

Review 1  
On GPP I/O Interface  
CMPE242  
S2018

1. Design by drawing schematics for input and output testing circuit for GPP input and output debugging.

(1.1) The output testing circuit with its input connected to the GPP port output, identify the GPP port of your choice, for example GPE3, find its connector name, and the pin number on the connector. The output testing circuit consists of 1 resistor to regulate the current to 8 mA, and 1 red LED to function as a visual indicator to be on and off per GPP output 1 or 0 accordingly. (You can take a look at the PPT from the lecture which gives a reference design for Jetson Tx1 board from Nvidia, however there is no need to use transistor. So use the simpler design). Be sure to calculate the resistance value.

(1.2) The input testing circuit with its output connected to the GPP port input, identify the GPP port of your choice, for example GPE2, find its connector name, and the pin number on the connector. The input testing circuit consists of 2 resistors to regulate the current to 8 mA, and 1 switch (SW) to toggle between input Vcc (3.3 VDC) and GND. Calculate the resistance value for each resistor in your design. (Note resistor R1 functions as regulator when SW is connected to Vcc, its resistance value can be calculated based on the preventive design, e.g., when GPP port as input, however, it is accidentally configured as output and set to “0”. Similarly, resistor R2 is calculated based when SW is toggled to GND, its value should be set to such to regulate current to 8 mA when GPP port accidentally set as output 1.

2. Write a c/c++ user application program to high light the use of “fd = open( );” to define file descriptor of the GPP port device driver program, and “ioctl( );” to realize input and output testing of the GPP design with GPP pins of your choice, for example, realize the GPP input and output debugging function with GPE2 as input and GPE3 as output.

3. Based on ARM11 CPU data-sheet answer the following questions:

(3.1) find the total number of GPP ports (see Figure 1), and identify 3 special purpose registers, GPxCON, DATCON, and GPxPUD, their addresses, location on the memory map, and the bit pattern definition for init and config for GPxCON table for the design of your choice, for example, GPE port (see Figure 2.)

S3C6410 includes 187 multi-functional input/output port pins. There are 17 ports as listed below:

PortName	Number of Pins.	Muxed pins	Power Inform.
GPA port	8	UART/EINT	1.8~3.3V
GPB port	7	UART/IrDA/I2C/CF/Ext.DMA/EINT	1.8~3.3V
GPC port	8	SPI/SDMMC/I2S_V40/EINT	1.8~3.3V
GPD port	5	PCM/I2S/AC97/EINT	1.8~3.3V
GPE port	5	PCM/I2S/AC97	1.8~3.3V
GPF port	16	CAMIF/PWM/EINT	1.8~3.3V
GPG port	7	SDMMC/EINT	1.8~3.3V
GPH port	10	SDMMC/KEYPAD/CF/I2S_V40/EINT	1.8~3.3V
GPI port	16	LCD	1.8~3.3V
GPJ port	12	LCD	1.8~3.3V
GPK port	16	HostIF/HIS/KEYPAD/CF	1.8~3.3V
GPL port	15	HostIF/KEYPAD/CF/OTG/EINT	1.8~3.3V
GPM port	6	HostIF/CF/EINT	1.8~3.3V
GPN port	16	EINT/KEYPAD	1.8~3.3V
GPO port	16	MemoryPort0/EINT	1.8~3.3V
GPP port	15	MemoryPort0/EINT	1.8~3.3V
GPQ port	9	MemoryPort0/EINT	1.8~3.3V

Figure 1. GPP ports (total 187 pins).

GPDCON	Bit	Description	
GPE0	[3:0]	0000 = Input 0010 = PCM SCLK[1] 0100 = AC97 BITCLK 0110 = Reserved	0001 = Output 0011 = I2S CLK[1] 0101 = Reserved 0111 = Reserved
GPE1	[7:4]	0000 = Input 0010 = PCM EXTCLK[1] 0100 = AC97 RESETn 0110 = Reserved	0001 = Output 0011 = I2S CDCLK[1] 0101 = Reserved 0111 = Reserved
GPE2	[11:8]	0000 = Input 0010 = PCM FSYNC[1] 0100 = AC97 SYNC 0110 = Reserved	0001 = Output 0011 = I2S LRCLK[1] 0101 = Reserved 0111 = Reserved
GPE3	[15:12]	0000 = Input 0010 = PCM SIN[1] 0100 = AC97 SDI 0110 = Reserved	0001 = Output 0011 = I2S DI[1] 0101 = Reserved 0111 = Reserved
GPE4	[19:16]	0000 = Input 0010 = PCM SOUT[1] 0100 = AC97 SDO 0110 = Reserved	0001 = Output 0011 = I2S DO[1] 0101 = Reserved 0111 = Reserved

Figure 2. GPECON control register bit settings (partial, for whole table see the CPU data sheet).

(3.2) Find binary pattern (32 bits) to set GPE2 as input and GPE3 as output. Write this binary pattern as hex value.

(3.3) At /drivers/char directory, to identify the GPx device driver, find the init module, use this module as reference to write your GPx device driver, to allow your choice of the design, for example, GPE2 as input and GPE3 as output for GPP interface testing.

(3.4) Compile and build user application program and GPP device driver program, copy both programs to the ARM development board, and run both programs.

(End)