Q1 Timer Limitations
4 Points

For Timer_A in `TIMER_A_UP_MODE` using SMCLK: If SMCLK was set to **6 MHz** and `TIMER_A_CLOCKSOURCE_DIVIDER_28`, what is the *maximum* achievable timer overflow/reset period? Answer in **seconds** or ms, µs, ns, etc.

Reminder: SMCLK is 6 MHz

Maximum:

0.310

Similarly, what is the *minimum non-zero* timer overflow/reset period? Answer in **seconds** or ms, µs, ns, etc.

Minimum:

0.0000046s

Q2
3 Points

Convert the following pseudocode to actual C code.

> Create all needed variables as `int16_t`
> Set the value of variable `a` to 2, `b` to 2
> Set `c` to the last two digits of your RIN
> Loop the number of times given in `c`:
> if `b` is larger than `a`: multiply `a` by `b`, save into `a`
> otherwise: subtract `b` from `a`, save into `a`
> increment `b`
> Double the value of `a` and save into `b`

```
int16_t a = 2, b = 2, c = 37;
int16_t i = 0;
for(i, i < c, i++){
   if (b > a){
      a *=b ;
   } else {
      a -= b;
   }
   b++;
}
b *= (2*a);
```

Q3 GPIO Usage
3 Points

Write a segment of code that will change the value of an output pin P4.3 to be the logical NOR (Not-OR) of the current values of inputs P5.7 and P2.1. You can use either registers or the DriverLib. Assume initializations are already done.

```
val1 = (P5IN & 0x80) != 0;
val2 = (P2IN & 0x01) != 0;
NOR_value = ~(val1 || val2);
if (NOR_value){
   P4OUT |= 0x04;
} else {
   P4OUT &= ~0x04;
}
```

Q4 Timers Setup
10 Points

Given the complete program below and knowing that SMCLK is **6 MHz**, answer the following questions.

```
void TimerInit();

void main(){
```

```
  SysInit();
  TimerInit();
  uint32_t j;
  while(1){
    __delay_cycles(1e6);
    printf("%u\r\n",j++);
  }
}

void TimerInit(){
  Timer_A_UpModeConfig tim_config;
  uint32_t timer_base = TIMER_A1_BASE;
  tim_config.clockSource = TIMER_A_CLOCKSOURCE_SMCLK;
  tim_config.clockSourceDivider = TIMER_A_CLOCKSOURCE_DIVIDER_10;
  tim_config.timerPeriod = 50000;
  tim_config.timerClear = TIMER_A_DO_CLEAR;
  Timer_A_configureUpMode(timer_base,&tim_config);
  Timer_A_startCounter(timer_base,TIMER_A_UP_MODE);
}
```

Q4.1
3 Points

How fast does the configured *timer count register* increment; that is, what is the timer counting frequency? This does NOT refer to the timer overflow/reset frequency. Answer in **Hz**.

Reminder: SMCLK is 6 MHz

600000 Hz

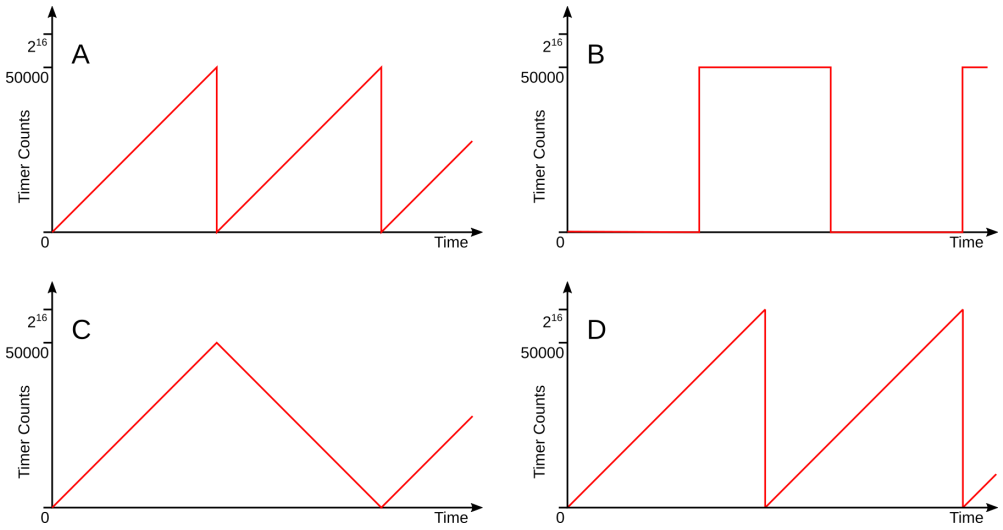
Q4.2
4 Points

What is the configured timer's overflow (reset) period in seconds?

0.083 seconds

Q4.3
2 Points

Select the figure that most closely resembles the timer operation as configured.



Enter the correct figure letter:

A

Q4.4
1 Point

How often does the `printf("%u\r\n",j++);` print?

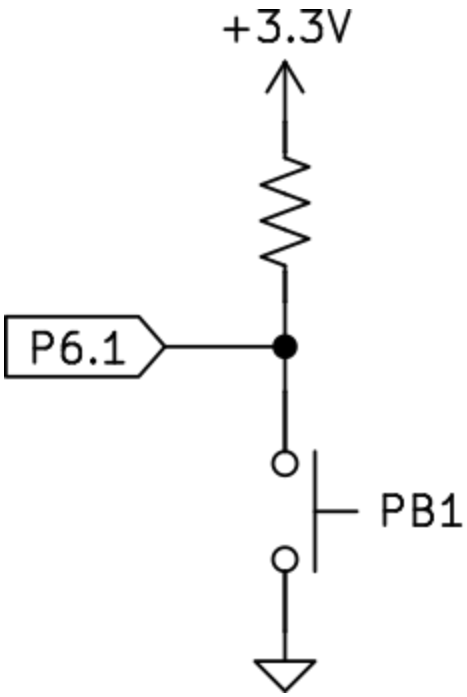
Q5 Basic GPIO
4 Points

Answer the following questions about GPIO functionality and usage.

Q5.1
2 Points

A pushbutton is wired as shown to P6.1 and configured as a normal input. What is the value of variable `res` after the

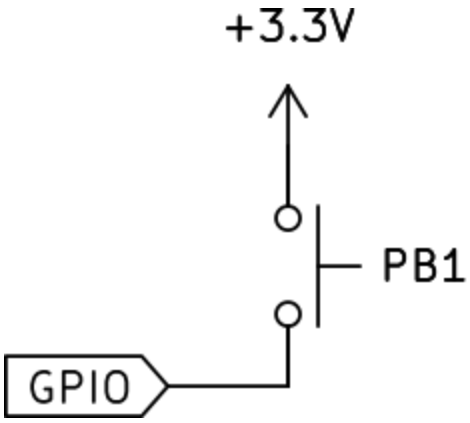
command below is run if the pushbutton is **pressed**.




```
uint8_t res = GPIO_getInputPinValue(GPIO_PORT_P6,GPIO_PIN1);
```

the value is 0

Q5.2
2 Points



A GPIO is monitoring PB1 as shown in the circuit above. Assuming the pin is P5.3, what function is the appropriate initialization, and what are the function inputs needed?

`GPIO_toggleOutputOnPin(...);`

`GPIO_setAsInputPinWithPullDownResistor(...);`

`GPIO_setAsOutputPin(...);`

`GPIO_setAsInputPinWithPullUpResistor(...);`

`GPIO_setAsInputPin(...);`

`GPIO_getInputPinValue(...);`

Function inputs / arguments. Give as comma separated list (as you would the function).

GPIO_setAsInputWithPullDownResistor(GPIO_PORT_5, GPIO_PIN3);

Q6 GPIO Configuration
6 Points

Q6.1
3 Points

Given the register configuration for Port 3 given below, list what pins are known to be inputs or outputs, or are unknown.

P3DIR &= ~0x29;
P3DIR |= 0xB2;

Inputs:

Pin: 0, 3 are inputs.

Outputs:

Pin: 1, 4, 5, 6 are outputs.

Unknown:

Pin: 2, 6 are unknown.

Q6.2
3 Points

Write code using the DriverLib to initialize the pins given below. No pull Up/Down resistors are required.

Outputs: P2.3, P2.5
Inputs: P2.1, P2.4, P1.2

```
GPIO_setAsInputPin(GPIO_PORT_2, GPIO_PIN1 | GPIO_PIN4);
GPIO_setAsInputPin(GPIO_PORT_1, GPIO_PIN2);
GPIO_setAsOutputPin(GPIO_PORT_2, GPIO_PIN3 |
GPIO_PIN5);
```

Quiz 2

● Graded

Select each question to review feedback and grading details.

Student
Ryan So

Total Points
27.5 / 30 pts

Question 1
Timer Limitations — **4** / 4 pts

Question 2

(no title) **2** / 3 pts

Question 3

GPIO Usage **3** / 3 pts

Question 4

Timers Setup **9** / 10 pts

4.1 (no title) **3** / 3 pts

4.2 (no title) **4** / 4 pts

4.3 (no title) **2** / 2 pts

4.4 (no title) **0** / 1 pt

Question 5

Basic GPIO **4** / 4 pts

5.1 (no title) **2** / 2 pts

5.2 (no title) **2** / 2 pts

Question 6

GPIO Configuration **5.5** / 6 pts

6.1 (no title) **2.5** / 3 pts

6.2 (no title) **3** / 3 pts