

# Operating System Final Project Checkpoint 5

Student ID : 210510232 莊凱予

1. Explain how you would implement the delay function in relation to the timer used for preemption. By default, we set it up for a 13-bit timer for the quantum. Considering that each thread gets its own delay call independently:

(1) What does your timer-0 ISR have to do to support these multiple delays and now()?

```
unsigned char now( void ) {  
    return time;  
}
```

I use unsigned char type function to return the current time.

```
void delay(unsigned char p) {  
    Delay_Time[ID] = now() + p;  
    Bitmap[ID] = -2;  
    ThreadYield();  
}
```

I use an array "Delay\_Time[MAXTHREADS]" to record each thread delay time units.

Unsigned char p represent delay time units. Then, I will set Bitmap[ID] which thread ID delay value to -2. And then, I will call ThreadYield() to change to another thread that can excute.

```
void myTimer0Handler(void)  
{  
    counter = (counter==4) ? 0 : counter+1;  
    if(!counter) time++;  
}
```

myTimer0Handler() will update time if interrupt happened four times.

If all thread call delay() and happen to finish their delays all at the same time?

In the worst case, thread which id is equal to zero can not use. Then, thread id equal to one holds CPU until first cycle ends. After thread1's cycle completed, program will context-switch to thread2. Before context-switch to thread3, thread two will excute like thread1 until second cycle ends. So I can make sure that the accuracy of my delay between n and n+0.5 time units.

## (2) Robust thread termination and creation

```
mov a,DPL
mov b,DPH
mov dptr,#_ThreadExit
push DPL
push DPH
push a
push b
```

I push the return address of ThreadExit() on the stack just like ppc5 slide request.

```
void Bootstrap(void) {
    SemaphoreCreate(thread, 4);
    /*****/
    ThreadID ThreadCreate(FunctionPtr fp) {
        SemaphoreWait( thread );
        /*****/
    void ThreadExit(void) {
        SemaphoreSignal( thread );
    }
}
```

I use SemaphoreCreate(thread, 4) to limit the max thread is 4. In addition to this, thread\_manager occupys thread0 forever. Everytime I calls TreadCreate() that it will call SemaphoreWait() to request one thread if thread is enough otherwise lock. Everytime I calls ThreadExit that it will call SemaphoreSignal() to relase one thread.

## (3) Parking lot example

The below memory dump of mine table to reflect the content of the log.

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
00	2F	27	00	00	00	00	00	00	25	2A	00	00	00	00	00	30
10	26	6D	00	00	00	00	00	06	27	4B	00	00	00	00	FF	06
20	48	58	68	78	01	00	00	00	00	51	00	04	86	00	03	01
30	01	02	81	83	02	03	03	06	83	85	00	01	00	00	00	00
40	04	07	10	05	01	80	00	00	01	00	00	07	85	05	0A	00
50	04	07	3B	06	FF	02	92	02	08	01	08	0A	00	00	00	00
60	04	07	D1	01	26	00	03	00	11	00	00	00	00	06	05	00
70	04	07	26	04	31	01	00	00	19	00	00	00	00	00	00	00

I set my log[10] memory at (0x30 to 0x39)

Every Log's value indicate its delayed time units.

Log which index is even number stands for parking.

Log which index is odd number stands for exiting.

Plot[2] represents the plot1 and the plot2 of the Car.

```
log[0] = now();
SemaphoreWait( mutex );
if ( !plot[0] ) {
    plot[0] = 1;
    __asm
        mov r0, #(_log + 0x0000)
        mov a, @r0
        mov r7, a
        mov a, #0x7F
        and a, r7
        mov @r0, a
    __endasm;
} else {
    plot[1] = 1;
    __asm
        mov r0, #(_log + 0x0000)
        mov a, @r0
        mov r7, a
        mov a, #0x80
        orl a, r7
        mov @r0, a
    __endasm;
}
SemaphoreSignal( mutex );
```

This is for Car1 partial code.

If Car parked to plot[0], the value of the log performs the “and” logical operation with 0x7F.

If Car parked to plot[1], the value of the log performs the “or” logical operation with 0x80.

Above explanation means that value is larger than 0x80 parked at plot[1], otherwise parked at plot[0]. So in the DataMemory show each log's situations, I explain it just like below.

My delayed design :

After Car1 parking, then delay 1 time unit.

After Car2 parking, then delay 2 time unit.

After Car3 parking, then delay 1 time unit.

After Car5 parking, then delay 2 time unit.

Log[0]	Log[1]	Log[2]	Log[3]	Log[4]
01	02	81	83	02
Log[5]	Log[6]	Log[7]	Log[8]	Log[9]
03	03	06	83	85

Car1 parks its car to the plot[0] at time 1

Car2 parks its car to the plot[1] at time 1

Car3 parks its car to the plot[0] at time 2

Car4 parks its car to the plot[0] at time 3

Car5 parks its car to the plot[1] at time 3

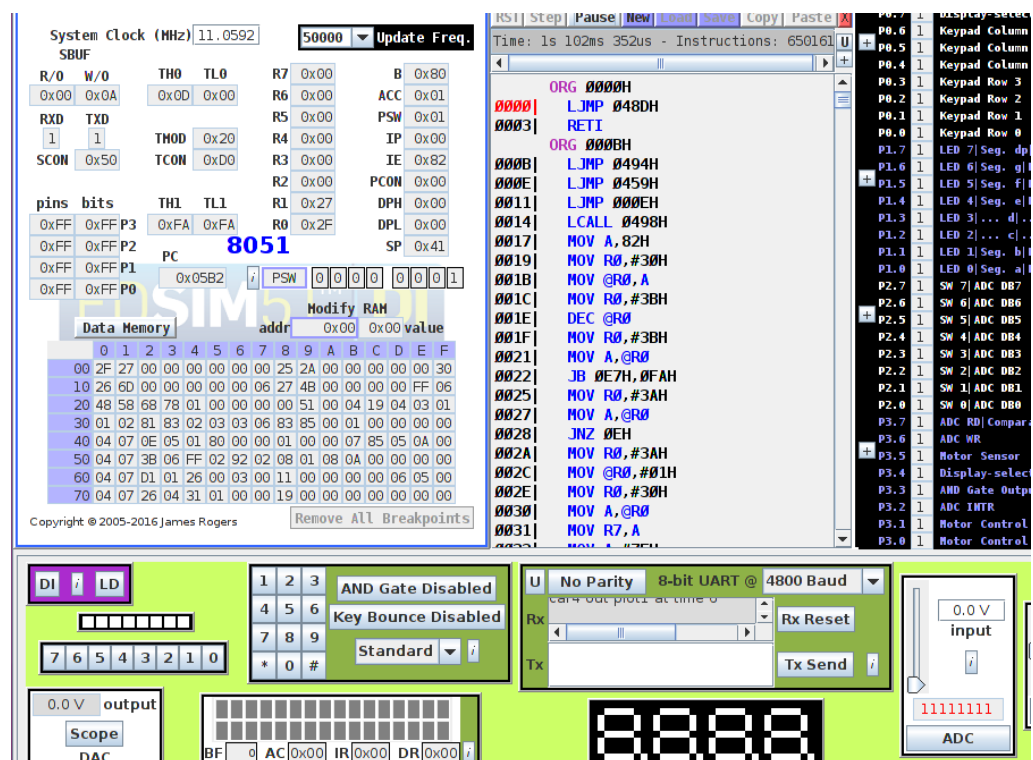
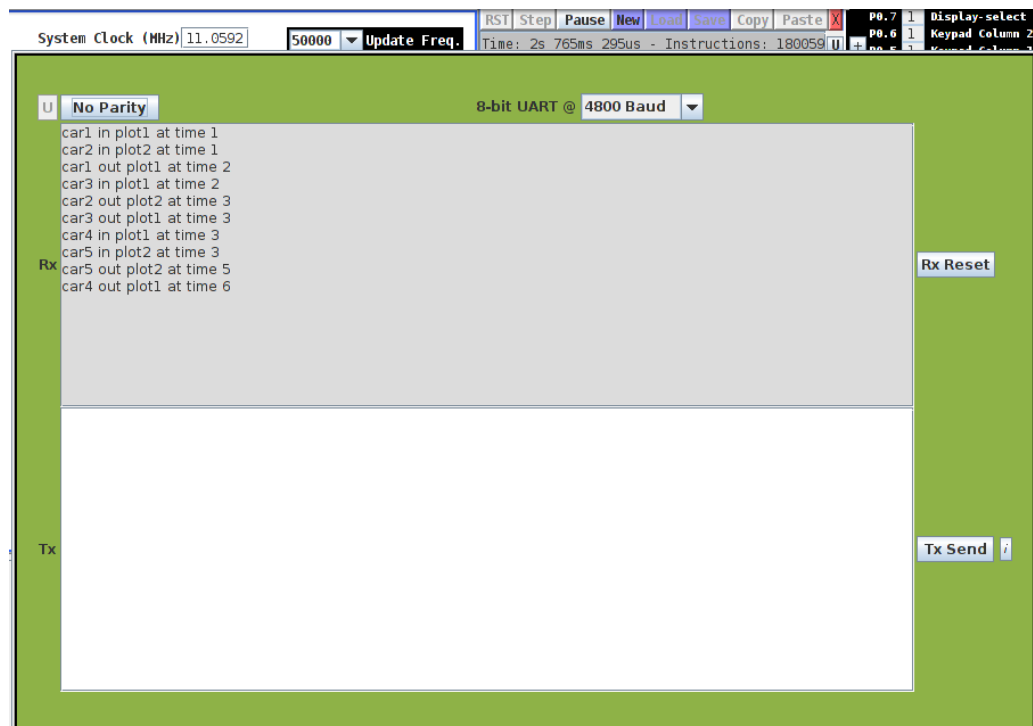
Car5 exits plot[1] at time 5

- (4) Extra credit. Display the output of mine log to UART in human-readable text format.

After we parked or exited, calling SemaphoreSignal( print ) to unlock printer. So Signal ten times, it will start printing.

[illegible]

## (5) TypeScript and screen shot



(6) Experience after completing Project Programming CheckPoint5

At the beginning, I don't know Edsim8051 how to work. I don't even know that System Clock should set it as 11.0592(MHz). Apart from this, I complete ppc1 which take most of mine time to understand what is SDCC and Edsim8051. Before I really started to write my ppc1, I have taken my two weeks to review class's slides and ask TA for advise. So I am very grateful for TA's help, I can't finish Project without TA's help and classmate's hep. Instead of this. Before completing the Programming CheckPoint1, I don't know what "ThreadYield" actually meaning. Due to this reason, I query network information about it. So I fully understand it after I have done "ppc1". And then, I feel great sense of accomplishment until "ppc5" is implemented. Although in the middle, I feel frustrated about implementing "ppc2-4". But not so strongly that I want to give up, I at most feel a little frustrated. When I start to implement "ppc5", I find a very big gap just like canyon. I run out of my weekends to think how to write it out perfectly but I failed. After that I entered a wall collision period just like infinite loop, even if I try hard to think and ask TA for advice. Because of "final exam", I couldn't spend all my time on project. So, I decided to spent one day to complete it. If I failed, I will give up. I still went to the lab for asking problem at one week before the final exam. Finally, I have some ideas for modifying previous mistakes. Although it spends more time than I can tolerate, I still feel worth it. Due to this, I know better about some chapter's problem solving skills of "Operating System". For example, I know more about "Critical Section", "Semaphore", "how to handle interrupt", "time units setting", "delay time", "current time", "Consumer and Producer", "Cooperative", etc... But the price I paid was my final exam, I feel a bit upset. So I hope the TA can see my efforts on project through rewarding me with grades. Then I will be more motivated to learn otherwise I will be disappointed. Finally I should thank anyone who helped me. I am not sure that TA will see what I write. I still want to let TA feel happy with the new year, then making my score happy.