



HY13P 系类
HYCON IP 使用说明书

目录

1	晶片说明.....	1-6
2	文件说明.....	2-7
3	数位IP(TIMERA).....	3-7
3.1	范例名称	3-7
3.2	范例说明	3-7
3.3	系统设定	3-7
3.4	软体流程	3-8
3.5	程式范例	3-8
4	数位IP(TIMERB).....	4-9
4.1	范例名称	4-9
4.2	范例说明	4-9
4.3	系统设定	4-10
4.4	软体流程	4-10
4.5	范例程序	4-11
5	数位IP(TIMERC CAPUTURE).....	5-12
5.1	范例名称	5-12
5.2	范例说明	5-12
5.3	系统设定	5-12
5.4	软体流程	5-13
5.5	范例程序	5-13
6	数位IP(WDT).....	6-16
6.1	范例名称	6-16
6.2	范例说明	6-16
6.3	系统设定	6-16
6.4	软体流程	6-17
6.5	范例程序	6-17
7	数位IP (PWM)	7-18
7.1	范例名称	7-18
7.2	范例说明	7-18
7.3	系统设定	7-19
7.4	软体流程	7-19
7.5	范例程序	7-20

8	类比IP(ADC).....	8-21
8.1	范例名称	8-21
8.2	范例说明	8-22
8.3	系统设定	8-22
8.4	软体流程	8-23
8.5	范例程序	8-23
9	类比IP(MC)	9-27
9.1	范例名称	9-27
9.2	范例说明	9-27
9.3	系统设定	9-27
9.4	软体流程	9-28
9.5	范例程序	9-28
10	通讯IP(SPI)	10-30
10.1	范例名称	10-30
10.2	范例说明	10-30
10.3	系统设置	10-30
10.4	软体流程	10-31
10.5	范例程序	10-31
11	通讯IP (UART)	11-33
11.1	范例名称	11-33
11.2	范例说明	11-33
11.3	软体流程	11-34
11.4	范例程序	11-34
12	其他IP(OSC).....	12-36
12.1	范例名称	12-36
12.2	范例说明	12-36
12.3	系统设定	12-36
12.4	软体流程	12-37
12.5	范例程序	12-37
13	其他IP(LCD).....	13-39
13.1	范例名称	13-39
13.2	范例说明	13-39
13.3	系统设置	13-39
13.4	软体流程	13-40

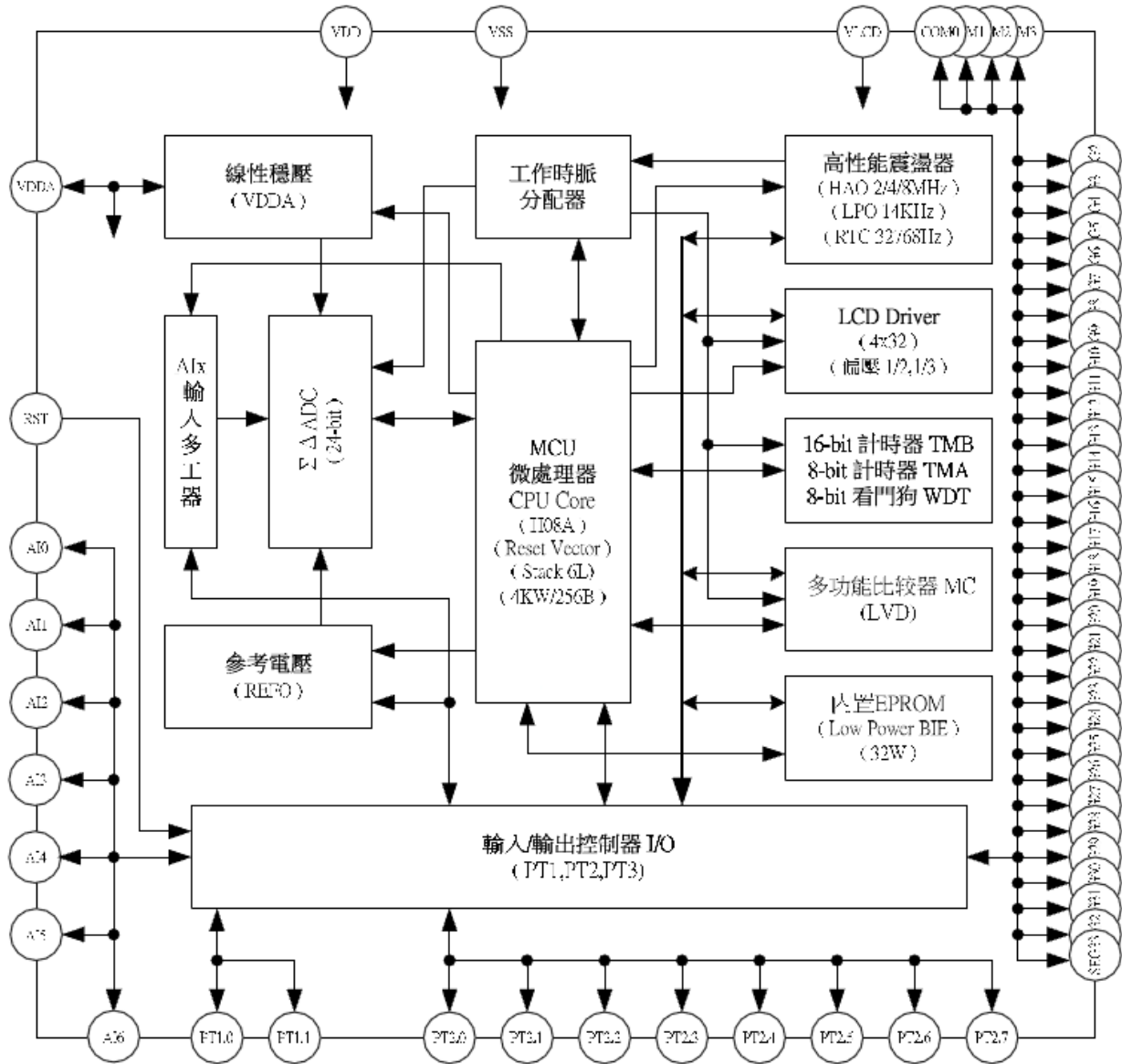
13.5	范例程序	13-40
14	其他IP(IO PORT).....	14-44
14.1	范例名称	14-44
14.2	范例说明	14-45
14.3	系统设定	14-45
14.4	软体流程	14-45
14.5	范例程序	14-46
15	其他IP (TOUCH KEY).....	15-47
15.1	范例名称	15-47
15.2	范例说明	15-47
15.3	系统设置	15-47
15.4	软体流程	15-48
15.5	范例程序	15-49
16	附件.....	16-50
17	修订纪录.....	17-50

注意:

- 1、本说明书中的内容，随着产品的改进，有可能不经过预告而更改。请客户及时到本公司网站下载更新 <http://www.hycontek.com>。
- 2、本规格书中的图形、应用电路等，因第三方工业所有权引发的问题，本公司不承担其责任。
- 3、本产品在单独应用的情况下，本公司保证它的性能、典型应用和功能符合说明书中的条件。当使用在客户的产品或设备中，以上条件我们不作保证，建议客户做充分的评估和测试。
- 4、请注意输入电压、输出电压、负载电流的使用条件，使 IC 内的功耗不超过封装的容许功耗。对于客户在超出说明书中规定额定值使用产品，即使是瞬间的使用，由此所造成的损失，本公司不承担任何责任。
- 5、本产品虽内置防静电保护电路，但请不要施加超过保护电路性能的过大静电。
- 6、本规格书中的产品，未经书面许可，不可使用在要求高可靠性的电路中。例如健康医疗器械、防灾器械、车辆器械、车载器械及航空器械等对人体产生影响的器械或装置，不得作为其部件使用。
- 7、本公司一直致力于提高产品的质量和可靠度，但所有的半导体产品都有一定的失效概率，这些失效概率可能会导致一些人身事故、火灾事故等。当设计产品时，请充分留意冗余设计并采用安全指标，这样可以避免事故的发生。
- 8、本规格书中内容，未经本公司许可，严禁用于其它目的之转载或复制。

1 晶片说明

HY13P56 IP 各功能基本使用说明



2 文件说明

- 数位 IP:TimerA/TimerB/TimerC capture/WDT/PWM
- 类比 IP:ADC/CMP
- 通讯 IP:SPI/UART
- 其他 IP:OSC/GPIO/touch/LCD

3 数位IP(TimerA)

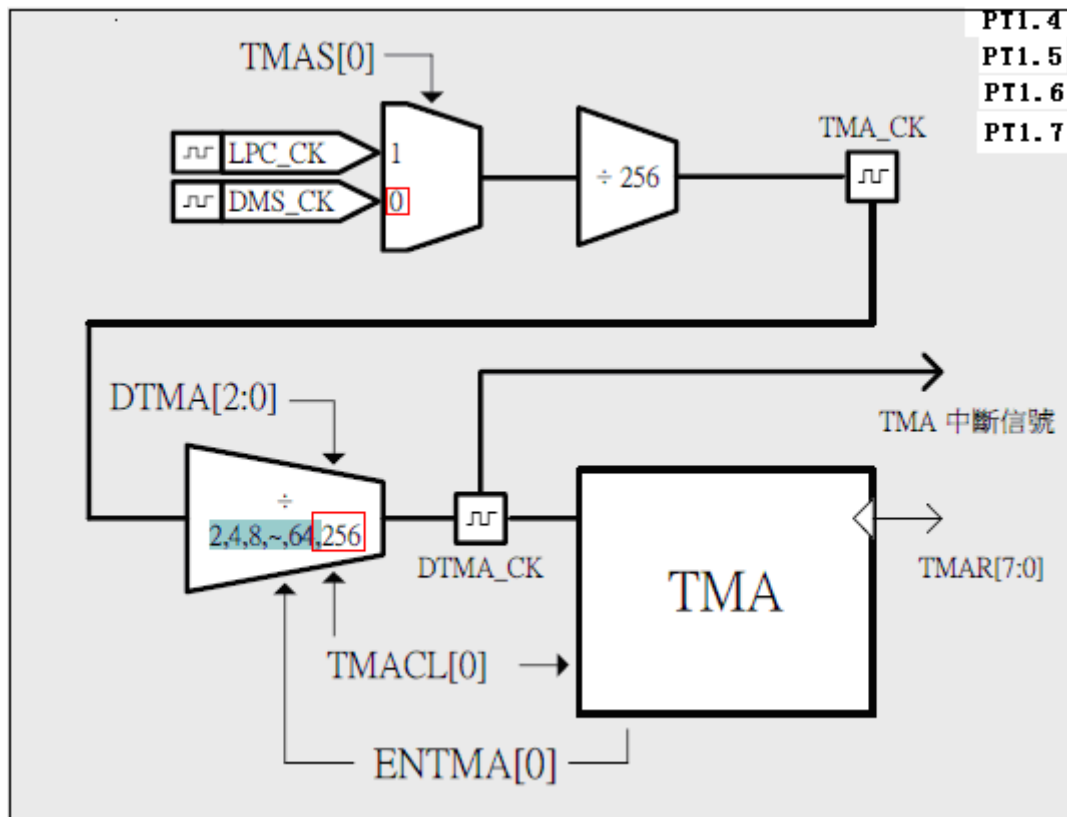
3.1 范例名称

TimerA 使用方式与说明(T6_TimerA)

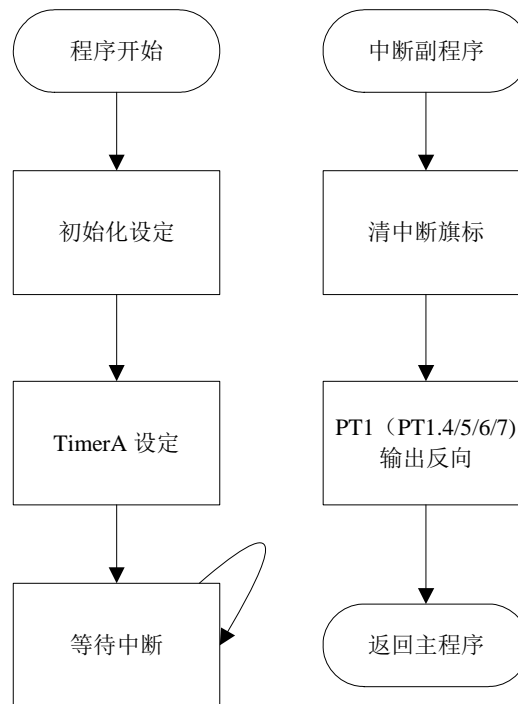
3.2 范例说明

- 系统时钟源选择内部 4M 且设置 DMS_CK=2M;
- 使能 TimerA 中断;
- 每次 timerA 中断反向 PT1 port 输出;

3.3 系统设定



3.4 软体流程



3.5 程式范例

```

include ..\13P.inc
;=====
org 0000h
jmp init
org 0004h
jmp TMA_interrupt
init:
    mvl 00h
    mvf OSCCN0,f,acce ;HS_CK frequency 选择 OSC_HAO DHS_CH 设置 HS_CK/2
                    ;OSCS[1:0]=00 DHS_CK[1:0]=00
                    ;DMS_CK 的频率设置 DHS_CK/2, CPU_CK 的频率选择 HS_CK/2;
                    ;DMS[2:0]=000 CPUS[0]=0

    mvl 1ch
    mvf OSCCN1,f,acce ;LPC_CK 的工作时钟 OSC_LPO ADC_CK 的工作时钟选 DHS_CK/4
                    ;OSC_LPO[1:0]=00 DADC[1:0]=01
                    ;DTMB[1:0]=11DTMB_CK 时钟 TMB_CK/8 TMB_CK 时钟为 HS_CK
                    ;DTMB[1:0]=00 TMCS[0]=0

    mvl 07h
    mvf OSCCN2,f,acce ;不使能外部振荡器;启用内部高速振荡器并选择 4M;使能内部低速振荡器
  
```



```

;ENRTC[0]=0 ENXT[0]=0 HAOM[1:0]=01 ENHAO[0]=1 LPO[0]=1
mvl 0f0h
mvf TRISC1,f,acce ;PT1.0-PT1.3 inputport PT1.4-PT1.7:output port
mvl 0fh
mvf PT1PU,f,acce ;PT1.0-PT1.3: INPUT, PULL HIGHT
clrf PT1,acce
mvl 11011100b ;ENTMA[0]=1 启动 timerA ;TMACL[0]=1 timerA 计数器归零
mvf TMACN,f,acce ;TMAS[0]=0 timerA 的工作时钟选择 DMS_CK ;
;DTMA[2:0]=111 TMA_CK÷256
mvl 10000100b ;使能 TimerA 中断与总中断源
mvf INTE0,f,acce
mainloop:
idle
nop
jmp mainloop
TMA_interrupt:
btss INTF0,TMAIF,acce
reti
bcf INTF0,TMAIF,acce
comf PT1,f,acce
reti

```

4 数位IP(TimerB)

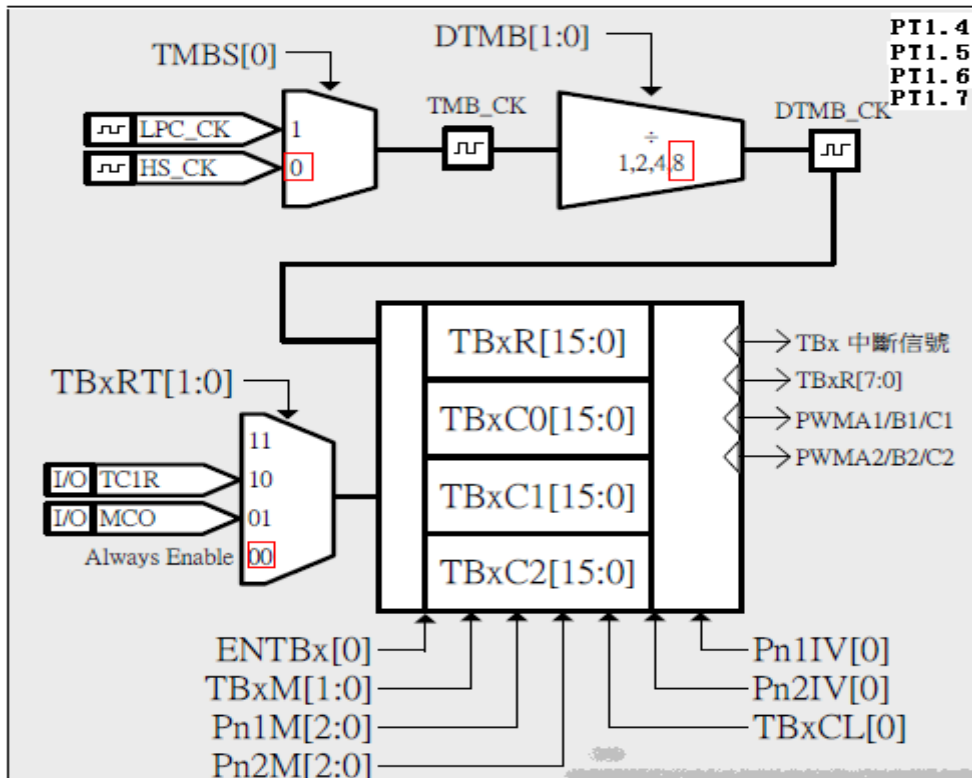
4.1 范例名称

TimerB 使用方式与说明(T5_TimerB1)

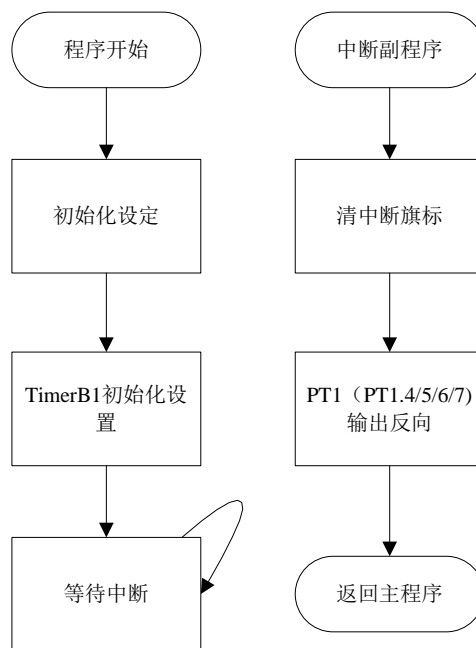
4.2 范例说明

- 系统时钟选择内部 4M，且 DTMB_CK 的时钟频率设置 TMB_CK/8;
- 使能 TimerB 计数与计数溢出中断，timerB 设置 16-bit 的工作模式;
- 每次 TimerB1 中断反向 IO port 输出;

4.3 系统设定



4.4 软体流程



4.5 范例程序

```

jmp    init
org    0004h
jmp    TMB_interrupt
init:
    mvl 00h
    mvf OSCCN0,f,acce ;HS_CK frequency 选择 OSC_HAO DHS_CH 设置为 HS_CK/2
                        ;OSCS[1:0]=00                DHS_CK[1:0]=00
                        ;DMS_CK 的频率设置 DHS_CK/2,    CPU_CK 的频率选择 HS_CK/2;
                        ;DMS[2:0]=000                CPUS[0]=0

    mvl 1ch
    mvf OSCCN1,f,acce ;LPC_CK 的工作时钟选择 OSC_LPO ;ADC_CK 的工作时钟选择 DHS_CK/4
                        ;OSC_LPO[1:0]=00                DADC[1:0]=01
                        ;DTMB[1:0]=11DTMB_CK 的时钟选择 TMB_CK/8 ;TMB_CK 的时钟设置为 HS_CK
                        ;DTMB[1:0]=00                TMCS[0]=0

    mvl 07h
    mvf OSCCN2,f,acce ;不使能外部振荡器; 启用内部高速振荡器并选择 4M; 使能内部低速振荡器
                        ;ENRTC[0]=0 ENXT[0]=0  HAOM[1:0]=01 ENHAO[0]=1  LPO[0]=1

    mvl 0f0h
    mvf RISC1,f,ace    ;PT1.0-PT1.3 inputport  PT1.4-PT1.7;output port
    mvl 0fh
    mvf PT1PU,f,acce  ;PT1.0-PT1.3, input, pull high
    clrf PT1,acce
    mvl 0ffh
    mvf TB1C0H,f,acce
    mvl 0ffh
    mvf TB1C0L,f,acce
    mvl 10000100b
    mvf TB1CN0,f,acce ;ENTB[0]=1 使能 TimerB1 TB1M[1:0]=00 16bit 计数模式
                        ;TB1CL[0]=1 计数寄存器 clear TC1ED[0]=0 TC2ED[0]=0

    mvl 10001000b    ;使能 TlmerB1 中断与总中断源
    mvf INTE0,f,acce

mainloop:
    idle
    nop
    jmp mainloop
TMB_interrupt:

```

```

btss INTF0,TB1IF,acce
reti
bcf INTF0,TB1IF,acce
comf PT1,f,acce
reti
    
```

5 数位IP(TimerC capture)

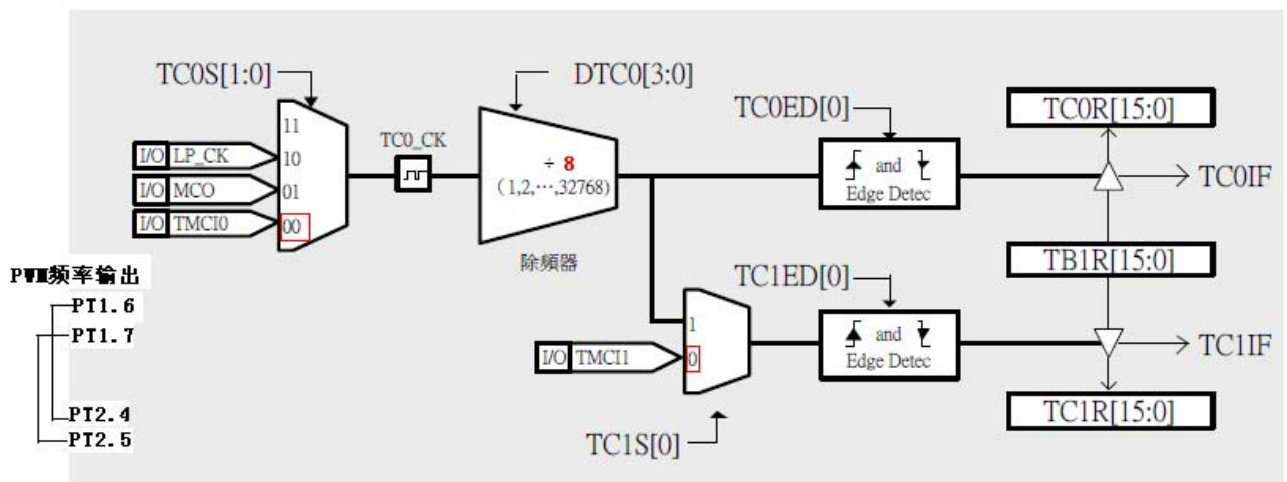
5.1 范例名称

TimerC capture 使用方式与说明(T8_TimerC)

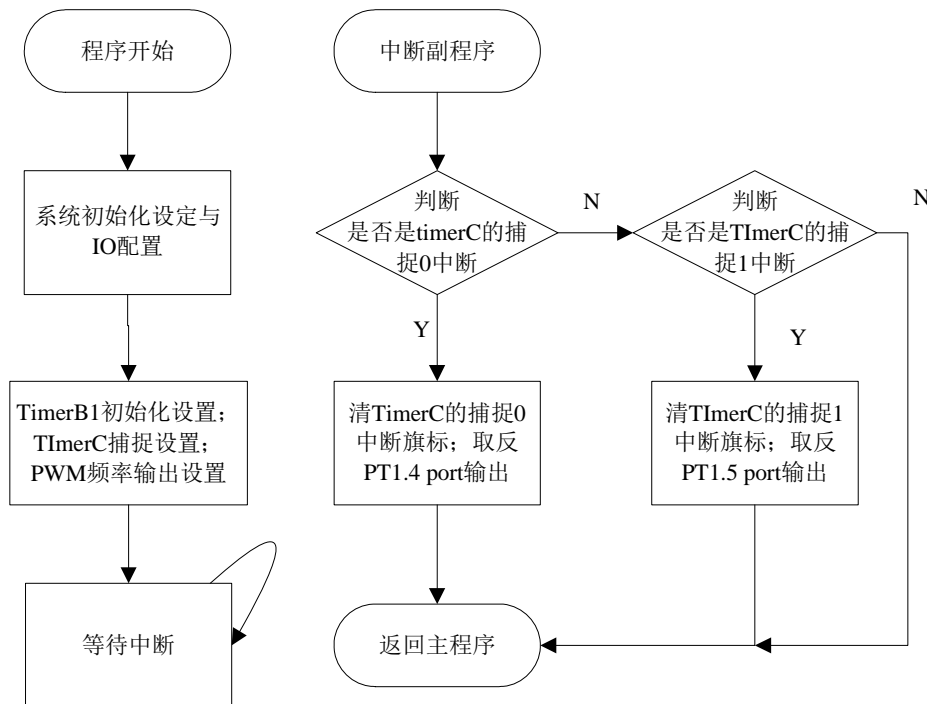
5.2 范例说明

- 系统时钟选择内部 4M，设置 DTMB_CK 的频率为 TMB_CK/8；
- 开启 timerB1 与设 TimerC；
- 配置 PT2.4 PT2.5 为 TimerC 捕捉信号输入 port；且配置 PT1.6 PT1.7 为 PWM 输出；
- 使能 TimerC 捕捉中断并反向 IO port 输出

5.3 系统设定



5.4 软体流程



5.5 范例程序

```
include ..\13P.inc
```

```
;=====
```

```
;for HY13S00 only
```

```
org 0000h
```

```
jmp init
```

```
org 0004h
```

```
jmp TMC_interrupt
```

```
init:
```

```
    mvl 00h
```

```
    mvf OSCCN0,f,acce ;HS_CK frequency 选择 OSC_HAO DHS_CH 设置为 HS_CK/2
```

```
                ;OSCS[1:0]=00                DHS_CK[1:0]=00
```

```
                ;DMS_CK 的频率设置 DHS_CK/2, CPU_CK 的频率选择 HS_CK/2;
```

```
                ;DMS[2:0]=000                CPUS[0]=0
```

```
    mvl 1ch
```

```
    mvf OSCCN1,f,acce ;LPC_CK 的工作时钟选择 OSC_LPO ADC_CK 的工作时钟选择 DHS_CK/4
```

```
                ;OSC_LPO[1:0]=00                DADC[1:0]=01
```

```
                ;DTMB_CK 的时钟选择 TMB_CK/8 TMB_CK 的时钟设置为 HS_CK
```

```
                ;DTMB[1:0]=11                TMCS[0]=0
```

```
    mvl 07h
```

```
mvf OSCCN2,f,acce ;不使能外部振荡器;启用内部高速振荡器为 4M;使能内部低速振荡器
;ENRTC[0]=0 ENXT[0]=0 HAOM[1:0]=01 ENHAO[0]=1 LPO[0]=1
```

```
mvl 0f0h
```

```
mvf TRISC1,f,acce ;PT1.0-PT1.3 inputport PT1.4-PT1.7: output port
```

```
mvl 0fh
```

```
mvf PT1PU,f,acce ;PT1.0-PT1.: INPUT, PULL HIGH
```

```
mvl 11000000b
```

```
mvf PT1DA,f,acce ;PT1.6 PT1.7: PWM OUTPUT
```

```
clrf PT1,acce
```

```
mvl 0cfh
```

```
mvf TRISC2,f,acce ;PT2.4 PT2.5: INPUT PORT
```

```
mvl 030h
```

```
mvf PT2PU,f,acce
```

```
mvl 01h
```

```
mvf PWRCN,f,acce ;CSFON[0]=1 使能 CSF 寄存器写入功能
```

```
mvl 40h
```

```
mvf CSFCN1,f,acce ;PWMBO 配置为 PT1.6 PT1.7 TMCIO 配置到 PT2.4 PT2.5
```

```
mvl 10000100b
```

```
mvf TB1CN0,f,acce ;ENTB1[0]=1 启动 timerB1 TB1M[1:0]=00 16-bit 模式
;TB1RT[1:0]=00 连续计数方式 TB1CL[0]=1 计数器归零
```

```
mvl 00000000b
```

```
mvf TB1CN1,f,acce ;PWMA1 波形选择 PWM6 PWMA0 波形输出选择 PWM6
```

```
mvl 0ffh
```

```
mvf TB1C0H,f,acce
```

```
mvf TB1C0L,f,acce
```

```
;;==PT1.6 PT1.7 PWM 输出设置 并用于 timerC 的捕捉==
```

```
lbsr 1
```

```
mvl 11000100b
```

```
mvf TB2CN0,f,b ;ENTB2[0]=1 启动 timerB1 TB2M[1:0]=10 8-bit 模式
;TB2RT[1:0]=00 连续计数方式 TB2CL[0]=1 计数器归零
```

```
mvl 00100011b
```

```
mvf TB2CN1,f,b ;PWMA1 波形选择 PWM3 PWMA0 波形输出选择 PWM4
```

```

mvl    0ffh
mvf    TB2C0L,f,b
mvl    07fh
mvf    TB2C1L,f,b
mvl    7fh
mvf    TB2C2L,f,b

;==timerC 捕捉 set==
;==timerC      set==
    clrf    TC0RH,acce    ;clear timerC 计数寄存器
    clrf    TC0RL,acce
    clrf    TC1RH,acce
    clrf    TC1RL,acce
    mvl    11101000b
    mvf    TCCN0,f,acce    ;ENTC[0]=1 启用 timerC;TC0S[1:0]=11 TC1 输入触发事件 TMC10
                                ;TC1S[0]=0 TC2 输入触发事件 TMC11  DTC0[3:0]=0011 TC0 除频 8

    mvl    00110000b
    mvf    INTE1,f,acce    ;使能 TC1 与 TC0 中断
    bsf    INTE0,GIE,acce

mainloop:
    idle
    nop
    jmp    mainloop
TMC_interrupt:
    btss   INTF1,TC0IF,acce
    jmp    TMC_interrupt1
    bcf    NTF1,TC0IF,acce
    bsf    TB1CN0,TB1CL,acce
    btgf   PT1,4,acce
    reti
TMC_interrupt1:
    btss   INTF1,TC1IF,acce
    reti
    bcf    INTF1,TC1IF,acce
    bsf    TB1CN0,TB1CL,acce
    btgf   PT1,5,acce
    reti

```

6 数位IP(WDT)

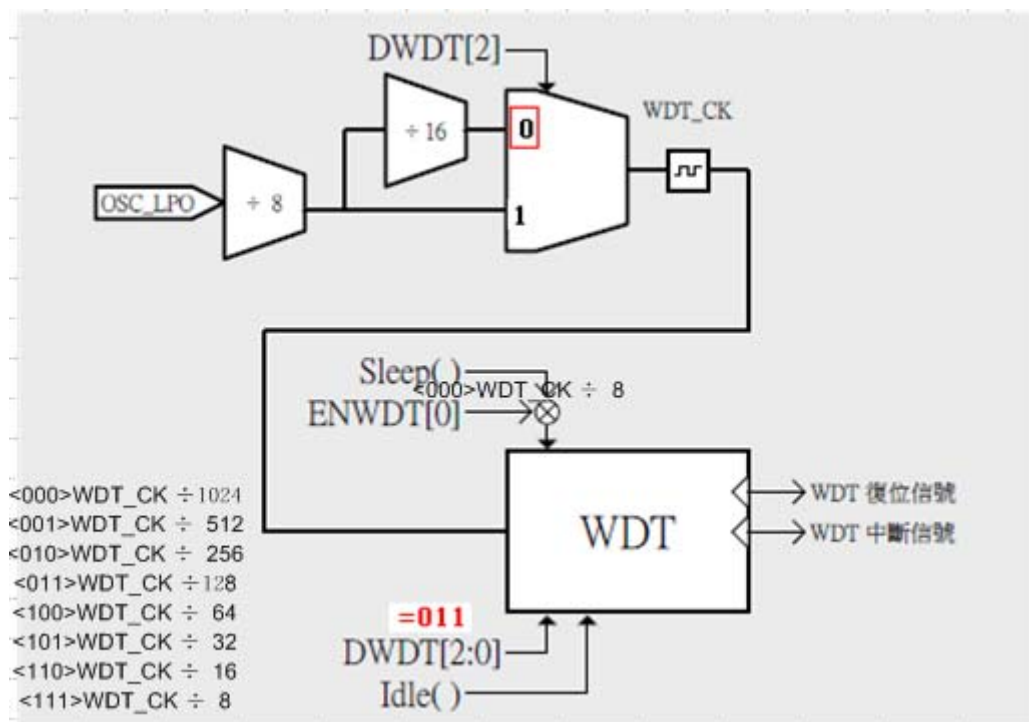
6.1 范例名称

WDT 使用方式与说明(T4_WDT)

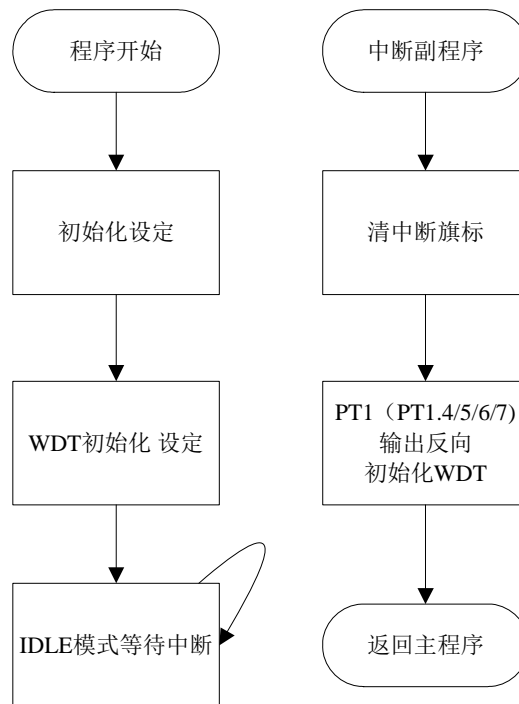
6.2 范例说明

- 初始化系统与 IOport 设置;
- IDLE 模式下每次进入 WDT 中断将 IOport (PT1.4/5/6/7) 输出反向

6.3 系统设定



6.4 软体流程



6.5 范例程序

```
include ..\13P.inc
```

```
;=====
```

```
org 0000h
```

```
jmp init
```

```
org 0004h
```

```
jmp WDT_interrupt
```

```
init:
```

```
    mvl 00h
```

```
    mvf OSCCN0,f,acce ;HS_CK frequency 选择 OSC_HAO DHS_CH 设置为 HS_CK/2
```

```
                ;OSCS[1:0]=00                DHS_CK[1:0]=00
```

```
                ;DMS_CK 的频率设置 DHS_CK/2, CPU_CK 的频率选择 HS_CK/2;
```

```
                ;DMS[2:0]=000                CPUS[0]=0
```

```
    mvl 10h
```

```
    mvf OSCCN1,f,acce ;LPC_CK 的工作时钟选择 OSC_LPO ADC_CK 的工作时钟选择 DHS_CK/4
```

```
                ;OSC_LPO[1:0]=00                DADC[1:0]=01
```

```
                ;DTMB_CK 的时钟选择 TMB_CK TMB_CK 的时钟设置为 HS_CK
```

```
                ;DTMB[1:0]=00                TMCS[0]=0
```

```

mvl 07h
mvf OSCCN2,f,acce ;不使能外部振荡器;启用内部高速振荡器并选择 4M; 使能内部低速振荡器
                    ;ENRTC[0]=0 ENXT[0]=0  HAOM[1:0]=01 ENHAO[0]=1    LPO[0]=1

mvl 0f0h
mvf TRISC1,f,acce ;PT1.0-PT1.3 inputport  PT1.4-PT1.7: output port
mvl 0fh
mvf PT1PU,f,acce ;PT1.0-PT1.3: input, pull high
clrf  PT1,acce
mvl 00001011b    ;使能看门狗  DWDT[2:0]=011  设置 128 分频
mvf WDTCN,f,acce
mvl 10010000b    ;使能看门狗中断与总中断源
mvf INTE0,f,acce
mainloop:
  idle
  nop
  jmp  mainloop

WDT_interrupt:
  btss  INTF0,WDTIF,acce
  reti
  bcf   INTF0,WDTIF,acce
  comf  PT1,f,acce
  mvl 00001011b    ;使能看门狗  DWDT[2:0]=011  设置 128 分频
  mvf WDTCN,f,acce
  reti

```

7 数位IP (PWM)

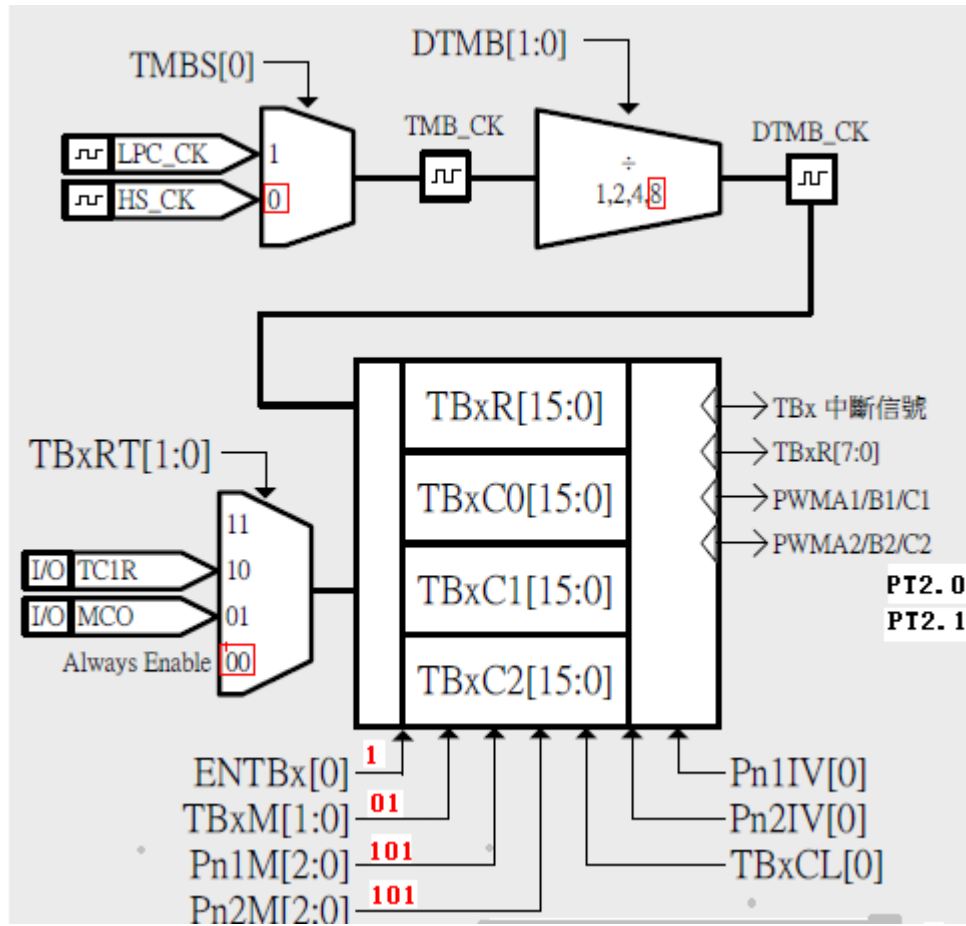
7.1 范例名称

PWM 的使用方式与说明 (T7_PWM)

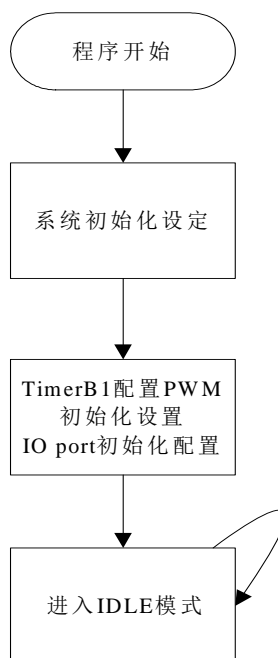
7.2 范例说明

- 系统时钟初始化配置;
- PWM 输出相关 IOport 配置;
- TlmerB1 PWM 输出配置

7.3 系统设定



7.4 软体流程



7.5 范例程序

```

include ..\13P.inc
;=====
org 0000h
jmp init
org 0004h
jmp TMB_interrupt
init:
    mvl 00h
    mvf OSCCN0,f,acce ;HS_CK frequency 选择 OSC_HAO DHS_CH 设置为 HS_CK/2
                        ;OSCS[1:0]=00 DHS_CK[1:0]=00
                        ;DMS_CK 的频率设置 DHS_CK/2, CPU_CK 的频率选择 HS_CK/2;
                        ;DMS[2:0]=000 CPUS[0]=0

    mvl 1ch
    mvf OSCCN1,f,acce ;LPC_CK 的工作时钟选择 OSC_LPO ADC_CK 的工作时钟选择 DHS_CK/4
                        ;OSC_LPO[1:0]=00 DADC[1:0]=01
                        ;DTMB[1:0]=11 DTMB_CK 时钟选择 TMB_CK/8 TMB_CK 的时钟设置为 HS_CK
                        ;DTMB[1:0]=00 TMCS[0]=0

    mvl 07h
    mvf OSCCN2,f,acce ;不使能外部振荡器; 启用内部高速振荡器并选择 4M; 使能内部低速振荡器
                        ;ENRTC[0]=0 ENXT[0]=0 HAOM[1:0]=01 ENHAO[0]=1 LPO[0]=1

    mvl 0f0h
    mvf TRISC1,f,acce ;PT1.0-PT1.3 inputport PT1.4-PT1.7: output port
    mvl 0fh
    mvf PT1PU,f,acce ;PT1.0-PT1.3: INPUT, PULL HIGH
    mvl 11110000b
    mvf PT1DA,f,acce ;FPWMB1[0]=1 PWMB1 输出 FPWMB0[0]=1
                        ;FPWMC1[0]=1 PWMC1 输出 FPWMC0[0]=1 PWMC0 输出

    clrf PT1,acce

    mvl 03h
    mvf TRISC2,f,acce ;PT2.0 PT2.1: OUTPUT port
    mvl 0fch
    mvf PT2PU,f,acce
    mvl 03h
    mvf PT2DA,f,acce ;FPWMA1[0]=1 FPWMA0[0]=1 设置 PT2.0 PT2.1 配置为 PWMA 的输出 port

```

```
mvl 00h
mvf CSFCN1,f,acce ;PWMCO PWMBO PWMAO 分别配置为
;PT1.4 PT1.5、 PT1.6 PT1.7、 PT2.0 PT2.1 输出
```

```
mvl 10100100b
mvf TB1CN0,f,acce ;ENTB1[0]=1 启动 timerB1 TB1M[1:0]=01 32-bit 模式
;TB1RT[1:0]=00 连续计数方式 TB1CL[0]=1 计数器归零
```

```
mvl 01010101b
mvf TB1CN1,f,acce ;PWMA1 波形选择 PWM6 PWMA0 波形输出选择 PWM6
```

```
mvl 0ffh
mvf TB1C0H,f,acce
mvf TB1C0L,f,acce
mvl 3fh
mvf TB1C1H,f,acce
mvl 0ffh
mvf TB1C1L,f,acce
mvl 7fh
mvf TB1C2H,f,acce
mvl 0ffh
mvf TB1C2L,f,acce
```

```
mainloop:
  idle
  nop
  jmp mainloop
```

```
TMB_interrupt:
  reti
```

8 类比IP(ADC)

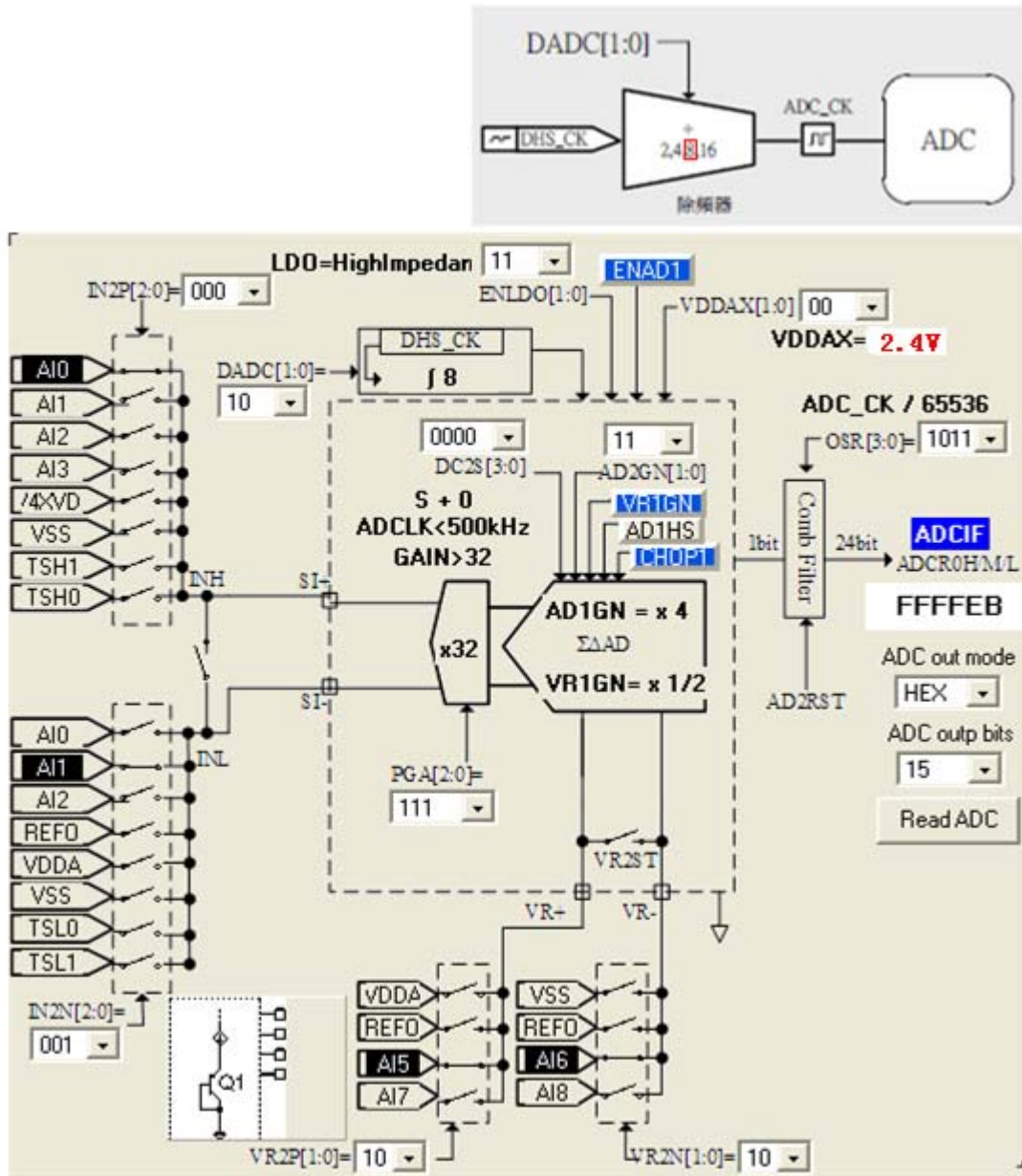
8.1 范例名称

ADC 使用方式与说明 (T2_ADC)

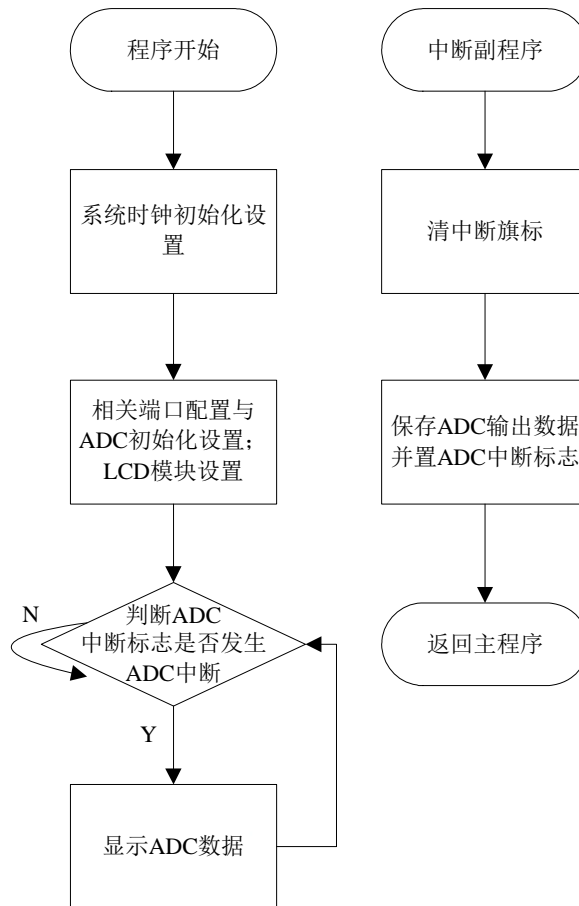
8.2 范例说明

- 系统初始化配置;
- 相关端口配置;
- ADC 初始化设置;
- 每次 ADC 中断采集到的 ADC 数据取有效位显示与 LCD 面板

8.3 系统设定



8.4 软体流程



8.5 范例程序

Main_ADC.asm\

=====

include main.inc

include ..\13P.inc

=====

org 0000h

jmp init

org 0004h

include interrupt.asm

init:

call sysinit

mainloop:

call display ;ADC display.

jmp mainloop

```

;=====
include display.asm
include LCDINDX.asm
include sysinit.asm
;=====
Sysinit.asm\
;=====
sysinit:
    mvl 00h
    mvf OSCCN0,f,acce ;OSC[1:0]=00 HS_CK=OSC_HAO DHS[1:0]=00 DHS_CK=HS_CK/2
                        ;DMS[1:0]=00 DMS_CK=DHS_CK/2 CPS[0]=0 CPU_CK=HS_CK/2

    mvl 20h
    mvf OSCCN1,f,acce ;LPCS[1:0]=00 LPC_CK=OSC_LPO DADC[1:0]=10 ADC_CK=DHS_CK/8
                        ;DTMB[1:0]=00 DTMB_CK/=TMB_CK/1 TMBS[0]=0 TMB_CK=HS_CK

    mvl 07h
    mvf OSCCN2,f,acce ;ENRTC[0]=0 ENXT[0]=0 XTS[1:0]=00
                        ;HAOM[1:0]=01 HAO=4M ENHAO=1 LPO=1

;=====Clear Memory=====
    LDPR 080h,FSR0
    MVL 080h
ClriniMen0:
    clrf POINC0,ACCE ;POINC0=0,FSR0=081h
    dcsz WREG,F,ACCE ;WREG-1 为 0 则跳过"jmp ClriniMen0"

    jmp ClriniMen0
    ldpr 100h,FSR0
    mvl 80h
ClriniMen1:
    clrf POINC0,ACCE
    dcsz WREG,F,ACCE
    jmp ClriniMen1

;=====
    mvl 0c8h
    mvf PWRCN,f,acce ;ENLDO[1:0]=11 LDO 输出 VDDA 电压 VDDAX[1:0]=00 VDDA=2.4V
    call delay
    lbr 1

;=====IO port=====
    mvl 0f0h

```



```

mvf TRISC1,f,acce
mvl 0eh
mvf PT1PU,f,acce
mvl 09h
mvf PT1DA,f,acce ;PT1.0 设置为比较器的输入通道
mvl 00h
mvf PT1,f,acce
mvl 03h
mvf TRISC2,f,acce
mvl 0fch
mvf PT2DA,f,acce ;设置 PT2.2-PT2.7 为模拟输入 port
mvl 00h
mvf PT2PU,f,acce
mvl 00h
mvf PT2,f,acce
mvl 04h ;PT3 port 配置为 analog signal input port
mvf CSFCN1,f,acce
clrf TRISC3,acce
mvl 0ffh
mvf PT3DA,f,acce
clrf PT3PU,acce
;=====
mvl 00000111b ;IN1P[2:0]=000 SI+ =AI0 IN1N[2:0]=001 SI- =AI1AD1GN[1:0]=11 Gain=4
mvf AD1CN1,f,acce
mvl 00001010b ;DC1S[3:0]=0000 DC offset=0 VR1N[1:0]=10 VR-=AI6 VR1P[1:0]=10 VR+=AI5
mvf AD1CN2,f,acce
mvl 10111111b ;OSR[3:0]=1011 ADC 超取样频率=ADC_CK/65536
;PGA[2:0]=111 *32 倍放大 VR1GN[0]=1 参考电压 VR/2
mvf AD1CN3,f,acce
mvl 00100001b ;ENAD[0]=1 使能 ADC(bit5) AD1HS[0]=0 ADC 工作频率 500K 以下
mvf AD1CN0,f,acce
;=====LCD set=====
lcd_init:
mvl 11010100b ;ENLCD[0]=1 使能 LCD LCDPR[0]=1 lcd 的电压有内部产生 VLCD[1:0]=01
;VLCD 为 3.05V 电压 LCDTE[0]=0 设置 1/3 偏压
mvf LCDCN1,f,b
mvl 60h ;LCDBL[0]=0 LCD 字节显示 LCDMX[1:0]=11 1/4 duty LCDS[0]=0
;LCD_CK=LPC_CK DLCD[1:0]=00 DLCD_CK=LCD_CK/2

```

```

mvf  LCDCN2,f,b
clrf lcd0,b
clrf lcd1,b
clrf lcd2,b
clrf lcd3,b
clrf lcd4,b
clrf lcd5,b
clrf ADCquit,acce
mvl  10000000b
mvf  INTE0,f,acce
mvl  01000000b
mvf  INTE1,f,acce
ret

;=====
delay:
  mvl 00
delay_loop:
  nop
  nop
  nop
  nop
  dcsz wreg,f,acce
  jmp  delay_loop
  ret

;=====
Interrupt.asm/
;=====
interrupt:
  btss INTF1,AD1IF,acce
  reti
  bcf  INTF1,AD1IF,acce
  mvl  02h
  cpsg ADCquit,acce ;丢弃前三笔数据

  jmp  Int_ADC0
  mvff AD1L,ADCoutL
  mvff AD1M,ADCoutM
  mvff AD1H,ADCoutH

```

```

bsf    flag,b_ADC,acce
jmp    interrupt_ext
Int_ADC0:
    inf ADCquit,f,acce
interrupt_ext:
    reti 1
    
```

9 类比IP(MC)

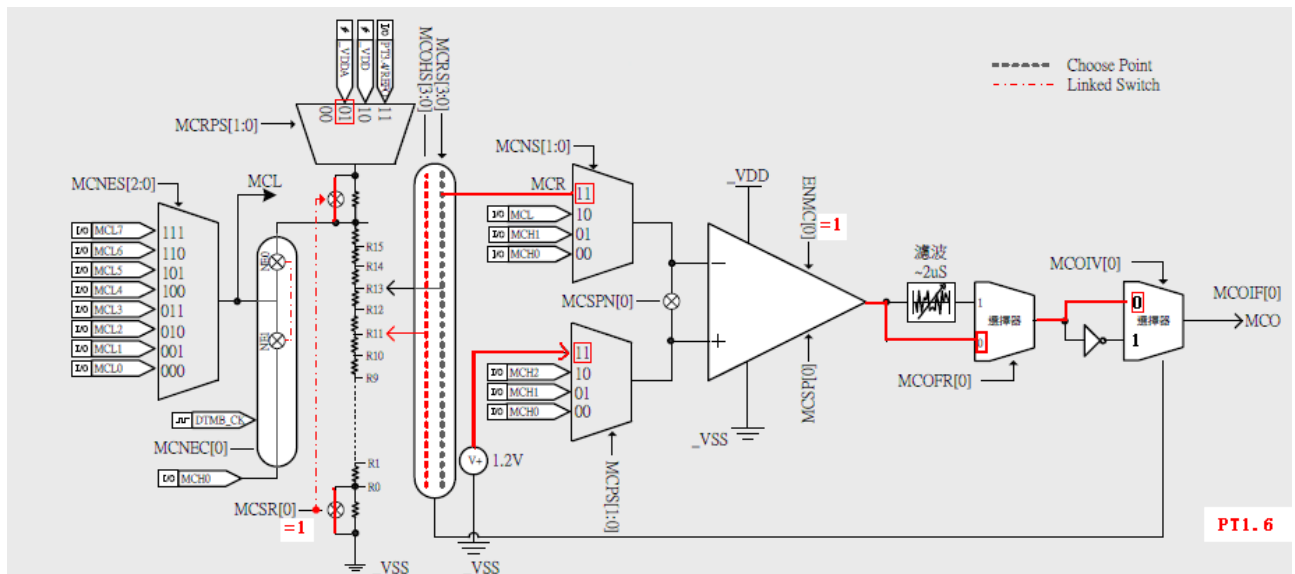
9.1 范例名称

比较器使用方式 (T3_MC)

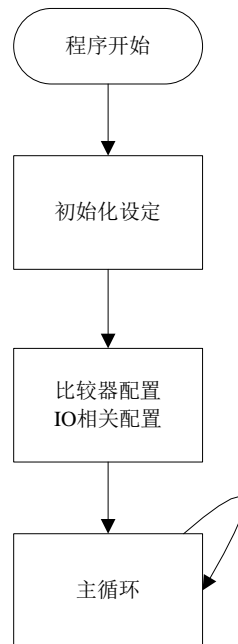
9.2 范例说明

- 系统初始化设置;
- 比较器输出端口配置及 IO port 配置;
- 比较器设置配置;

9.3 系统设定



9.4 软体流程



9.5 范例程序

```
include ..\13P.inc
```

```
=====
```

```
org 0000h
```

```
jmp init
```

```
org 0004h
```

```
reti
```

```
init:
```

```
    mvl 00h
```

```
    mvf OSCCN0,f,acce ;HS_CK frequency 选择 OSC_HAO DHS_CH 设置为 HS_CK/2
```

```
                ;OSCS[1:0]=00                DHS_CK[1:0]=00
```

```
                ;DMS_CK 的频率设置 DHS_CK/2, CPU_CK 的频率选择 HS_CK/2;
```

```
                ;DMS[2:0]=000                CPUS[0]=0
```

```
    mvl 10h
```

```
    mvf OSCCN1,f,acce ;LPC_CK 的工作时钟选择 OSC_LPO ADC_CK 的工作时钟选择 DHS_CK/4
```

```
                ;OSC_LPO[1:0]=00                DADC[1:0]=01
```

```
                ;DTMB_CK 的时钟选择 TMB_CK TMB_CK 的时钟设置为 HS_CK
```

```
                ;DTMB[1:0]=00                TMCS[0]=0
```

```
    mvl 07h
```

```
    mvf OSCCN2,f,acce ;不使能外部振荡器; 启用内部高速振荡器并选择 4M; 使能内部低速振荡器
```

```
                ;ENRTC[0]=0 ENXT[0]=0 HAOM[1:0]=01 ENHAO[0]=1 LPO[0]=1
```

```
mvl 0c8h
mvf PWRCN,f,acce ;ENLDO[1:0]=11 LDO 输出 VDDA 电压 VDDAX[1:0]=00 VDDA=2.4V
call delay
mvl 0f0h
mvf TRISC1,f,acce ;PT1.0-PT1.3 inputport PT1.4-PT1.7: output port
mvl 0fh
mvf PT1PU,f,acce ;PT1.0-PT1.3: input, pull high
mvl 08h
mvf PT1DA,f,acce ;PT1.6 配置为比较器的输出 port

mvl 00000100b
mvf CSFCN0,f,acce ;MCO 比较器的输出配置到 PT1.6 port

mvl 10101111b ;ENMC[0]=1 启用多功能比较器 MCLP[0]=0 低功耗模式
;MCSR[0]=1 4-bit 的阶梯电阻上串联电阻短路
mvf MCCN0,f,acce ;MCSPN[0]=0 输入信号不短路 MCPS[1:0]=11 比较器正端参考电压 1.2V
;MCNS[1:0]=11 比较器的负 MCR 内部 4bit 的电阻网络分压

mvl 00001000b
mvf MCCN1,f,acce ;MCNES[2:0]=000 MCPRS[1:0]=01 4-bit 电阻电源选择 VDDA
;MCNEC[0]=0 MCOFR[0]=0 MCOIV[0]=0

mvl 10010000b ;MCRS[3:0]=1001 4-bit 电阻网络选择 R9 MCOHS[3:0]=0000 不进行迟滞控制
mvf MCCN2,f,acce

mainloop:
    jmp mainloop

delay:
    mvl 00h
delayloop:
    nop
    nop
    dcsz wreg,f,acce
    jmp delayloop
    ret
```

10 通讯IP(SPI)

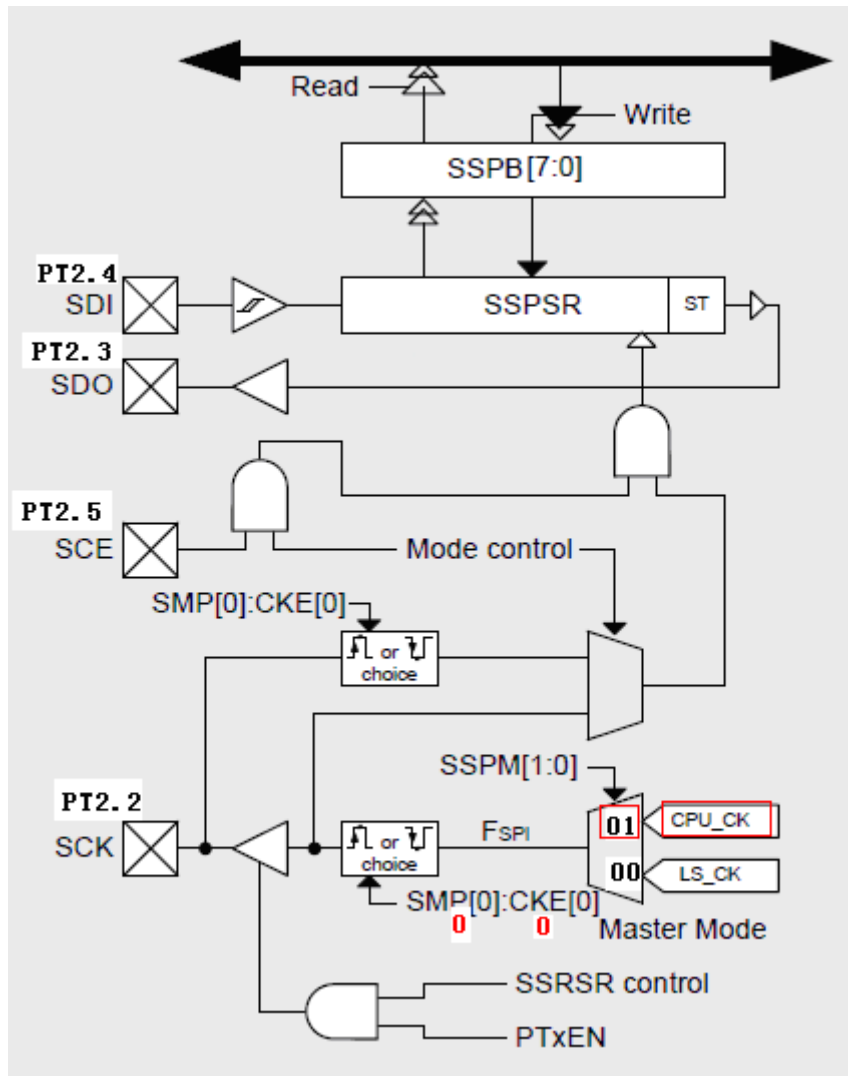
10.1 范例名称

SPI 数据发送及中断设定 (T10_SPI)

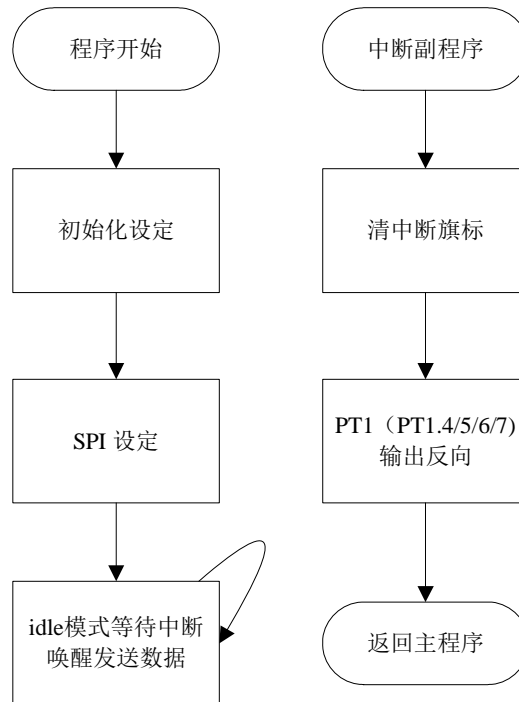
10.2 范例说明

- 初始化系统时钟
- SPI 模块配置
- 每次 SPI 中断取反 IO port 输出 (PT1.4/5/6/7)

10.3 系统设置



10.4 软体流程



10.5 范例程序

```

include ..\13P.inc
;=====
;for HY13S00 only
org 0000h
jmp init
org 0004h
jmp SPI_interrupt
init:
    mvl 00h
    mvf OSCCN0,f,acce ;HS_CK frequency 选择 OSC_HAO DHS_CH 设置为 HS_CK/2
                        ;OSCS[1:0]=00 DHS_CK[1:0]=00
                        ;DMS_CK 的频率设置 DHS_CK/2, CPU_CK 的频率选择 HS_CK/2;
                        ;DMS[2:0]=000 CPUS[0]=0

    mvl 1ch
    mvf OSCCN1,f,acce ;LPC_CK 的工作时钟选择 OSC_LPO ADC_CK 的工作时钟选择 DHS_CK/4
                        ;OSC_LPO[1:0]=00 DADC[1:0]=01
                        ;DTMB[1:0]=11 DTMB_CK 时钟选择 TMB_CK/8 TMB_CK 的时钟设置为 HS_CK
                        ;DTMB[1:0]=00 TMCS[0]=0

    mvl 07h
  
```

```

mvf OSCCN2,f,acce ;不使能外部振荡器; 启用内部高速振荡器并选择 4M; 使能内部低速振荡器
;ENRTC[0]=0 ENXT[0]=0 HAOM[1:0]=01 ENHAO[0]=1 LPO[0]=1

mvl 01h
mvf PWRCN,f,acce ;使能 CSF 寄存器写入功能
mvl 0f0h
mvf TRISC1,f,acce ;PT1.0-PT1.3 inputport PT1.4-PT1.7: output port
mvl 0fh
mvf PT1PU,f,acce ;PT1.0-PT1.3: input, pull hight
clrf PT1,acce
mvl 00101100b ;PT2.5(CS) PT2.3(SDO) PT2.2(SCK)设置为输出port PT2.4(SDI)设置为输入port
mvf TRISC2,f,acce
mvl 11010011b
mvf PT2PU,f,acce
clrf PT2,acce
MVL 00H
MVF CSFCN1,F,ACCE ;选择 PT2.2 PT2.3 PT2.4 PT2.5 为 SPI 模块 port
clrf SSPSTA,acce
mvf SSPB,w,acce
mvl 11000001b ;ENSSP[0]=1 使能 SPI CKP[0]=1 高电平为空闲状态
;CKE[0]=0 当空闲变成有效数据发送;SMP[0]=0 数据输出的中间采样
;SSMP[1:0]=01 主动模式 时钟源 CPU_CK

mvf SSPCN0,f,acce
mvl 01000000b
mvf INTE2,f,acce ;使能 SPI 中断
mvl 10000000b ;使能总中断源
mvf INTE0,f,acce
mainloop:
mvl 55h
mvf SSPB,f,acce
idle
nop
jmp mainloop
SPI_interrupt:
btss INTF2,SPIIF,acce
reti
bcf INTF2,SPIIF,acce
mvf SSPB,w,acce
comf PT1,f,acce

```


reti

11 通讯IP (UART)

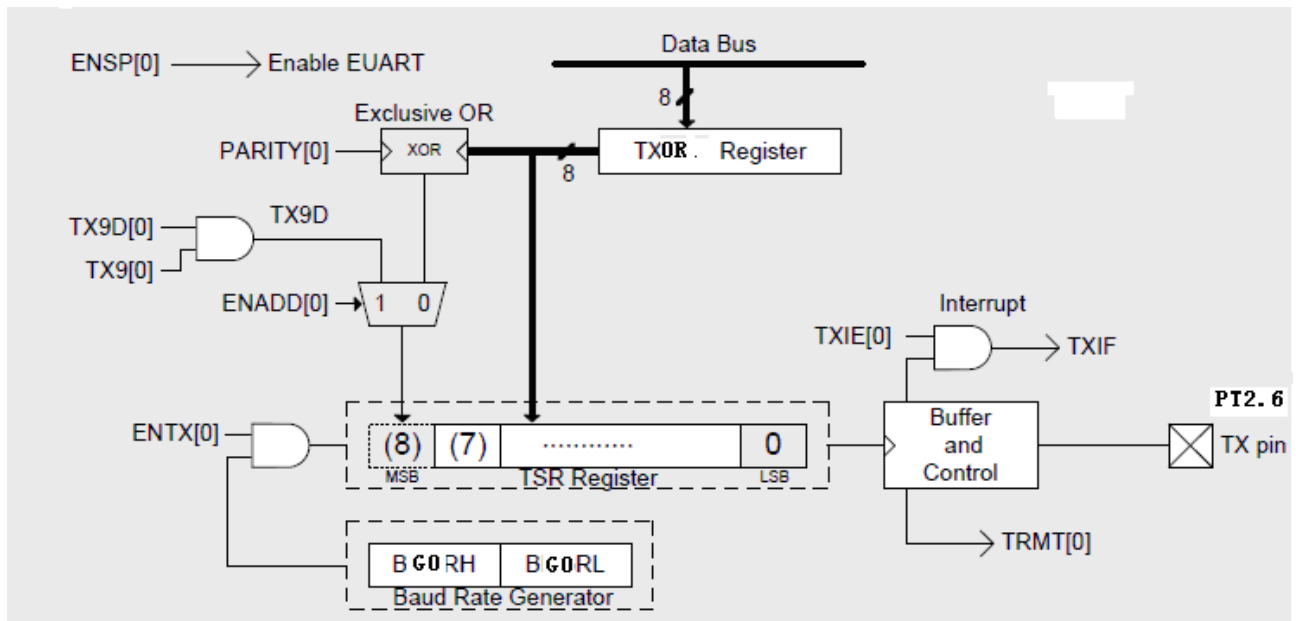
11.1 范例名称

UART 通讯设定 (T11_EUART)

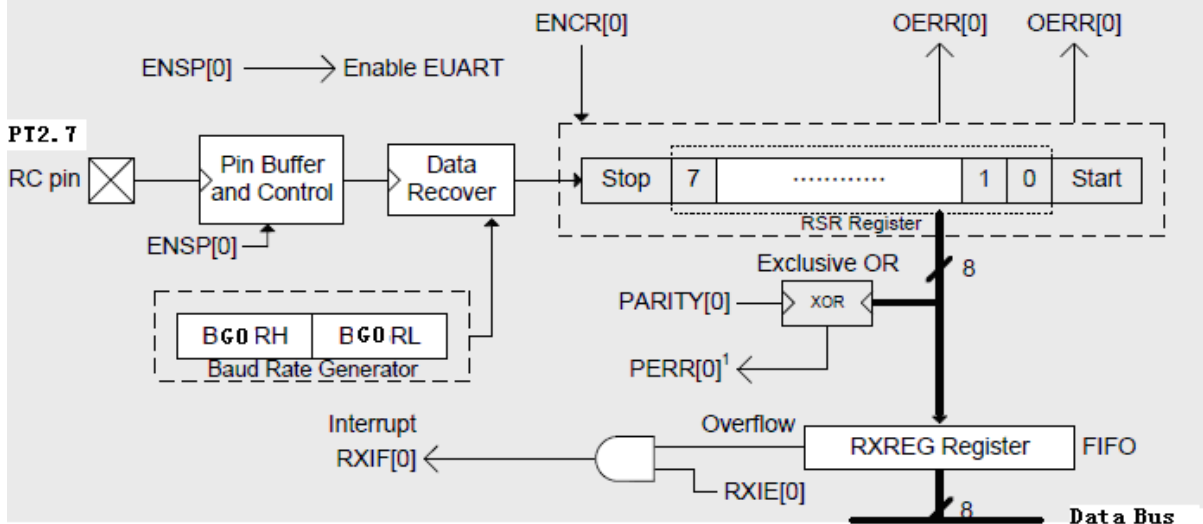
11.2 范例说明

- 初始化系统时钟设置;
- 相关 IO port 初始化及配置;
- 串口配置, 并且每次串口接收中断发送 0aah;

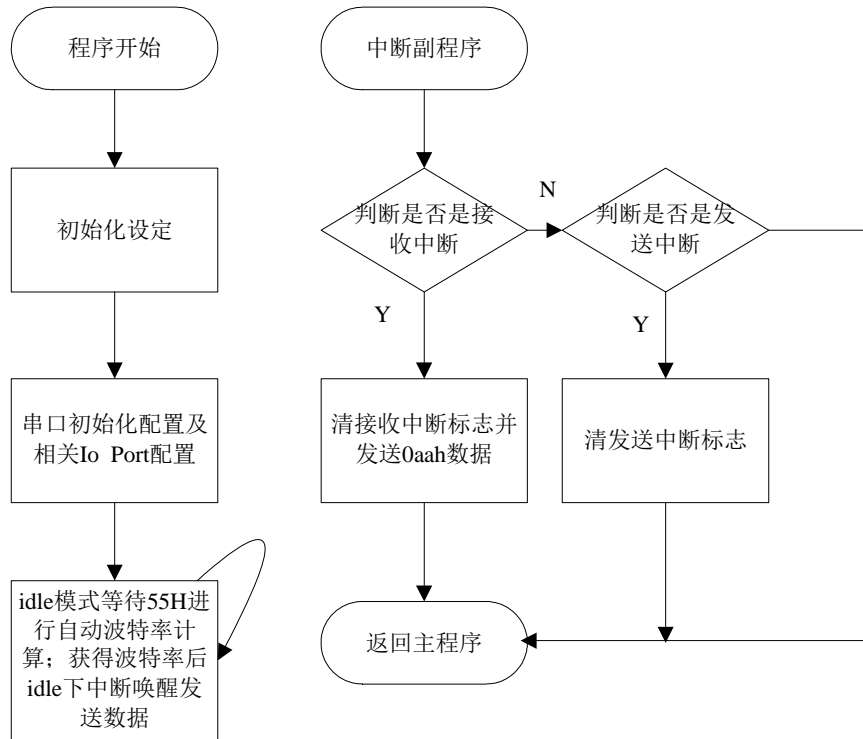
系统设定



EUART 8-BITs RECEIVE BLOCK DIAGRAM



11.3 软体流程



11.4 范例程序

```

include ..\13P.inc
;=====
;for HY13S00 only
org 0000h
jmp init
org 0004h
jmp UART_interrupt
init:
    mvl 00h
    mvf OSCCN0,f,acce ;HS_CK frequency 选择 OSC_HAO DHS_CH 设置为 HS_CK/2
                        ;OSCS[1:0]=00 DHS_CK[1:0]=00
                        ;DMS_CK 的频率设置 DHS_CK/2, CPU_CK 的频率选择 HS_CK/2;
                        ;DMS[2:0]=000 CPUS[0]=0

    mvl 1ch
    mvf OSCCN1,f,acce ;LPC_CK 的工作时钟选择 OSC_LPO ADC_CK 的工作时钟选择 DHS_CK/4
                        ;OSC_LPO[1:0]=00 DADC[1:0]=01
                        ;DTMB[1:0]=11 DTMB_CK 时钟选择 TMB_CK/8 TMB_CK 的时钟设置为 HS_CK
    
```

```

;DTMB[1:0]=00          TMCS[0]=0

mvl 07h
mvf OSCCN2,f,acce ;不使能外部振荡器; 启用内部高速振荡器并选择 4M; 使能内部低速振荡器
; ENRTC[0]=0 ENXT[0]=0 HAOM[1:0]=01 ENHAO[0]=1 LPO[0]=1

mvl 01h
mvf PWRCN,f,acce ;使能 CSF 寄存器写入功能
mvl 0f0h
mvf TRISC1,f,acce ;PT1.0-PT1.3 inputport PT1.4-PT1.7: output port
mvl 0fh
mvf PT1PU,f,acce ;PT1.0-PT1.3: input, pull hight
clrf PT1,acce

mvl 01000000b ;PT2.6(TX)设置为输出 port PT2.7(RX)设置为输入 port
mvf TRISC2,f,acce
mvl 10111111b
mvf PT2PU,f,acce
clrf PT2,acce
MVL 00H
MVF CSFCN2,F,ACCE ;选择 PT2.6 PT2.7 为 EUART 模块 port
;mvl 56h
;mvf BG0RL,f,acce
;clrf BG0RH,acce ;配置串口的波特率 9600
mvl 11000000b
mvf UR0CN,f,acce ;ENSP[0]=1 使能串口接收 ENTX[0]=1 使能串口发送
;TX9[0]=0 TX9D[0]=0 不使能 9bit 发送 PARITY[0]=0 WUE[0]=0 唤醒功能关闭
clrf UR0STA,acce
mvl 00001001b ;ENABD[0]=1 使能自动波特率 ENCR[0]=1 使能连续接收
mvf BA0CN,f,acce

mvl 00010000b
mvf INTE2,f,acce ;使能 TX RX 中断

clrf INTF2,acce
mvl 10000000b ;使能总中断源
mvf INTE0,f,acce
mainloop:
idle
nop

```

```

    jmp    mainloop
UART_interrupt:
    btss  INTF2,RCIF,acce
    reti
    bcf  INTF2,RCIF,acce
    mvf  RCREG,w,acce
    mvl  0aah
    mvf  TX0R,f,acce
    reti
    
```

12 其他IP(OSC)

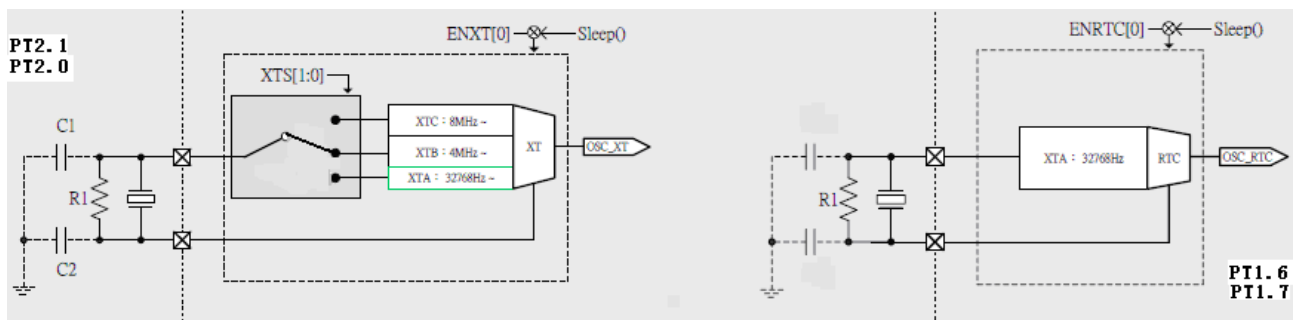
12.1 范例名称

外部高速与低速时钟设定 (T0_OSC)

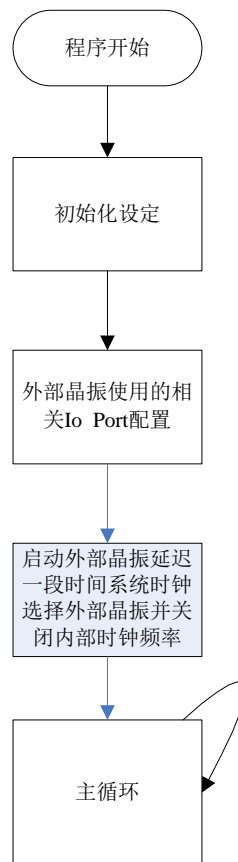
12.2 范例说明

- 系统初始化;
- 相关 IO Port 配置;
- 启用外部高速低速晶振并关闭内部时钟源;

12.3 系统设定



12.4 软体流程



12.5 范例程序

```
include ..\13P.inc
```

```
=====
```

```
org 0000h
```

```
jmp init
```

```
org 0004h
```

```
reti
```

```
=====
```

```
init:
```

```
    mvl 00h
```

```
    mvf OSCCN0,f,acce ;HS_CK frequency 选择 OSC_HAO DHS_CH 设置为 HS_CK/2
```

```
                ;OSCS[1:0]=00                DHS_CK[1:0]=00
```

```
                ;DMS_CK 的频率设置 DHS_CK/2, CPU_CK 的频率选择 HS_CK/2;
```

```
                ;DMS[2:0]=000                CPUS[0]=0
```

```
    mvl 10h
```

```
    mvf OSCCN1,f,acce ;LPC_CK 的工作时钟选择 OSC_LPO ADC_CK 的工作时钟选择 DHS_CK/4
```

```
                ;OSC_LPO[1:0]=00                DADC[1:0]=01
```

```

;DTMB_CK 的时钟选择 TMB_CK   TMB_CK 的时钟设置为 HS_CK
;DTMB[1:0]=00                 TMCS[0]=0

mvl 07h
mvf OSCCN2,f,acce ;不使能外部振荡器; 启用内部高速振荡器并选择 4M; 使能内部低速振荡器
;ENRTC[0]=0 ENXT[0]=0   HAOM[1:0]=01 ENHAO[0]=1   LPO[0]=1

mvl 0fch
mvf TRISC2,f,acce ;PT2.0 PT2.1 set input port
mvl 00h
mvf PT2PU,f,acce ;PT2.0 PT2.1 set no pullhigh
mvl 3fh
mvf TRISC1,f,acce ;PT1.6 PT1.7 :input port
mvl 00h
mvf PT1PU,f,acce ;PT1.6 PT1.7 :no pullhigh
mvl 11110011b
mvf OSCCN2,f,acce ;启动外部 RTC 与 XT 振荡器 且外部振荡器的频率选择 4M
call delay
mvl 10000000b
mvf OSCCN0,f,acce ;HS_CK 的频率选择 OSC_XT
mvl 01000000b
mvf OSCCN1,f,acce ;LPC_CK 的频率选择 OSC_RTC
OSCCN bcf 2,1,acce ;关闭内部 HAO 振荡器
bcf OSCCN2,0,acce ;关闭内部 LPO 振荡器
mainloop:
    nop
    jmp mainloop
delay:
    mvl 00h
delayloop:
    nop
    nop
    nop
    nop
    dcsz wreg,f,acce
    jmp delayloop
ret

```

13 其他IP(LCD)

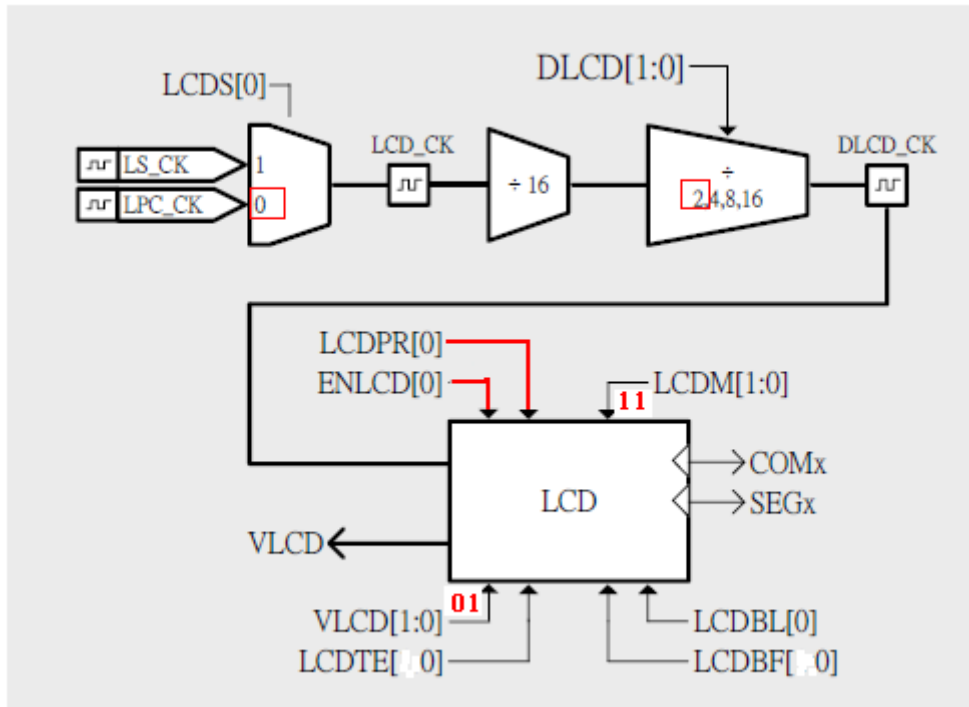
13.1 范例名称

LCD 模块使用方式的相关设定 (T9_LCD)

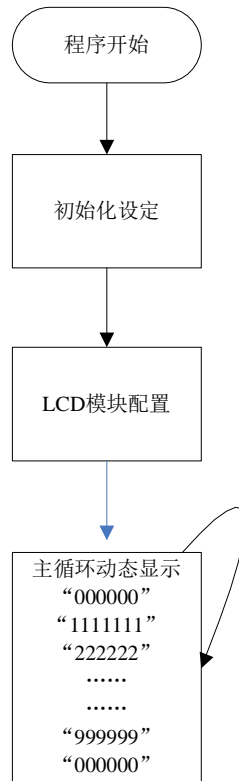
13.2 范例说明

- 系统初始化设置;
- LCD 模块配置
- 写 LCD 寄存器的相关数据来显示相关数据;

13.3 系统设置



13.4 软体流程



13.5 范例程序

Main_LCD.asm/

=====

```
include main.inc
include ..\13P.inc
```

=====

```
org 0000h
jmp init
org 0004h
reti
```

init:

```
call sysinit
clrf temp+1,acce
mvf temp+1,w,acce
mvf dispdata2,f,acce
mvf dispdata1,f,acce
mvf dispdata0,f,acce
```


mainloop:

```
call display ;lcd0-lcd5 0-9 变化显示数据显示 data
```

```
inf temp+1,f,acce
```

```
mvl 9
```

```
cpsg temp+1,acce
```

```
jmp mainloop1
```

```
clrf temp+1,acce
```

mainloop1:

```
mvf temp+1,w,acce
```

```
mvf dispdata2,f,acce
```

```
mvf dispdata1,f,acce
```

```
mvf dispdata0,f,acce
```

```
call delay
```

```
jmp mainloop
```

```
=====
```

```
include display.asm
```

```
include LCDINDX.asm
```

```
include sysinit.asm
```

```
=====
```

Sysinit.asm\

```
=====
```

sysinit:

```
mvl 00h
```

```
mvf OSCCN0,f,ace ;OSC[1:0]=00 HS_CK=OSC_HAO;DHS[1:0]=00 DHS_CK=HS_CK/2
;DMS[1:0]=00 DMS_CK=DHS_CK/2;CPS[0]=0 CPU_CK=HS_CK/2
```

```
mvl 20h
```

```
mvf OSCCN1,f,ace ;LPCS[1:0]=00 LPC_CK=OSC_LPO; DADC[1:0]=10 ADC_CK=DHS_CK/8
;DTMB[1:0]=00 DTMB_CK/=TMB_CK/1;TMBS[0]=0 TMB_CK=HS_CK
```

```
mvl 07h
```

```
mvf OSCCN2,f,ace ;;ENRTC[0]=0 ENXT[0]=0 XTS[1:0]=00
;HAOM[1:0]=01 HAO=4M ENHAO=1 LPO=1
```

```
=====Clear Memory=====
```

```
LDPR 080h,FSR0 ;FSR0=080h
```

```
MVL 080h ;W=080h
```

ClriniMen0:

```
clrf POINC0,ACCE ;POINC0=0,FSR0=081h
```

```
dcsz WREG,F,ACCE ;WREG-1 为 0 则跳过"jmp ClriniMen0"
```

```

    jmp    ClriniMen0
    ldpr  100h,FSR0
    mvl   80h
ClriniMen1:
    clrf  POINC0,ACCE
    dcsz  WREG,F,ACCE
    jmp   ClriniMen1
;=====
;=====LCD set=====
lcd_init:
    lbsr  1
    mvl   11010100b    ;ENLCD[0]=1 使能 LCD LCDPR[0]=1 lcd 的电压有内部产生
                                ;VLCD[1:0]=01 VLCD 为 3.05V 电压;LCDTE[0]=0 设置 1/3 偏压

    mvf   LCDCN1,f,b
    mvl   60h          ;LCDBL[0]=0 LCD 字节显示; LCDMX[1:0]=11 1/4 duty
                                ;LCDS[0]=0 LCD_CK=LPC_CK DLCD[1:0]=00 DLCD_CK=LCD_CK/2

    mvf   LCDCN2,f,b
    clrf  lcd0,b
    clrf  lcd1,b
    clrf  lcd2,b
    clrf  lcd3,b
    clrf  lcd4,b
    clrf  lcd5,b
    clrf  lcd6,b
    clrf  lcd7,b
    ret
;=====
delay:
    mvl   00h
    mvf   temp,f,acce
delayloop0:
    mvl   00
delay_loop1:
    nop
    nop
    nop
    nop
    dcsz  wreg,f,acce

```

```
    jmp    delay_loop1
    dcsz  temp,f,acce
    jmp    delayloop0
    ret

;=====
Display.asm\
;=====
display:
    mvl   0fh
    andf  dispdata2,w,acce
    bcf   MSTAT,c,ACCE
    rrfc  WREG,F,ACCE
    call  LDISP
    mvf   TBLDL,W,ACCE
    btss  dispdata2,0,ACCE
    mvf   TBLDH,W,ACCE
    mvf   lcd1,F,b

; swpf  dispdata2,f,acce
    mvl   0fh
    andf  dispdata2,w,acce
    bcf   MSTAT,c,ACCE
    rrfc  WREG,F,ACCE
    call  LDISP
    mvf   TBLDL,W,ACCE
    btss  dispdata2,0,ACCE
    mvf   TBLDH,W,ACCE
    mvf   lcd0,F,b

    mvl   0fh
    andf  dispdata1,w,acce
    bcf   MSTAT,c,ACCE
    rrfc  WREG,F,ACCE
    call  LDISP
    mvf   TBLDL,W,ACCE
    btss  dispdata1,0,ACCE
    mvf   TBLDH,W,ACCE
    mvf   lcd3,F,b
```

```
; swpf  dispdata1,f,acce
mvl  0fh
andf  dispdata1,w,acce
bcf   MSTAT,c,ACCE
rrfc  WREG,F,ACCE
call  LDISP
mvf   TBLDL,W,ACCE
btss  dispdata1,0,ACCE
mvf   TBLDH,W,ACCE
mvf   lcd2,F,b
```

```
mvl  0fh
andf  dispdata0,w,acce
bcf   MSTAT,c,ACCE
rrfc  WREG,F,ACCE
call  LDISP
mvf   TBLDL,W,ACCE
btss  dispdata0,0,ACCE
mvf   TBLDH,W,ACCE
mvf   lcd5,F,b
```

```
; swpf  dispdata0,f,acce
mvl  0fh
andf  dispdata0,w,acce
bcf   MSTAT,c,ACCE
rrfc  WREG,F,ACCE
call  LDISP
mvf   TBLDL,W,ACCE
btss  dispdata0,0,ACCE
mvf   TBLDH,W,ACCE
mvf   lcd4,F,b
ret
```

14 其他IP(IO port)

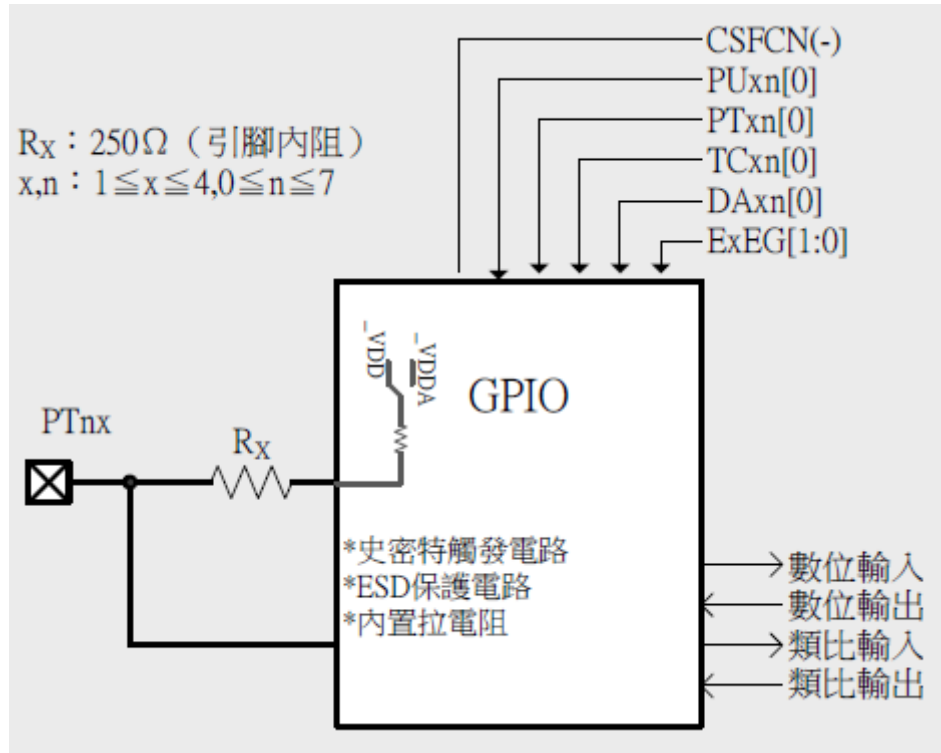
14.1 范例名称

IO port 相关的使用设定 (T1_IO)

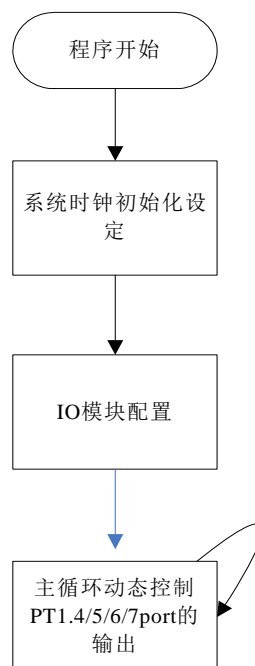
14.2 范例说明

- 系统初始化;
- IO port 相关的配置设定;

14.3 系统设定



14.4 软体流程



14.5 范例程序

```

include ..\13P.inc
;=====
org 0000h
jmp init
org 0004h
reti
init:
    mvl 00h
    mvf OSCCN0,f,acce ;HS_CK frequency 选择 OSC_HAO DHS_CH 设置为 HS_CK/2
                        ;OSCS[1:0]=00 DHS_CK[1:0]=00
                        ;DMS_CK 的频率设置 DHS_CK/2, CPU_CK 的频率选择 HS_CK/2;
                        ;DMS[2:0]=000 CPUS[0]=0

    mvl 10h
    mvf OSCCN1,f,acce ;LPC_CK 的工作时钟选择 OSC_LPO ADC_CK 的工作时钟选择 DHS_CK/4
                        ;OSC_LPO[1:0]=00 DADC[1:0]=01
                        ;DTMB_CK 的时钟选择 TMB_CK TMB_CK 的时钟设置为 HS_CK
                        ;DTMB[1:0]=00 TMCS[0]=0

    mvl 07h
    mvf OSCCN2,f,acce ;不使能外部振荡器; 启用内部高速振荡器并选择 4M; 使能内部低速振荡器
                        ; ENRTC[0]=0 ENXT[0]=0 HAOM[1:0]=01 ENHAO[0]=1 LPO[0]=1

    mvl 0f0h
    mvf TRISC1,f,acce ;PT1.0-PT1.3 inputport PT1.4-PT1.7: output port
    mvl 0fh
    mvf PT1PU,f,acce ;PT1.0-PT1.3: input, pull high
    mvl 0f0h
    mvf PT1,f,acce

mainloop:
    call delay
    call delay
    call delay
    call delay
    call delay
    call delay
    call delay
    call delay
    call delay
    comf PT1,f,acce

```

```
    jmp    mainloop
delay:
    mvl    00h
delayloop:
    nop
    nop
    nop
    nop
    nop
    nop
    nop
    nop
    nop
    nop
    nop
    nop
    nop
    dcsz  wreg,f,acce
    jmp    delayloop
ret
```

15 其他IP (touch key)

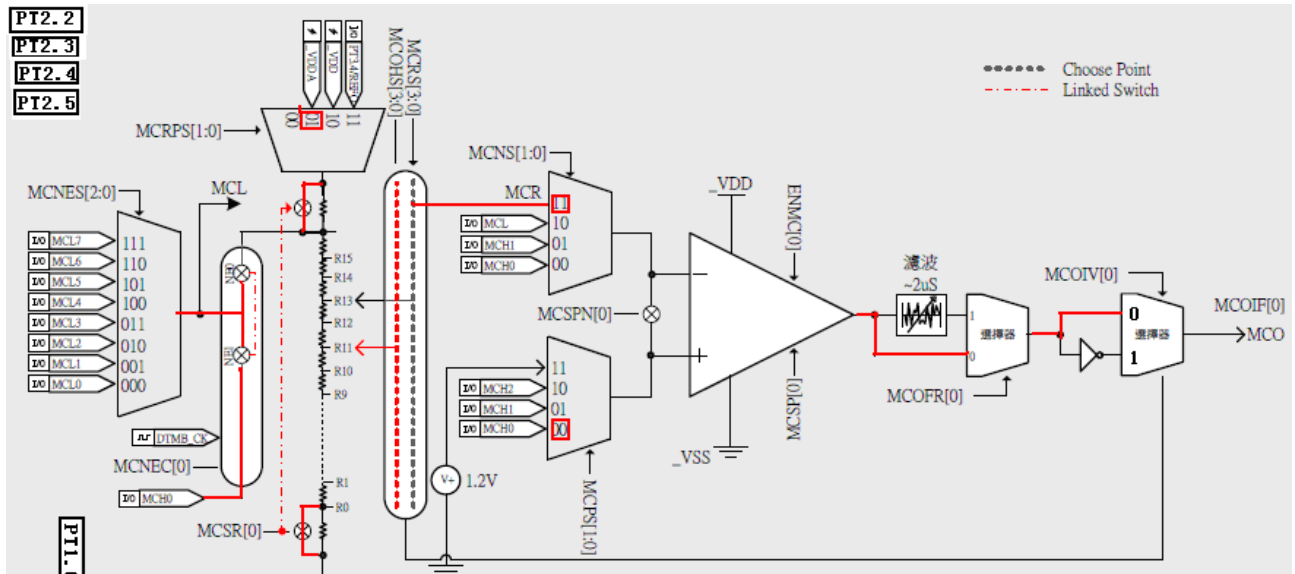
15.1 范例名称

比较器结合 TimerBtouch key 应用方式 (T12_touch)

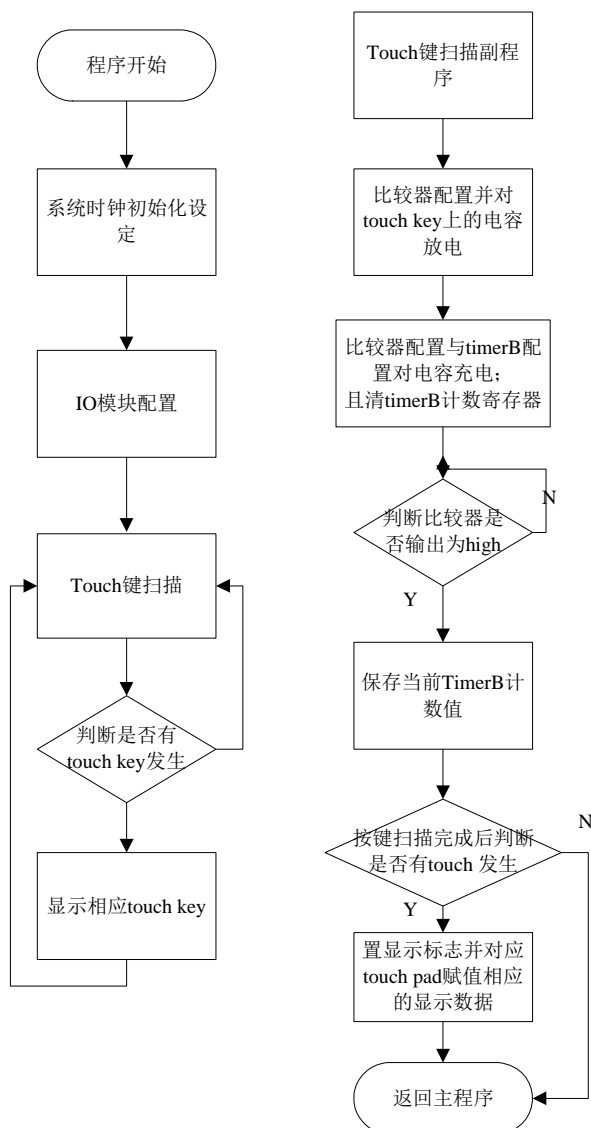
15.2 范例说明

- 系统初始化配置
- Touch pad 的相关端口配置;
- 比较器与 timerB 结合扫描 touch key 的配置;
- Touch key 的扫描及判断;

15.3 系统设置



15.4 软体流程



15.5 范例程序

```

Main_touch.asm/
;=====
include main.inc
include ..\13P.inc
;=====
org 0000h
jmp init
org 0004h
include interrupt.asm

init:
    call sysinit
mainloop:
    call touch_keyscan
    call display
    jmp mainloop
;=====
include keyscan.asm
include display.asm
include LCDINDX.asm
include sysinit.asm
;=====
touch_keyscan:
;=====
keyscan0:
    clrf TB1CN0,acce
    mvl 00000000b
    mvf MCCN2,f,acce
    mvl 00000000b
    mvf MCCN1,f,acce ;电容放电

    mvl 10110011b
    mvf MCCN0,f,acce
    call delay
    mvl 10000000b
    mvf MCCN2,f,acce
    
```

```
mvl 10100011b
mvf MCCN0,f,acce
mvl 00001100b
mvf MCCN1,f,acce
clrf TB1RH,acce
clrf TB1RL,acce
mvl 10000100b
mvf TB1CN0,f,acce
```

loop0:

```
btss PSTAT,MCO,acce
jmp loop0
clrf TB1CN0,acce
btsz flag,Flag_Base,acce
jmp keyscan0A
mvff TB1RL,base0dataL
mvff TB1RH,base0dataH
jmp keyscan1
```

keyscan0A:

```
mvff TB1RL,touch0dataL
mvff TB1RH,touch0dataH
```

;=====详见附件=====

16 附件



IPCode.rar

17 修订纪录

以下描述本檔差异较大的地方，而标点符号与字形的改变不在此描述范围。

版本	页次	变更摘要
V01	ALL	初版发行