

微控制器

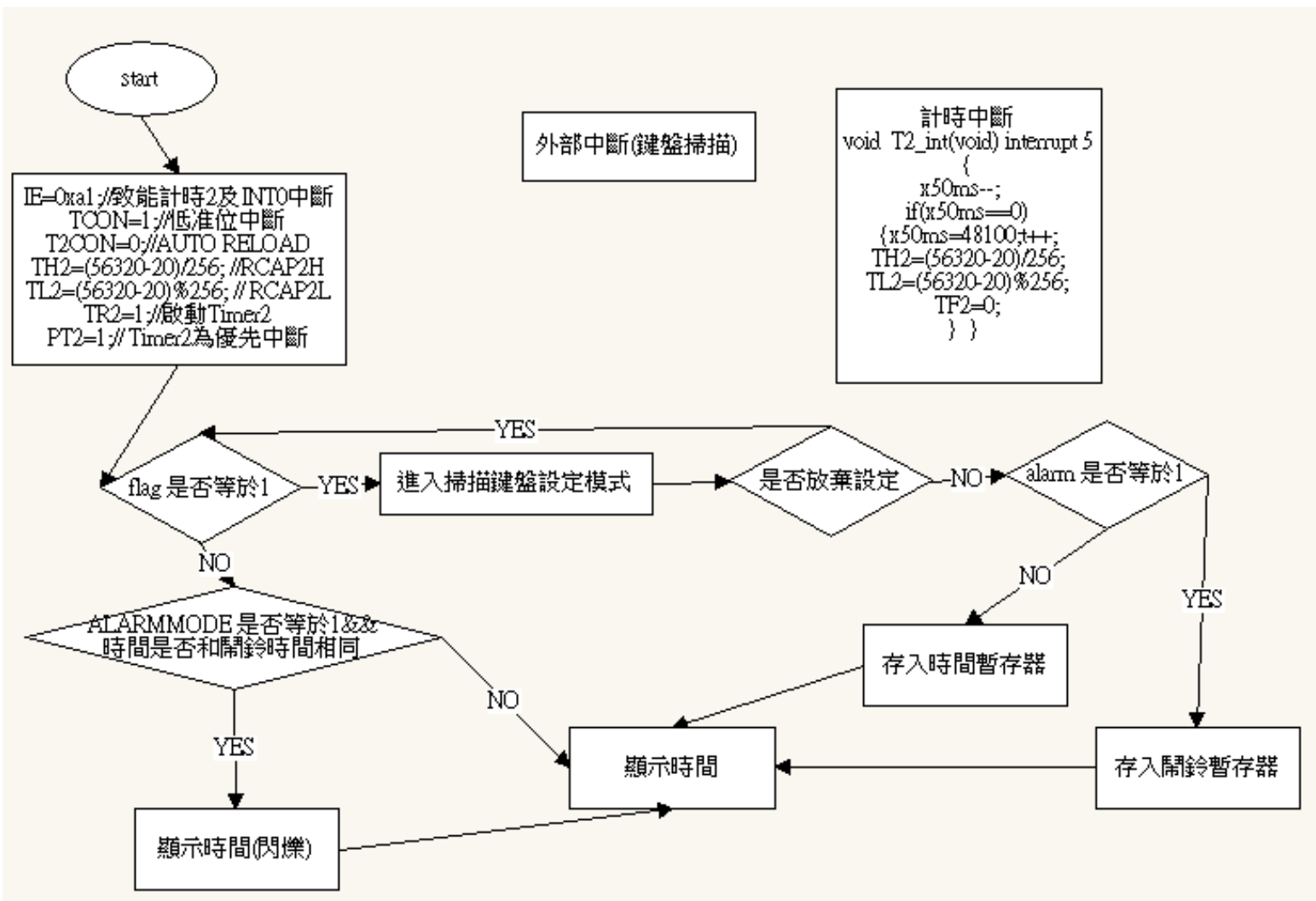
實驗 9

班級：機械 1A

學號：983003037

姓名：林耕宇

日期：100/6/5



一、實驗數據

變數代號	變數種類	功能
Coun1	函式	A[0]~A[9]七節管碼
Count2	函式	鍵號解譯-字元轉整數
Count3	函式	防彈跳
Count5	函式	鍵號解譯-模式設定
interINT0	interrupt 5函式	計時中斷
T2_int	interrupt 0函式	外部中斷(掃描)
A[10]	int	七節管碼
i	int	延遲
k	Int	紀錄掃描鍵盤後的"行"值
l	Int	紀錄掃描鍵盤後的"列"值
X	Int	跳出電晶體所需的迴圈
a1~a3	Int	為防彈跳所設的參數
s1~s3	Int	記錄前兩次按鈕所按下的訊號

b	int	控制s1~s3所對應的次序
alarmmode	char	鬧鐘模式旗標
alarm	char	鬧鐘設定模式旗標
num	int	累加進位所用的參數
W1~W3	int	時間設定模式所用的參數
X50ms	int	運算器所得的結果
f	int	時間設定模式旗標

```

IE=0xa1;//致能計時2及INT0中斷
TCON=1;//低准位中斷
T2CON=0;//AUTO RELOAD
TH2=(56320-20)/256; //RCAP2H
TL2=(56320-20)%256; // RCAP2L
TR2=1;//啟動Timer2
PT2=1;// Timer2為優先中斷
while (1)
{ P1=0xf0; count1();
/* .....以下為掃描顯示..... */
if(f==1)  {   IE=0xa1;
            if(b==0) { X=b;P2=A[0];
                      while(b==X) {P1=0xf0;
                                      P0=~1; P2=A[0]; P2=0xff;  c++;
            if(c>10){c=0;IE=0xa1;}}}
            if(b==1) {   X=b;    P1=k&l;
                      s1=P1;
                      P0=~1; count2();a1=P2;W1=num;
                      while(b==X) {P1=0xf0;
                                      P0=~1; P2=a1; P2=0xff;  c++;
            if(c>10){c=0;IE=0xa1;}}}
            if(b==2) {   X=b;    P1=k&l;
                      s2=P1;
                      P1=s2; count2();a1=P2;W1=num;
                      P1=s1;  count2();a2=P2;W2=num;
                      while(b==X) {P1=0xf0;
                                      P0=~1; P2=a1; P2=0xff;
                                      P0=~2; P2=a2; P2=0xff;  c++;
            if(c>10){c=0;IE=0xa1;}} }   start:

            if(b>=3) {if(b>3){s1=s2;s2=s3;} P1=k&l;

```

```

        s3=P1; X=b;
        P1=s3; count2();a1=P2;W1=num;
        P1=s2; count2();a2=P2;W2=num;
        P1=s1; count2();a3=P2;W3=num;
        while(b==X) { P1=0xf0;
            P0=~1; P2=a1; P2=0xff;
            P0=~2; P2=a2; P2=0xff;
            P0=~4; P2=a3; P2=0xff; c++;
        }
    }
    if(c>10){c=0;IE=0xa1; } } if(b>3){goto start; } }
}
/*.....以下為時間顯示.....*/
if(f==0) { IE=0xa1;
if(t>=10){t=0;t1++;}
if(t1>=6){t1=0;t2++;}
if(t2>=10){t=0;t1=0;t2=0;}
if(alarmmode>=1&& c==0){alarmmode=1;c=1;}
if(alarmmode>=2&& c==1){alarmmode=0;c=0;}
while(t==a1&&t1==a2&&t2==a3&&alarmmode==1){
    P0=~1; P2=A[a1]; for(i=0;i<500;i++);P2=0xff;
    P0=~2; P2=A[a2]; for(i=0;i<500;i++);P2=0xff;
    P0=~4; P2=A[a3]; for(i=0;i<500;i++);P2=0xff; a1=t;a2=t1;a3=t2;}
count1(); P1=0xf0;
    P0=~1; P2=A[t]; for(i=0;i<5;i++);P2=0xff;
    P0=~2; P2=A[t1]; for(i=0;i<5;i++);P2=0xff;
    P0=~4; P2=A[t2]; for(i=0;i<5;i++);P2=0xff;
} } }
/*.....以下為計時中斷.....*/
void T2_int(void) interrupt 5
{
x50ms--;
if(x50ms==0)
{x50ms=48100;t++;
TH2=(56320-20)/256;
TL2=(56320-20)%256;
TF2=0;
} }
/*.....以下為外部中斷.....*/
void interINT0(void) interrupt 0

```

```

{ IE=0xa0;          P1=0xf0;
  k=0xfe;
  while((k&0x10)!=0)
  { P1=k;
    if((P1&0xf0)!=0xf0)
    { l=(P1&0xf0)|0x0f;
      P1=k&l; while(P1!=0xf0){count3();P1=0xf0;}
      P1=k&l;
    }
    if(P1==~0x28||P1==~0x48||P1==0x7d||P1==0x7b||P1==0x77||P1==~0x18)
    {a1=~0;a2=~0;a3=~0;b=0;count5();}
    else{b++;}
  }
  break;
}
else{
  k=(k<<1)|0x01;
}
}
}

```

```

void count1(void)

```

```

{ A[0]=~0x3f;//0
  A[1]=~0x06;//1
  A[2]=~0x5b;//2
  A[3]=~0x4f;//3
  A[4]=~0x66;//4
  A[5]=~0x6d;//5
  A[6]=~0x7c;//6
  A[7]=~0x07;//7
  A[8]=~0x7f;//8
  A[9]=~0x6f;//9
}

```

```

void count2(void) {

```

```

if(P1==0x7e)
  {P2=A[0];num=0;}
if(P1==~0x41)
  {P2=A[1];num=1;}
if(P1==~0x42)
  {P2=A[2];num=2;}

```

```

if(P1==~0x44)
{P2=A[3];num=3;}
if(P1==~0x21)
{P2=A[4];num=4;}
if(P1==~0x22)
{P2=A[5];num=5;}
if(P1==~0x24)
{P2=A[6];num=6;}
if(P1==~0x11)
{P2=A[7];num=7;}
if(P1==~0x12)
{P2=A[8];num=8;}
if(P1==~0x14)
{P2=A[9];num=9;} }
void count3(void){ //stop jump
if(b==0){a1=~0;a2=~0;a3=~0;}
P0=~4;if(b==3||b==2){P0=~0;} P2=a3; P2=0xff;
P0=~2;if(b==2){P0=~0;} P2=a2; P2=0xff;
P0=~1;P2=a1; P2=0xff; }
void count5(void)
{ if(P1==0x7d) //(A)
{f=0;b=(-1); if(alarm==1){al1=W1;al2=W2;al3=W3;goto
start2;}t=W1;t1=W2;t2=W3;start2:W1=0;W2=0;W3=0;alarm=0;}
if(P1==0x7b) //(B)
{f=0;b=(-1);}
if(P1==~0x18) //(C)
{alarmmode++;}
if(P1==~0x28) //(D)
{f=1;}
if(P1==~0x48) //(E)
{f=1;alarm=1; } }

```

按數字鍵時：不論系統在何模式下將數字累進入存入輸入值暫存器。

按輸入完畢鍵時(A)：

設定時鐘模式下：輸入值存入目前時間，改回一般操作模式。

設定鬧鈴模式下：輸入值存入鬧鈴時間，改回一般操作模式。

在一般操作模式下：輸入值變為零。

按放棄輸入鍵時(B)：輸入值變為零，改變為一般操作模式。

按鬧鈴切換鍵時(C)：改變鬧鈴模式。

按設定目前時間鍵時(D)：改變為設定時鐘模式。

按設定鬧鈴時間鍵時(E)：改變為設定鬧鈴模式。

二、實驗問題

1. 在本次實驗中你學到了什麼？

如何去做一個CLOCK

2. 本實驗所使用的振盪器為11MHz。若未來改用可接受33MHz的微控器，你的RCAP2H，RCAP2L應該設為多少？

8051為標準的12T型式，其工作週期為每12個時脈一個工作週期。本課程所使用之8051以11.0592MHz振盪頻率之石英發振器為時脈產生器。在本次實驗中設定工作模式為計時內部時脈，因此其工作頻率為 $11.0592/12=0.9216\text{MHz}$ (每秒0.9216百萬個週期)。若要讓硬體計數器每秒鐘產生100次中斷，必需要讓計數器每數 $0.9216\text{MHz}/100(\text{次/s})=9216(\text{個脈波/每個中斷})$ ，

若為33MHz，則其工作頻率為 $33/12=2.75\text{MHz}$ (每秒2.75百萬個週期)，則RCAP2H
RCAP2L= $65536-27500=38035$

3. 若是你要做的不是時鐘，而是碼錶，你將選用那個計數器的那個工作模式？其相關的控制旗標及暫存器值各為何？

Timer2 auto reload C/T2=1當計數器

計數器是Up Down Counter上下數計時器，其上數或則下數是由外部接腳T2EX(P1.0)來控制。當T2EX=1時上數，T2EX=0時下數。當上數時，計數值滿溢(over flow)，即等於0FFFFH時，會設定TF2並產生中斷要求以及重填初始值為RCAP2L，RCAP2H。當下數時，計數值低下時(under flow)，即等於RCAP2L，RCAP2H時，會設定TF2並產生中斷要求，以及重填計數為最大值以及重填計數為最大值0FFFFH。EXF2在這個工作模式下，會隨滿溢或則是低下的發生做切換動作，若計數器是由00000H數到0FFFFH的話這個旗標就變成第17BIT

三、實驗討論

討論以Delay，以計時器除頻中斷的計時的差異，並討論其使用時機的不同。

以Delay所做的計數器也能夠達到使用計時器除頻中斷計時的效果，但在delay時，無法進行多於的事情，但計時器除頻中斷計時卻可以做到，delay的使用時機可能是用在間斷不同的計時時使用。