 [65]	R (16b)	REG_PWM_IN1 PWM Ch1 Pulse Width Input Register
0x42 [66]	R (16b)	REG_PWM_IN2 PWM Ch2 Pulse Width Input Register
0x43 [67]	R (16b)	REG_PWM_IN3 PWM Ch3 Pulse Width Input Register
0x44 [68]	R (16b)	REG_PWM_IN4 PWM Ch4 Pulse Width Input Register
0x45 [69]	R (16b)	REG_PWM_IN5 PWM Ch5 Pulse Width Input Register
0x46 [70]	R (16b)	REG_PWM_IN6 PWM Ch6 Pulse Width Input Register
0x47 [71]	R (16b)	REG_PWM_IN7 PWM Ch7 Pulse Width Input Register
0x48 [72]	R (16b)	REG_PWM_IN8 PWM Ch8 Pulse Width Input Register
0x49 [73]	R (16b)	REG_PWM_IN9 PWM Ch9 Pulse Width Input Register
0x4A [74]	R (16b)	REG_PWM_IN10 PWM Ch9 Pulse Width Input Register
0x4B [75]	R (16b)	REG_PWM_IN11 PWM Ch9 Pulse Width Input Register
0x4D [77]	W (16b)	RESERVED
0x4E [78]	W (16b)	RESERVED
0x4F [79]	W (16b)	REG_CFG PWM Configuration Register

General Purpose Timer Registers		
Address:	**Read / Write**	**Register Function**
0x50 [80]	W (16b)	REG_TIMER0 Timer Output Register (bits 15:0)
0x51 [81]	W (16b)	REG_TIMER1 Timer Output Register (bits (31:16)
0x54 [83]	W (16b)	REG_TIMER_CFG Timer Control Register

General Purpose Counter Registers		
Address:	**Read / Write**	**Register Function**
0x58 [88]	W (16b)	REG_COUNTER0 Timer Output Register (bits 15:0)
0x59 [89]	W (16b)	REG_COUNTER1 Timer Output Register (bits (31:16)
0x5C [82]	W (16b)	REG_COUNTER_CFG Counter Control Register

Rotary Encoder Registers		
Address:	**Read / Write**	**Register Function**
0x60 [96]	W (16b)	REG_ENC0 Encoder Ch0 Position Register
0x61 [97]	W (16b)	REG_ENC1 Encoder Ch1 Position Register
0x62 [98]	W (16b)	REG_ENC2 Encoder Ch2 Position Register
0x63 [99]	W (16b)	REG_ENC3 Encoder Ch3 Position Register
0x64 [100]	W (16b)	REG_ENC4 Encoder Ch4 Position Register
0x65 [101]	W (16b)	REG_ENC5 Encoder Ch5 Position Register
0x66 [102]	W (16b)	REG_ENC6 Encoder Ch6 Position Register
0x67 [103]	W (16b)	REG_ENC7 Encoder Ch7 Position Register

UART Configuration & Data Registers		
Address:	**Read / Write**	**Register Function**
0x80 [128]	W (16b)	REG_START_UART1 16550 UART Channel 1 Base Address
0x88 [136]	W (16b)	REG_START_UART2 16550 UART Channel 2 Base Address
0x90 [144]	W (16b)	REG_START_UART3 16550 UART Channel 3 Base Address
0x98 [152]	W (16b)	REG_START_UART4 16550 UART Channel 4 Base Address

Pi Power Control Registers		
Address:	**Read / Write**	**Register Function**
0xEF [239]	W (16b)	REG_PI_SWITCH Pi Power Control Register

Run Time Counter Registers		
Address:	**Read / Write**	**Register Function**
0xF0 [240]	W (16b)	REG_RUNTIME0 (Increments once per second after configuration) Run-time Counter (bits 15:0)
0xF1 [241]	W (16b)	REG_RUNTIME0 Run-time Counter (bits 31:16)

FPGA 'DNA' Serial Number Registers		
Address:	**Read / Write**	**Register Function**
0xF4 [244]	R (16b)	REG_DNA0 DNA (15:0)
0xF5 [245]	R (16b)	REG_DNA1 DNA (31:16)
0xF6 [246]	R (16b)	REG_DNA2 DNA (47:32)
0xF7 [247]	R (16b)	REG_DNA3 DNA (55:48)

Register Map Description

The following tables provide details on how to registers available over SPI & I2C can be used:

Test Registers (Details)		
Address:	**Read / Write**	**Register Function**
0x00 [0] Bits(15:0)	R (16b)	Build Time Register (0xMMYY) Returns a time-stamp from when the FPGA was compiled YY: FPGA Build-time (year) in BCD format MM: FPGA Build-time (month) in BCD format
0x01 [1] Bits(15:0)	R (16b)	Build Time Register (0xssDD) Returns a time-stamp from when the FPGA was compiled DD: FPGA Build-time (day) in BCD format ss: FPGA Build-time (seconds) in BCD format
0x02 [2] Bits(15:0)	R (16b)	Build Time Register (0xhhmm) Returns a time-stamp from when the FPGA was compiled mm: FPGA Build-time (minute) in BCD format hh: FPGA Build-time (hour) in BCD format
0x03 [3] Bits(15:0)	R (16b)	Test register 3: Basic test register to help verify SPI / I2C read/write functions Fixed read value (0x3210)
0x04 [4] Bits(15:0)	R (16b)	Test register 4: Basic test register to help verify SPI / I2C read/write functions Fixed read value (0x7654)
0x05 [5] Bits(15:0)	R (16b)	Test register 5: Basic test register to help verify SPI / I2C read/write functions Fixed read value (0x5555)
0x06 [6] Bits(15:0)	R (16b)	Test register 6: Basic test register to help verify SPI / I2C read/write functions Fixed read value (0xAAAA)
0x07 [7] Bits(15:0)	R/W (16b)	Test register 7: Basic test register to help verify SPI / I2C read/write functions Reading this register returns the data that was last written to this register

I2C Configuration Register (0x08)	
Bit(s)	**Function:**
7:0	I2C Address (Default = 0x75)

SPI Configuration Register (0x09)	
Bit(s)	**Function:**
3:0	~~SPI Channel 0 Function~~ ~~"0000": PiXi FPGA~~ ~~"0001": Not used (select if using Ch1 for external applications)~~ ~~"0010": SPI Flash~~ ~~"0011": Reserved~~
7:4	~~SPI Channel 1 Function~~ ~~"0000": PiXi FPGA~~ ~~"0001": Not used (select if using Ch1 for external applications~~ ~~Such as the PiXi on-board ADC)~~ ~~"0010": SPI Flash~~ ~~"0011": Reserved~~

Serial Port Configuration Register (Details) (0x0A)	
Bit(s)	**Function:**
2:0	RX / TX pin Configuration Configures which UART connects to RX/TX pins on the Serial Port "000": Raspberry Pi RXD/TXD (Default) "001": UART1 RX/TX "010": UART2 RX/TX
3	Not used
6:4	CTS/RX2 / RTS/TX2 pin Configuration Configures which UART connects to CTS/RTS pins on the Serial Port "000": Raspberry Pi RXD/TXD "001": UART1 RX/TX (Default) "010": UART2 RX/TX
15:7	Not used

Board Information Register	
Bit(s)	**Function:**
0	FPGA_PUDC_B '0': This indicates that all FPIO I/O will be pulled to '1' when the FPGA is not configured. '1': This indicates that all FPGA I/O will be tri-stated when the FPGA is not configured.
2:1	FPGA_M(1:0) "10": Master SPI Mode (Program from flash) "11": Slave Serial Mode (Program from Raspberry Pi)

Raspberry Pi GPIO Configuration Registers (0x10, 0x11, 0x12, 0x13, 0x14)	
Bit(s)	**Function:**
0x10: 3:0	PI_GPIO_GEN(0) Function Select Configures how GPIO_GEN(0) on the Raspberry Pi is used on the PiXi “0000”: Input (to PiXi) “0001”: Reserved “0010”: Reserved “0011”: Reserved “0100”: Reserved “0101”: Reserved “0110”: Reserved “0111”: Reserved
0x10: 7:4	PI_GPIO_GEN(1) Input Select Configures which source drives GPIO_GEN(1) on the Raspberry Pi “0000”: Input (to PiXi) “0001”: Reserved “0010”: Reserved “0011”: Reserved “0100”: Reserved “0101”: Reserved “0110”: Reserved “0111”: Reserved
0x10: 11:8	PI_GPIO_GEN(2) Input Select Configures which source drives GPIO_GEN(2) on the Raspberry Pi “0000”: Input (to PiXi) “0001”: Reserved “0010”: Reserved “0011”: Reserved “0100”: Reserved “0101”: Reserved “0110”: Reserved “0111”: Reserved
0x10: 15:12	PI_GPIO_GEN(3) Input Select Configures which source drives GPIO_GEN(3) on the Raspberry Pi “0000”: Input (to PiXi) “0001”: MPU_INT_N “0010”: UART(0) INTERRUPT “0011”: UART(0) RECEIVE_FIFO_EMPTY “0100”: UART(0) TRANSMIT_FIFO_FULL “0101”: UART(1) INTERRUPT “0110”: UART(1) RECEIVE_FIFO_EMPTY “0111”: UART(1) TRANSMIT_FIFO_FULL
0x11: 3:0	PI_GPIO_GEN(4) Input Select Configures which source drives GPIO_GEN(4) on the Raspberry Pi “0000”: Input (to PiXi) “0001”: DAC_RDY “0010”: UART(0) INTERRUPT “0011”: UART(0) RECEIVE_FIFO_EMPTY “0100”: UART(0) TRANSMIT_FIFO_FULL “0101”: UART(1) INTERRUPT “0110”: UART(1) RECEIVE_FIFO_EMPTY “0111”: UART(1) TRANSMIT_FIFO_FULL
0x11: 7:4	PI_GPIO_GEN(5) Input Select Configures which source drives GPIO_GEN(5) on the Raspberry Pi “0000”: Input (to PiXi) “0001”: MPU_INT_N “0010”: UART(2) INTERRUPT “0011”: UART(2) RECEIVE_FIFO_EMPTY “0100”: UART(2) TRANSMIT_FIFO_FULL

	“0101”: UART(3) INTERRUPT “0110”: UART(3) RECEIVE_FIFO_EMPTY “0111”: UART(3) TRANSMIT_FIFO_FULL
0x11: 7:4	PI_GPIO_GEN(6) Input Select Configures which source drives GPIO_GEN(6) on the Raspberry Pi “0000”: Input (to PiXi) “0001”: DAC_RDY “0010”: UART(2) INTERRUPT “0011”: UART(2) RECEIVE_FIFO_EMPTY “0100”: UART(2) TRANSMIT_FIFO_FULL “0101”: UART(3) INTERRUPT “0110”: UART(3) RECEIVE_FIFO_EMPTY “0111”: UART(3) TRANSMIT_FIFO_FULL
0x11: 11:8	PI_GPIO_GCLK Input Select Configures which source drives GPIO_GCLK on the Raspberry Pi “0000”: Input (to PiXi) “0001”: Reserved “0010”: Reserved “0011”: Reserved “0100”: Reserved “0101”: Reserved “0110”: Reserved “0111”: Reserved

GPIO1 Mode Control Register 1 (0x28)	
Bit(s)	**Function:**
3:0	GPIO1(0) Mode: "0000": Input (read pin status at register 0x20, bit0) "0001": Output (set using register 0x20, bit0) “0010”: Matrix keypad I/O "1000": Raspberry Pi RXD (Input)
7:4	GPIO1(1) Mode: "0000": Input (read pin status at register 0x20, bit1) "0001": Output (set using register 0x20, bit1) “0010”: Matrix keypad I/O "1000": Raspberry Pi TXD (Output)
6:4	GPIO1(2) Mode: "0000": Input (read pin status at register 0x20, bit2) "0001": Output (set using register 0x20, bit2) “0010”: Matrix keypad I/O "1000": UART1 RXD (Input)
15:7	GPIO1(3) Mode: "0000": Input (read pin status at register 0x20, bit3) "0001": Output (set using register 0x20, bit3) “0010”: Matrix keypad I/O "1000": UART1 TXD (Output)

GPIO1 Mode Control Register 2 (0x29)	
Bit(s)	**Function:**
3:0	GPIO1(4) Mode: "0000": Input (read pin status at register 0x20, bit4) "0001": Output (set using register 0x20, bit4) “0010”: Matrix keypad I/O "1000": UART2 RXD (Input)
7:4	GPIO1(5) Mode: "0000": Input (read pin status at register 0x20, bit5) "0001": Output (set using register 0x20, bit5) “0010”: Matrix keypad I/O "1000": UART2 TXD (Output)
6:4	GPIO1(6) Mode: "0000": Input (read pin status at register 0x20, bit6) "0001": Output (set using register 0x20, bit6) “0010”: Matrix keypad I/O "1000": UART3 RXD (Input)
15:7	GPIO1(7) Mode: "0000": Input (read pin status at register 0x20, bit7) "0001": Output (set using register 0x20, bit7) “0010”: Matrix keypad I/O "1000": UART3 TXD (Output)

GPIO1 Mode Control Register 3 (0x2A)	
Bit(s)	**Function:**
3:0	GPIO1(8) Mode: "0000": Input (read pin status at register 0x21, bit0) "0001": Output (set using register 0x21, bit0) “0010”: Matrix keypad I/O "1000": Raspberry Pi RXD (Input)
7:4	GPIO1(9) Mode: "0000": Input (read pin status at register 0x21, bit1) "0001": Output (set using register 0x21, bit1) “0010”: Matrix keypad I/O "1000": Raspberry Pi TXD (Output)
6:4	GPIO1(10) Mode: "0000": Input (read pin status at register 0x21, bit2) "0001": Output (set using register 0x21, bit2) “0010”: Matrix keypad I/O "1000": UART1 RXD (Input)
15:7	GPIO1(11) Mode: "0000": Input (read pin status at register 0x21, bit3) "0001": Output (set using register 0x21, bit3) “0010”: Matrix keypad I/O "1000": UART1 TXD (Output)

GPIO1 Mode Control Register 4 (0x2B)	
Bit(s)	**Function:**
3:0	GPIO1(12) Mode: "0000": Input (read pin status at register 0x21, bit4) "0001": Output (set using register 0x21, bit4) “0010”: Matrix keypad I/O "1000": Raspberry Pi RXD (Input)
7:4	GPIO1(13) Mode: "0000": Input (read pin status at register 0x21, bit5) "0001": Output (set using register 0x21, bit5) “0010”: Matrix keypad I/O "1000": Raspberry Pi TXD (Output)
6:4	GPIO1(14) Mode: "0000": Input (read pin status at register 0x21, bit6) "0001": Output (set using register 0x21, bit6) “0010”: Matrix keypad I/O "1000": UART1 RXD (Input)
15:7	GPIO1(15) Mode: "0000": Input (read pin status at register 0x21, bit7) "0001": Output (set using register 0x21, bit7) “0010”: Matrix keypad I/O "1000": UART1 TXD (Output)

GPIO1 Mode Control Register 5 (0x2C)	
Bit(s)	**Function:**
3:0	GPIO1(16) Mode: "0000": Input (read pin status at register 0x22, bit3) "0001": Output (set using register 0x22, bit0)
7:4	GPIO1(17) Mode: "0000": Input (read pin status at register 0x22, bit3) "0001": Output (set using register 0x22, bit1)
6:4	GPIO1(18) Mode: "0000": Input (read pin status at register 0x22, bit3) "0001": Output (set using register 0x22, bit2)
15:7	GPIO1(19) Mode: "0000": Input (read pin status at register 0x22, bit3) "0001": Output (set using register 0x22, bit3)

GPIO1 Mode Control Register 6 (0x2D)	
Bit(s)	**Function:**
3:0	GPIO1(20) Mode: "0000": Input (read pin status at register 0x22, bit5) "0001": Output (set using register 0x22, bit4)
7:4	GPIO1(21) Mode: "0000": Input (read pin status at register 0x22, bit5) "0001": Output (set using register 0x22, bit5)
6:4	GPIO1(22) Mode: "0000": Input (read pin status at register 0x22, bit6) "0001": Output (set using register 0x22, bit6)
15:7	GPIO1(23) Mode: "0000": Input (read pin status at register 0x22, bit7) "0001": Output (set using register 0x22, bit7)

GPIO2 Mode Control Register 1 (0x2E)	
Bit(s)	**Function:**
3:0	GPIO2 bit(0) Mode: "0000": Not used (input is not supported on GPIO2) "0001": Output (set using register 0x23, bit0) "0010": PWM Ch0
7:4	GPIO2 bit(1) Mode: "0000": Not used (input is not supported on GPIO2) "0001": Output (set using register 0x23, bit1) "0010": PWM Ch1
11:8	GPIO2 bit(2) Mode: "0000": Not used (input is not supported on GPIO2) "0001": Output (set using register 0x23, bit2) "0010": PWM Ch2
15:12	GPIO2 bit(3) Mode: "0000": Not used (input is not supported on GPIO2) "0001": Output (set using register 0x23, bit3) "0010": PWM Ch3

GPIO2 Mode Control Register 2 (0x2F)	
Bit(s)	**Function:**
3:0	GPIO2 bit(4) Mode: "0000": Not used (input is not supported on GPIO2) "0001": Output (set using register 0x23, bit4) "0010": PWM Ch4
7:4	GPIO2 bit(5) Mode: "0000": Not used (input is not supported on GPIO2) "0001": Output (set using register 0x23, bit5) "0010": PWM Ch5
11:8	GPIO2 bit(6) Mode: "0000": Not used (input is not supported on GPIO2) "0001": Output (set using register 0x23, bit6) "0010": PWM Ch6
15:12	GPIO2 bit(7) Mode: "0000": Not used (input is not supported on GPIO2) "0001": Output (set using register 0x23, bit7) "0010": PWM Ch7

GPIO2 Mode Control Register 3 (0x30)	
Bit(s)	**Function:**
3:0	GPIO2 bit(8) Mode: "0000": Not used (input is not supported on GPIO2) "0001": Output (set using register 0x24, bit0) "0010": ‘1’ (fixed to enable GND on servo connector)
7:4	GPIO2 bit(9) Mode: "0000": Not used (input is not supported on GPIO2) "0001": Output (set using register 0x24, bit1) "0010": PWM Ch8
11:8	GPIO2 bit(10) Mode: "0000": Not used (input is not supported on GPIO2) "0001": Output (set using register 0x24, bit2) "0010": ‘1’ (fixed to enable GND on servo connector)
15:12	GPIO2 bit(11) Mode: "0000": Not used (input is not supported on GPIO2) "0001": Output (set using register 0x24, bit3) "0010": PWM Ch9

GPIO2 Mode Control Register 4 (0x31)	
Bit(s)	**Function:**
3:0	GPIO2 bit(12) Mode: "0000": Not used (input is not supported on GPIO2) "0001": Output (set using register 0x24, bit4) "0010": ‘1’ (fixed to enable GND on servo connector)
7:4	GPIO2 bit(13) Mode: "0000": Not used (input is not supported on GPIO2) "0001": Output (set using register 0x24, bit5) "0010": PWM Ch10
11:8	GPIO2 bit(14) Mode: "0000": Not used (input is not supported on GPIO2) "0001": Output (set using register 0x24, bit6) "0010": ‘1’ (fixed to enable GND on servo connector)
15:12	GPIO2 bit(15) Mode: "0000": Not used (input is not supported on GPIO2) "0001": Output (set using register 0x24, bit7) "0010": PWM Ch11

GPIO3 Mode Control Register 1 (0x32)	
Bit(s)	**Function:**
3:0	GPIO3(0) Mode: "0000": Input (read pin status at register 0x25, bit0) "0001": Output (set using register 0x25, bit0) “0010”: LCD / VFD (D1) “0111”: PWM Input Ch0
7:4	GPIO3(1) Mode: "0000": Input (read pin status at register 0x25, bit1) "0001": Output (set using register 0x25, bit1) “0010”: LCD / VFD (D0) “0111”: PWM Input Ch1
6:4	GPIO3(2) Mode: "0000": Input (read pin status at register 0x25, bit2) "0001": Output (set using register 0x25, bit2) “0010”: LCD / VFD (D3) “0111”: PWM Input Ch2
15:7	GPIO3(3) Mode: "0000": Input (read pin status at register 0x25, bit3) "0001": Output (set using register 0x25, bit3) “0010”: LCD / VFD (D2) “0111”: PWM Input Ch3

GPIO3 Mode Control Register 2 (0x33)	
Bit(s)	**Function:**
3:0	GPIO3(4) Mode: "0000": Input (read pin status at register 0x25, bit4) "0001": Output (set using register 0x25, bit4) “0010”: LCD / VFD (D5) “0111”: PWM Input Ch4
7:4	GPIO3(5) Mode: "0000": Input (read pin status at register 0x25, bit5) "0001": Output (set using register 0x25, bit5) “0010”: LCD / VFD (D4) “0111”: PWM Input Ch5
6:4	GPIO3(6) Mode: "0000": Input (read pin status at register 0x25, bit6) "0001": Output (set using register 0x25, bit6) “0010”: LCD / VFD (D7) “0111”: PWM Input Ch6
15:7	GPIO3(7) Mode: "0000": Input (read pin status at register 0x25, bit7) "0001": Output (set using register 0x25, bit7) “0010”: LCD / VFD (D6) “0111”: PWM Input Ch7

GPIO3 Mode Control Register 3 (0x34)	
Bit(s)	**Function:**
3:0	GPIO3(8) Mode: "0000": Input (read pin status at register 0x26, bit0) "0001": Output (set using register 0x26, bit0) “0010”: LCD / VFD (RS)
7:4	GPIO3(9) Mode: "0000": Input (read pin status at register 0x26, bit1) "0001": Output (set using register 0x26, bit1) “0010”: Input (read pin status at register 0x26, bit1)
6:4	GPIO3(10) Mode: "0000": Input (read pin status at register 0x26, bit2) "0001": Output (set using register 0x26, bit2) “0010”: LCD / VFD (#WR)
15:7	GPIO3(11) Mode: "0000": Input (read pin status at register 0x26, bit3) "0001": Output (set using register 0x26, bit3) “0010”: LCD / VFD (#RD)

GPIO3 Mode Control Register 4 (0x35)	
Bit(s)	**Function:**
3:0	GPIO3(12) Mode: "0000": Input (read pin status at register 0x26, bit4) "0001": Output (set using register 0x26, bit4) “0010”: Input (read pin status at register 0x26, bit4) “0111”: PWM Input Ch8
7:4	GPIO3(13) Mode: "0000": Input (read pin status at register 0x26, bit5) "0001": Output (set using register 0x26, bit5) “0010”: Input (read pin status at register 0x26, bit5) “0111”: PWM Input Ch9
6:4	GPIO3(14) Mode: "0000": Input (read pin status at register 0x26, bit6) "0001": Output (set using register 0x26, bit6) “0010”: Input (read pin status at register 0x26, bit6) “0111”: PWM Input Ch10
15:7	GPIO3(15) Mode: "0000": Input (read pin status at register 0x26, bit7) "0001": Output (set using register 0x26, bit7) “0010”: Input (read pin status at register 0x26, bit7) “0111”: PWM Input Ch11

LED Output Register Details (0x36)	
Bit(s)	**Function:**
1:0	LED0_OUTPUT "00": Off "01": Slow flash “10”: Fast flash “11”: On
3:2	LED1_OUTPUT "00": Off "01": Slow flash “10”: Fast flash “11”: On
5:4	LED2_OUTPUT "00": Off "01": Slow flash “10”: Fast flash “11”: On
7:6	LED2_OUTPUT "00": Off "01": Slow flash “10”: Fast flash “11”: On

LED Configuration Register Details (0x37)	
Bit(s)	**Function:**
3:0	LED Driver Function Select: "0000": Direct according to LED Output Register "0001": Reserved “0010”: Reserved “0011”: Reserved

LCD / VFD Output Register Details (0x38)	
Bit(s)	**Function:**
7:0	LCD / VFD Write Data Character data to be written direct to the display
8	Not used
9	RS Sometimes used to select between data, configuration or other special function.
11:10	Not used
15:12	Wait time Used to delay writes in case they occur to fast for a particular make of LCD / VFD. "0000": No wait

LCD / VFD Configuration Register Details (0x38)	
Bit(s)	**Function:**
0	LCD/VFD Interface Mode "00"': Motorola (R/#W, EN#) (Active-Low Enable) "10"': Motorola (R/#W, EN) (Active-High Enable) "01": Intel i80 (#RD, #WR)
15:12	Reserved

Switch Input Register Details (0x3A)	
Bit(s)	**Function:**
 0 1 2 3	Switch Status: '0' = Off / Released, '1' = On / Pressed SW1 SW2 SW3 SW4
 4 5 6 7	Switch Activity since last read: '0' = no activity, '1' = change detected SW1 SW2 SW3 SW4

Keypad Input Register Details (0x3B)	
Bit(s)	**Function:**
7:0	Key Code (ASCII)
8	Buffer Empty
9	Buffer Full
10	Key Up (Key was released)
11	Key Down (Key was pressed)

Buzzer Control Register Register Details (0x3C, 0x3D)	
Bit(s)	**Function:**
0x3C: 15:0 0x3D: 7:0	Buzzer Frequency Control (15:0) Buzzer Frequency Control (23:16) Buzzer frequency = 25MHz / (2 x 'Frequency Control')
0x3D: 15	Buzzer On / Off control '0': Buzzer is off '1': Buzzer is on

PWM Output & Configuration Registers (0x40, 0x41, 0x42, …0x4B)	
Bit(s)	**Function:**
9:0	Read: PWM Input pulse width (0 = 0%, 1023 = 100%) Write: PWM Output pulse width (0 = 0%, 1023 = 100%)
0x4D 15:0	Reserved
0x4E 15:0	Reserved
0x4F 15:0	Reserved

General Purpose Timer Registers	
Bit(s)	**Function:**
0x50: 15:0 0x51: 31:16	Read: 32-bit unsigned timer, incrementing or decrementing depending on direction of count.
0x54: 15:0	Write: TBD

General Purpose Counter Registers	
Bit(s)	**Function:**
0x58: 15:0 0x59: 31:16	Read: 32-bit signed (2's comp) counter, incrementing or decrementing depending on direction of count.
0x5C: 15:0	Write: TBD

Rotary Encoder Registers (0x60, 0x61, 0x62, … 0x67)	
Bit(s)	**Function:**
0x00: 7:0	Read: 8-bit signed (2's comp) counter, incrementing or decrementing depending on direction of rotation & amount of rotation.

UART Configuration & Data Registers (0x80, 0x88, 0x90, 0x98)	
Bit(s)	**Function:**
0x00: 7:0	LCR(7) = '0': Read: RBR - RX Data Write: THR - TX Data LCR(7) = '1' Read/Write: DLL
0x01: 7:0	LCR(7) = '0': Read/Write: IER LCR(7) = '1' Read/Write: DLM
0x02: 7:0	Read: IIR Write: FCR
0x03: 7:0	Read/Write: LCR
0x04: 7:0	Read/Write: MCR
0x05: 7:0	Read: LSR
0x06: 7:0	Read: MSR
0x07: 7:0	Read/Write: SCR

Pi Power Control Registers	
Bit(s)	**Function:**
1:0	Read: "00": Power Off "01": Starting Up "11": Running "10": Shutting Down Write: "00": Power Off immediately "01": Start up & wait for start-up completion "11": Switch on immediately "10": Shut Down (Safe Request)

Run Time Counter Registers (0xF0, 0xF1)	
Bit(s)	**Function:**
0xF0: 15:0 0xF1: 15:0	Runtime0 Runtime1 Runtime since last configuration = Runtime1 x 65536 + Runtime0

FPGA 'DNA' Serial Number Registers	
Bit(s)	**Function:**
0xF4: 15:0	DNA Serial Number (15:0)
0xF4: 15:0	DNA Serial Number (31:16)
0xF6: 15:0	DNA Serial Number (47:32)
0xF7: 7:0	DNA Serial Number (55:48)
0xF7: 15	DNA Busy '0': Ready to read '1': Busy – wait…

All of these registers can be set up using the Raspberry Pi's SPI interface. PiXi-Tools provides some general-purpose functions for writing to and reading the PiXi FPGAs registers over SPI. Please see application note AN-020 for more information on installing and using the PiXi-Tools applications and libraries.

Further Reading

The PiXi User Manual (UM-002) has complete information on the pin functions for serial and other interfaces on the PiXi.

If you want to learn more about programming the FPGA on the PiXi to customise the serial port, please take a look at application notes AN-002 "Programming the FPGA on the PiXi" and AN-003 "FPGA Development on the PiXi".

PiXi-Tools is described in more detail in application note AN-020 "Installing PiXi-Tools on the Raspberry Pi".

The full register map for the PiXi can be found in application note AN-025 "PiXi SPI & I2C Register Map".

All of these documents are available for download from www.astro-designs.com.

Acknowledgements

"Raspberry Pi" is a trademark of the Raspberry Pi Foundation.