Review Questions (GPIO)

Week 2

Review Questions

1. Determine if the following statements are TRUE or FALSE:
   a) Each GPIO port can have up to 16 port pins. **[FALSE]**
   b) GPIO port pins can be individually configured to be Input or Output pins. **[TRUE]**
   c) The maximum toggle speed of a GPIO pin when connected to the APB bus is the system bus clock speed. **[FALSE]**
   d) The default drive strength of a GPIO port pin is 2mA. **[TRUE]**
   e) The GPIO DEN register needs to be set to '1' to configure a port pin to be an Analog function. **[FALSE]**
   f) Setting the corresponding bit in the GPIOAMSEL register enables the analog function for that pin (if there is a valid a mapping). **[TRUE]**
   g) Analog function is an Alternate function. **[FALSE]**
   h) GPIO port pins with 'Special Considerations' are set to their Alternate functions upon power up. **[TRUE]**
2. I would like to map port PE0 to function as an alternate function as UART7 Rx. What value should be written to the GPIOPCTL register?
   - To map to UART0 Rx, GPIOE->PCTL |= 0x01 << 0.
3. Why are port pins PC0 to PC3 not available on the Tiva LaunchPad?
   - PC0 to PC3 are already assigned to be JTAG/SWD pins for debug purposes.

Review Questions

4. What is the GPIODEN register used for? What is its default function upon power up?
   - GPIODEN is to enable the digital function for GPIO port pins. By default, GPIODEN is all GPIO pins are not driven (in tristate) when out of reset – that is, their digital function is disabled.
5. GPIO pin drive strengths can be set through the GPIODR2R (for 2mA), GPIODR4R or GPIODR8R. What is the drive strength for a GPIO pin if we write to both GPIODR4R and GPIOR8R consecutively during initialization?
   - Writing to the GPIODR8R register will automatically reset the corresponding bits in GPIODR4R.
6. How would you check if a GPIO port is ready to be accessed after initialization?
   - Check the PRGPIO (GPIO Peripheral Ready) register. It is one of the system control registers.
   - If the corresponding port bit is set, the GPIO port register are ready for access. For example, if bit 0 of PRGPIO register is set, GPIO Port A registers are ready to be accessed.

Review Questions

7. Write a C function **Port_Init()** to initialize Port E, bit 4 as Input & Port B, pin 0 as Output.

```
#define BIT( x )        (1U<<(x))     /* defined in common.h   */
#define PB_0            0U            /* bit 0 */
#define PE_4            4U            /* bit 4 */

void Port_Init( void )
{
   /* enable clocks to GPIO ports B & E   */
    SYSCTL->RCGCGPIO |= SYSCTL_RCGCGPIO_R1 | SYSCTL_RCGCGPIO_R4;
    /* Wait for GPIOB & GPIOE to be ready */
   while( 0 == (SYSCTL->PRGPIO & SYSCTL_PRGPIO_R1)){};
   while( 0 == (SYSCTL->PRGPIO & SYSCTL_PRGPIO_R4)){};

   GPIOB->DIR |= BIT(PB_0);  /* output */
   GPIOB->DEN |= BIT(PB_0);  /* enable digital function */
   GPIOB->AFSEL &= ~BIT(PB_0);

   GPIOE->DIR &= ~BIT(PE_4); /* input  */
   GPIOE->DEN |= BIT(PE_4);  /* enable digital function */
   GPIOE->AFSEL &= ~BIT(PE_4);

   /** insert other initialization codes **/
}
```

Review Questions

8. List the GPIO port pins that do not default to be GPIO pins?
   - GPIO pins that are labelled as pins with 'special considerations' do not default to GPIO function. The pins are listed below:

Table 10-1. GPIO Pins With Special Considerations

GPIO Pins	Default Reset State	GPIOAFSEL	GPIODEN	GPIOPDR	GPIOPUR	GPIOPCTL	GPIOCR
PA[1:0]	UART0	0	0	0	0	0x1	1
PA[5:2]	SSI0	0	0	0	0	0x2	1
PB[3:2]	I^2C0	0	0	0	0	0x3	1
PC[3:0]	JTAG/SWD	1	1	0	1	0x1	0

Pins do not default to digital functions.

9. Which are the GPIO pins that requires us to unlock and commit before it's configuration can be change?
   - PF0, PD7, PC[3:0].
10. List the GPIO port registers that requires an 'unlock-commit' sequence before access.
   - GPIOAFSEL, GPIOPUR, GPIOPDR & GPIODEN registers.

Review Questions

11. I would like to use UART7 TX and RX in my design. How should I configure the port pins in my program? Consider the DEN, AMSEL, PCTL,, AFSEL, … registers.
    - UART 7 pins are: U7Rx at PE0 and U7Tx at PE1.

```
#define BIT( x )  (1U<<(x))    /* defined in common.h   */
#define PE_0      0U           /* U7 Rx  */
#define PE_1      1U           /* U7 Tx  */

void Port_Init( void )
{
   SYSCTL->RCGCGPIO |= SYSCTL_RCGCGPIO_R4; /* enable GPIOE clock */
   /* Wait for GPIOE to be ready */
   while( 0 == (SYSCTL->PRGPIO & SYSCTL_PRGPIO_R4)){};

   GPIOE->DIR |= BIT(PE_1);         /* PE1 = output  */
   GPIOE->DIR &= ~( BIT(PE_0) );    /* PE0 = input   */
   GPIOE->DEN |= BIT(PE_0) | BIT(PE_1); /* digital enable */
   GPIOE->AFSEL |= BIT(PE_0) | BIT(PE_1);
   GPIOE->PCTL |= 0x01<<0 | 0x01<<4;
   GPIOE->AMSEL &= ~( BIT(PE_0) | BIT(PE_1) );
   /** add other initialization statements **/
 }
```

Review Questions

12. I would like to write to only bits 6, 3 and 1 of GPIOB register. Using address mask, which offset address should I write to?
    - Address mask = 0x128.

Bit:	9	8	7	6	5	4	3	2	1	0
ADDR MASK (0b)	0	1	0	0	1	0	1	0	0	0
ADDR MASK (0x)	1		2				8			

Review Questions

13. I want to read only bits 5 and 4 of a GPIODATA register through a Read address mask. The current GPIODATA is 0xAA. What is the data read using an address mask?
    - Data Read = 0b0010.0000.

Bit:	9	8	7	6	5	4	3	2	1	0	
ADDR MASK (0b)	0	0	**1**	**1**	0	0	0	0	0	0	
ADDR MASK (0x)	0		C				0				
Bit:	7	6	5	4	3	2	1	0			
GPIO->DATA	1	0	1	0	1	0	1	0	CURRENT DATA		
Bit:	7	6	5	4	3	2	1	0			
RETURNED VALUE	0	0	**1**	**0**	0	0	0	0			

Review Questions

14. What is the GPIOLOCK & GPIOCR used for?
    - GPIOLOCK and GPIOCR registers enable write access to the GPIO port registers.
    - Writing a special Pattern of 0x4C4F.434B will unlock the GPIOLOCK register.
        - Writing any other value will re-enable the lock state.
    - GPIOCR register determines which bits of the GPIOAFSEL, GPIOPUR, GPIOPDR and GPIODEN registers are committed. A bit value of '1' allows the port bit to be committed.
    - This set of registers is designed to prevent accidental programming of certain critical registers, for example, registers that control connectivity to the NMI interrupt function & JTAG debug hardware.
15. A GPIO port can be accessed through either the APB and AHB bus. Which register is used to select the bus connection?
    - GPIO High-Performance Bus Control register (GPIOHBCTL).

GPIO High-Performance Bus Control (GPIOHBCTL)

Base 0x400F.E000
Offset 0x06C
Type RW, reset 0x0000.7E00

	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	reserved															
Type	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	reserved										PORTF	PORTE	PORTD	PORTC	PORTB	PORTA
Type	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RW	RW	RW	RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit=1 : AHB bus
Bit=0 : APB bus