PCIE to UART (F81504/508/512) Programming Guide

Released Date: 2013/06

Table of Contents

1. PCIE to UART Bridge Type 0 Configuration Settings

PCIE Configuration

31:42	23:16	15:08	7:0	Byte Offset
Device ID		Vendor ID		000h
Status		Command		004h
Class Code			Revision ID	008h
BIST	Header Type	Latency Timer	Cache Line	00Ch
Base Address Registers				010h
				014h
				018h
				01Ch
				020h
				024h
Cardbus CIS Pointer				028h
Subsystem ID		Subsystem Vendor ID		02Ch
Expansion ROM Base Address				030h
Reserved			Capabilities PTR	034h
Reserved				038h
Max_Lat	Min_Gnt	Int. Pin	Int. Line	03Ch

	63:0	Byte Offset
F81504 **Device ID**: 1104 UART 1 ~ UART 4	UART 1 Device Configuration Registers	040h
	UART 2 Device Configuration Registers	048h
	UART 3 Device Configuration Registers	050h
	UART 4 Device Configuration Registers	058h
F81508 **Device ID**: 1108 UART 1 ~ UART 8	UART 5 Device Configuration Registers	060h
	UART 6 Device Configuration Registers	068h
	UART 7 Device Configuration Registers	070h
	UART 8 Device Configuration Registers	078h
F81512 **Device ID**: 1112 UART 1 ~ UART 12	UART 9 Device Configuration Registers	080h
	UART 10 Device Configuration Registers	088h
	UART 11 Device Configuration Registers	090h

- Device ID 1104(F81504) only needs to config UART 1(40h) to UART 4(58h)
- Device ID 1108(F81508) only needs to config UART 1(40h) to UART 8(78h)
- Device ID 1112(F81512) only needs to config UART 1(40h) to UART 12(98h)

1.1 UART Register (PCIE Configuration 040h-09Fh)

Only config four (+0/+1/+4/+5) following offset register base on UART I/O address and can let function work.

UART Device Configuration Registers(40h/48h/50h/58h/60h/68h/70h/78h/80h/88h/90h/98h)									
Register 0x[HEX]	Register Name	Default Value MSB							LSB
+0	UART control Register	0	0	-	-	-	-	-	1
+1	FIFO Mode Register	0	0	0	0	0	-	0	0
+4	Base Address High Register	0	0	0	0	0	0	1	0
+5	Base Address Low Register	1	1	1	1	1	0	0	0

UART Control Register — UART Configure Address + 0

Bit	Name	R/W	Default	Description
0	UART_EN	R/W	0	0: disable UART I/O Port. 1: enable UART I/O Port.

FIFO Select Register — UART CONFIGURE ADDRESS + 1

Bit	Name	R/W	Default	Description
5-4	RXFTHR_MODE	R/W	00b	The RX FIFO threshold select. 00: FIFO threshold is set by RXFTHR. 01: FIFO threshold will be 2X of RXFTHR. 10: FIFO threshold will be 4X of RXFTHR. 11: FIFO threshold will be 8X of RXFTHR.
1-0	FIFO_MODE	R/W	00b	Select the FIFO depth. 00: 16-byte FIFO. 01: 32-byte FIFO. 10: 64-byte FIFO. 11: 128-byte FIFO.

Base Address Low Register — UART Configure Address + 4

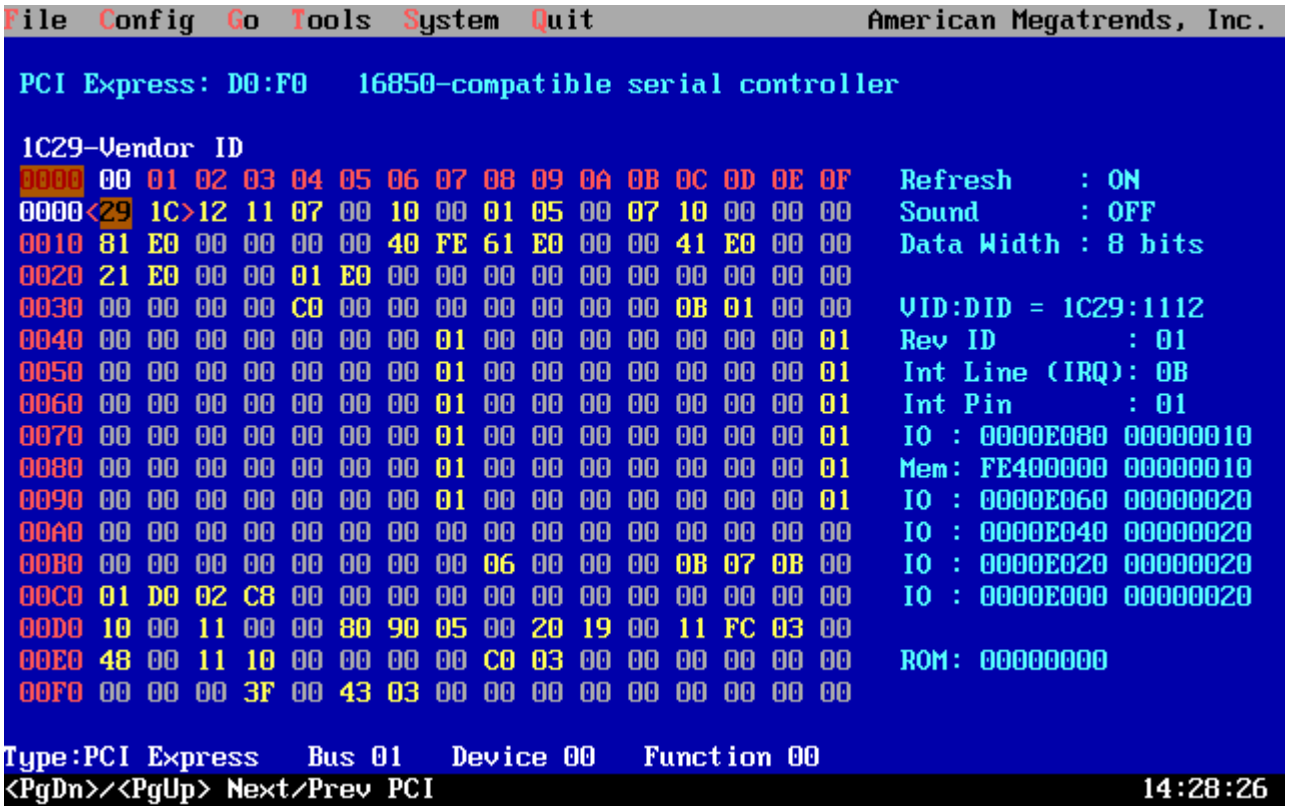
Bit	Name	R/W	Default	Description
7-0	BASE_ADDR_LO	R/W	00h	The LSB of UART ADDRESS.

Base Address High Register — UART Configure Address + 5

Bit	Name	R/W	Default	Description
7-0	BASE_ADDR_HI	R/W	00h	The MSB of UART base address.

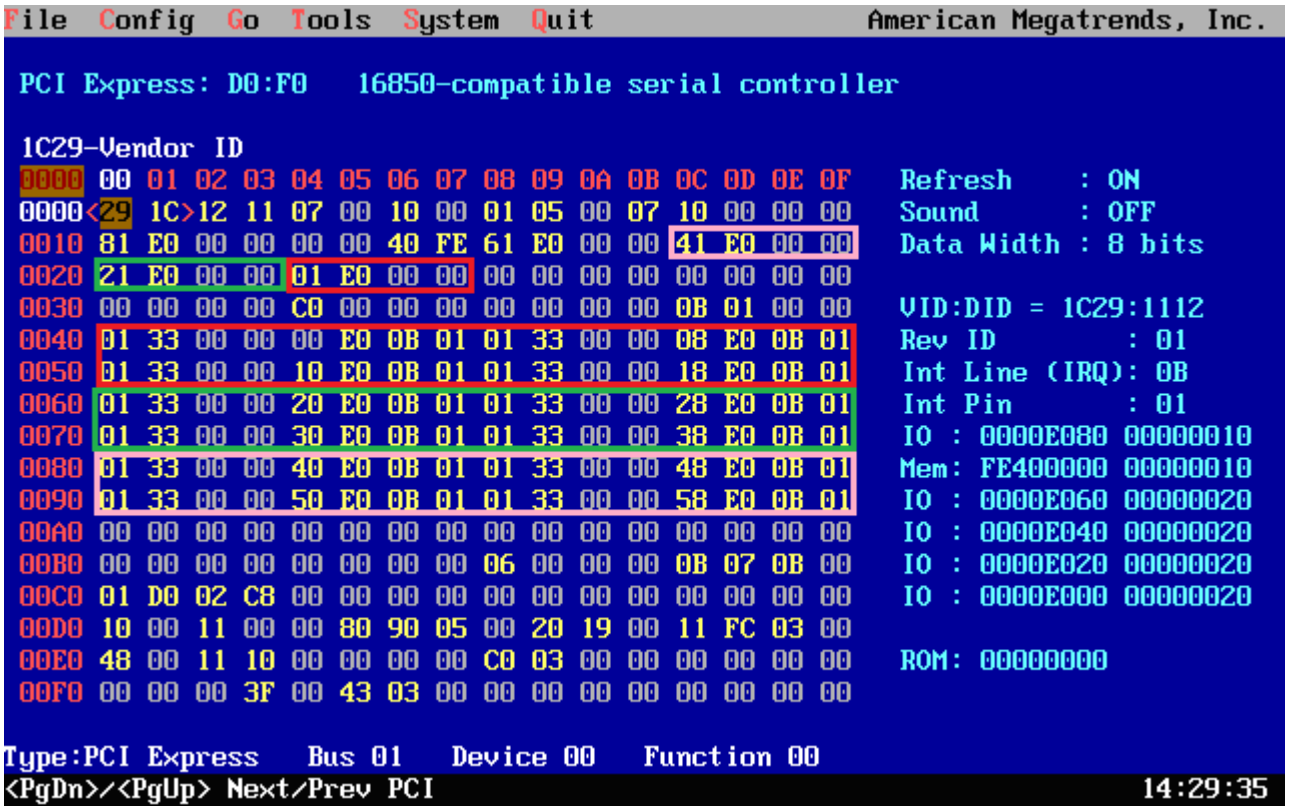
1.2 UART Configuration Example

1.2.1 PCIE to UART Card Initial Status



1.2.2 PCIE to UART Configuration Result

Using BAR 24h address to config UART 1 to UART 4
Using BAR 20h address to config UART 5 to UART 8
Using BAR 1Ch address to config UART 9 to UART 12
See below figure:



1.2.2.1 UART 1 Configuration

I/O Address = 0xE000

Step1. Read 4 bytes of PCIE BAR 24h: 0x0000E001
Step2. Filter BAR 24h bit [4:0]: 0x0000E001 & 0xFFFFFFE0 = 0x0000E000
Step3. Write 0x01 to 40h: enable UART 1 I/O Port
Step4. Write 0x33 to 41h: Select 128-byte FIFO and 8X FIFO threshold
Step5. Write 0x00 to 44h: The LSB of UART 1 ADDRESS
Step6. Write 0xE0 to 45h: The MSB of UART 1 ADDRESS

1.2.2.2 UART 2 Configuration

I/O Address = 0xE008
Step1. Read 4 bytes of PCIE BAR 24h: 0x0000E001
Step2. Filter BAR 24h bit [4:0]: 0x0000E001 & 0xFFFFFFE0 = 0x0000E000
Step3. Write 0x01 to 48h: enable UART 2 I/O Port
Step4. Write 0x33 to 49h: Select 128-byte FIFO and 8X FIFO threshold
Step5. Write 0x08 to 4Ch: The LSB of UART 2 ADDRESS
Step6. Write 0xE0 to 4Dh: The MSB of UART 2 ADDRESS

1.2.2.3 UART 3 Configuration

I/O Address = 0xE010
Step1. Read 4 bytes of PCIE BAR 24h: 0x0000E001
Step2. Filter BAR 24h bit [4:0]: 0x0000E001 & 0xFFFFFFE0 = 0x0000E000
Step3. Write 0x01 to 50h: enable UART 3 I/O Port
Step4. Write 0x33 to 51h: Select 128-byte FIFO and 8X FIFO threshold
Step5. Write 0x10 to 54h: The LSB of UART 3 ADDRESS
Step6. Write 0xE0 to 55h: The MSB of UART 3 ADDRESS

1.2.2.4 UART 4 Configuration

I/O Address = 0xE018
Step1. Read 4 bytes of PCIE BAR 24h: 0x0000E001
Step2. Filter BAR 24h bit [4:0]: 0x0000E001 & 0xFFFFFFE0 = 0x0000E000
Step3. Write 0x01 to 58h: enable UART 4 I/O Port
Step4. Write 0x33 to 59h: Select 128-byte FIFO and 8X FIFO threshold
Step5. Write 0x18 to 5Ch: The LSB of UART 4 ADDRESS
Step6. Write 0xE0 to 5Dh: The MSB of UART 4 ADDRESS

1.2.2.5 UART 5 Configuration

I/O Address = 0xE020
Step1. Read 4 bytes of PCIE BAR 20h: 0x0000E021
Step2. Filter BAR 20h bit [4:0]: 0x0000E021 & 0xFFFFFFE0 = 0x0000E020

Step3. Write 0x01 to 60h: enable UART 5 I/O Port
Step4. Write 0x33 to 61h: Select 128-byte FIFO and 8X FIFO threshold
Step5. Write 0x20 to 64h: The LSB of UART 5 ADDRESS
Step6. Write 0xE0 to 65h: The MSB of UART 5 ADDRESS

1.2.2.6 UART 6 Configuration

I/O Address = 0xE028
Step1. Read 4 bytes of PCIE BAR 20h: 0x0000E021
Step2. Filter BAR 20h bit [4:0]: 0x0000E021 & 0xFFFFFFE0 = 0x0000E020
Step3. Write 0x01 to 68h: enable UART 6 I/O Port
Step4. Write 0x33 to 69h: Select 128-byte FIFO and 8X FIFO threshold
Step5. Write 0x28 to 6Ch: The LSB of UART 6 ADDRESS
Step6. Write 0xE0 to 6Dh: The MSB of UART 6 ADDRESS

1.2.2.7 UART 7 Configuration

I/O Address = 0xE030
Step1. Read 4 bytes of PCIE BAR 20h: 0x0000E021
Step2. Filter BAR 20h bit [4:0]: 0x0000E021 & 0xFFFFFFE0 = 0x0000E020
Step3. Write 0x01 to 70h: enable UART 7 I/O Port
Step4. Write 0x33 to 71h: Select 128-byte FIFO and 8X FIFO threshold
Step5. Write 0x30 to 74h: The LSB of UART 7 ADDRESS
Step6. Write 0xE0 to 75h: The MSB of UART 7 ADDRESS

1.2.2.8 UART 8 Configuration

I/O Address = 0xE038
Step1. Read 4 bytes of PCIE BAR 20h: 0x0000E021
Step2. Filter BAR 20h bit [4:0]: 0x0000E021 & 0xFFFFFFE0 = 0x0000E020
Step3. Write 0x01 to 78h: enable UART 8 I/O Port
Step4. Write 0x33 to 79h: Select 128-byte FIFO and 8X FIFO threshold
Step5. Write 0x38 to 7Ch: The LSB of UART 8 ADDRESS
Step6. Write 0xE0 to 7Dh: The MSB of UART 8 ADDRESS

1.2.2.9 UART 9 Configuration

I/O Address = 0xE040
Step1. Read 4 bytes of PCIE BAR 40h: 0x0000E041
Step2. Filter BAR 1Ch bit [4:0]: 0x0000E041 & 0xFFFFFFE0 = 0x0000E040
Step3. Write 0x01 to 80h: enable UART 9 I/O Port
Step4. Write 0x33 to 81h: Select 128-byte FIFO and 8X FIFO threshold
Step5. Write 0x40 to 84h: The LSB of UART 9 ADDRESS
Step6. Write 0xE0 to 85h: The MSB of UART 9 ADDRESS

1.2.2.10 UART 10 Configuration

I/O Address = 0xE048
Step1. Read 4 bytes of PCIE BAR 40h: 0x0000E041
Step2. Filter BAR 1Ch bit [4:0]: 0x0000E041 & 0xFFFFFFE0 = 0x0000E040
Step3. Write 0x01 to 88h: enable UART 10 I/O Port
Step4. Write 0x33 to 89h: Select 128-byte FIFO and 8X FIFO threshold
Step5. Write 0x48 to 8Ch: The LSB of UART 10 ADDRESS
Step6. Write 0xE0 to 8Dh: The MSB of UART 10 ADDRESS

1.2.2.11 UART 11 Configuration

I/O Address = 0xE050
Step1. Read 4 bytes of PCIE BAR 40h: 0x0000E041
Step2. Filter BAR 1Ch bit [4:0]: 0x0000E041 & 0xFFFFFFE0 = 0x0000E040
Step3. Write 0x01 to 90h: enable UART 11 I/O Port
Step4. Write 0x33 to 91h: Select 128-byte FIFO and 8X FIFO threshold
Step5. Write 0x50 to 94h: The LSB of UART 11 ADDRESS
Step6. Write 0xE0 to 95h: The MSB of UART 11 ADDRESS

1.2.2.12 UART 12 Configuration

I/O Address = 0xE058
Step1. Read 4 bytes of PCIE BAR 40h: 0x0000E041
Step2. Filter BAR 1Ch bit [4:0]: 0x0000E041 & 0xFFFFFFE0 = 0x0000E040

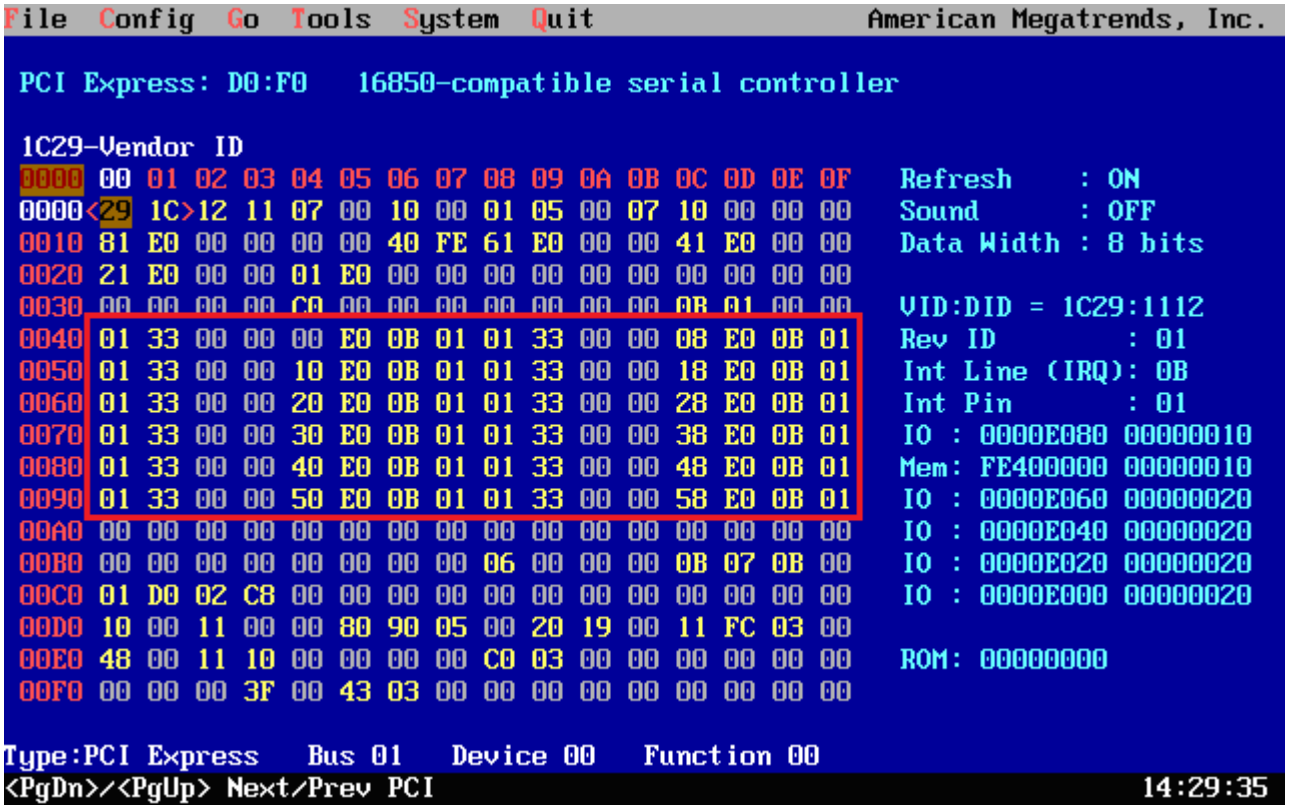
Step3. Write 0x01 to 98h: enable UART 12 I/O Port

Step4. Write 0x33 to 99h: Select 128-byte FIFO and 8X FIFO threshold

Step5. Write 0x58 to 9Ch: The LSB of UART 12 ADDRESS

Step6. Write 0xE0 to 9Dh: The MSB of UART 12 ADDRESS

The picture below is CONFIG result:



2. WDT Function

For example PFA=0xF0000000 NOTE 1, WDT Base address = 0xE000, set WDT INT= IRQ5 10s issue interrupt

2.1 WDT Configuration Register

```
write (PFA+0xB8, 0x01);      // enable SIRQ mode
                               // pin IRQSEL pull low and EN_WDT pull high
write (PFA+0xA0, 0x01);      // enable WDT base address
write (PFA+0xA1, 0x00);      // The LSB of WDT base address
write (PFA+0xA2, 0xE0);      // The MSB of WDT base address
```

2.2 WDT Timer IO function

```
write (WDT Base address+0x00, 0x01); // enable WDT INT and set IRQ 5 for WDT
write (WDT Base address+0x01, 0x02); // set WDT Timer unit = 1 sec
write (WDT Base address+0x02, 0x0A); // set WDT Timer count = 10
write (WDT Base address+0x02, 0x0A); // set WDT Timer count = 10
```

PS：write the same non-zero value twice to enable the timer.

3. EEPROM Memory Function

For example PFA=0xF0000000, EEPROM Base address = 0xF4100000, READ DEVICE ID from EEPROM

3.1 EEPROM Configuration Register

```
write (PFA+0xB4, 0x00);        // EEPROM base address [7:0]
write (PFA+0xB5, 0x00);        // EEPROM base address [15:8]
write (PFA+0xB6, 0x10);        // EEPROM base address [23:16]
write (PFA+0xB7, 0xF4);        // EEPROM base address [31:24]
```

3.2 EEPROM Memory function

```
write (Base address+0x06, 0x01);   // enable WR_DBI and WR_EEPROM command
write (Base address+0x03, 0x00);   // DBI_ADDR
write (Base address+0x00, 0x10);   // start read single byte data from DBI interface
read (Base address+0x04)                 // read LSB of DEVICE ID
write (Base address+0x03, 0x01);   // DBI_ADDR
write (Base address+0x00, 0x10);   // start read single byte data from DBI interface
read (Base address+0x04)                 // read MSB of DEVICE ID
```

4. GPIO Function

4.1 GPIO0 Function

For example, PFA=0xF0000000, GPIO Base address = 0xE060, set GPIO1 for input

4.1.1 GPIO0 Configuration Address Registers

write (PFA+0xF0, 0x01); //Enable GPIO0 I/O port
write (PFA+0xF1, 0x60); //GPIO Base Address Low Register
write (PFA+0xF2, 0xE0); //GPIO Base Address High Register
write (PFA+0xF3, 0x00); //set pin 95-102 for GPIO
write (PFA+0xF8, 0x00); //set GPIO00-GPIO07 are input

4.1.2 GPIO0 IO Function

read (GPIO Base address+0x00); //read input data from GPIO0 data register

4.2 GPIO1 Function

For example, PFA=0xF0000000, GPIO Base address = 0xE060, set GPIO1 for output(open drain)

4.2.1 GPIO1 Configuration Address Registers

write (PFA+0xF0, 0x02); //Enable GPIO1 I/O port
write (PFA+0xF1, 0x60); //GPIO Base Address Low Register
write (PFA+0xF2, 0xE0); //GPIO Base Address High Register
write (PFA+0xF3, 0x00); //set pin 112-119 for GPIO
write (PFA+0x100, 0xFF); //set GPIO00-GPIO07 are output
write (PFA+0x101, 0x00); //set GPIO00-GPIO07 are open drain in output mode

4.2.2 GPIO1 IO Function

write (GPIO Base address+0x01, 0x55);//write '0x55' to GPIO1 data register

4.3 GPIO2 Function

For example, PFA=0xF0000000, GPIO Base address = 0xE060, set GPIO2 for output(push pull)

4.3.1 GPIO2 Configuration Address Registers

write (PFA+0xF0, 0x04); //Enable GPIO1 I/O port
write (PFA+0xF1, 0x60); //GPIO Base Address Low Register
write (PFA+0xF2, 0xE0); //GPIO Base Address High Register
write (PFA+0xF3, 0x00); //set pin 112-119 for GPIO
write (PFA+0x108, 0xFF); //set GPIO00-GPIO07 are output
write (PFA+0x109, 0xFF); //set GPIO00-GPIO07 are push pull in output mode

4.3.2 GPIO2 IO Function

write (GPIO Base address+0x01, 0xAA); //write '0xAA' to GPIO2 data register

NOTE 1：PFA (PCI Function Address)=Bus (8 Bits) / Device/Function (5 + 3 Bits)/ Register (12 Bits)+PCI-E Base Address (MMIO) total 32 Bits。

Example 1：Bus = 0x01, Device = 0x1F, Function: 0x02, (PCI-E Base Address：0xF0000000)

Read Register 0x100.
PFA = 0xF0000000 + (0x01FA100) = 0xF01FA100