

程式說明：

1· 復位TRW-900C 副程式： RESET\_TR\_MODE。

2· 配置TRW-900C 副程式： CONFIG\_900C。

3· 選擇工作模式：

發射模式： 副程式： TX\_MODE\_900C。

接收模式： 副程式： RX\_MODE\_900C。

空閒模式： 副程式： IDLE\_MODE\_900C。

4· 資料傳遞：

發送資料 副程式： SEND\_DATA。

接收資料 副程式： RECV\_DATA。

5· 其他模式：

設置功率 副程式： POWER\_SET。

```
#include"C:\SiLabs\MCU\IDEfiles\C51\ASM\C8051F320.INC"
    CLK equ P1.1                //900C模組引腳定義
    MISO equ P1.2
    MOSI equ P1.3
    CS equ P1.6                //CS 為低電平有效
    DR equ P1.4
    TX_LED equ P2.1           //發送模式指示燈 D5
    RX_LED equ P2.3           //接收模式指示燈 D6
    IDLE_LED equ p2.2
    TR_NUM equ 35h            //傳遞封包長度寄存器
    RSSI equ 36h
    CRC equ 37h                //接收校驗

    delh equ 30h                //延時參數
    dell equ 31h
    addr equ 32h                //位址傳送單元
    wdata equ 33h
    power equ 34h
    flag equ 20h.3             //讀數不滿 50 標誌
    error equ 20h.4           //接收錯誤標誌
    NULL equ 20h.7            //接收空標誌位元
org 0000h
    ajmp main
org 0100h
main:
    lcall init_mcu
    mov delh,#100
    mov dell,#100              //延時參數
    lcall delay
    lcall RESET_TR_MODE
    lcall CONFIG_900C
```

```

    mov delh,#0ah
    mov dell,#0dh
    lcall delay
    mov 20h,#00h
start:
    setb TX_LED                //開發射指示燈
    lcall TX_MODE_900C        //轉到發射模式
    mov TR_NUM,#10            //發送 10 個位元組的資料包
    lcall SEND_DATA           //發射資料
    lcall GET_DATA            //將 900C 模組轉換到接收模式並接收資料
start_00:
    lcall TX_MODE_900C
    mov TR_NUM,#20            //發送 20 個位元組的資料包
    lcall SEND_DATA
    lcall GET_DATA
start_01:
    lcall TX_MODE_900C
    mov TR_NUM,#80            //發送 80 個位元組的資料包
    lcall SEND_DATA
    lcall GET_DATA
start_02:
    ajmp start
//*****
//函數名      : GET_DATA
//功能描述    : 將900C模組轉換到接收模式並接收資料
//函數說明    : 接收資料的同時開接收指示燈，接收完畢關指示燈
//調用函數    : RX_MODE_900C，RECV_DATA
//傳入參數    : 無
//返回值      : 無
//*****
GET_DATA:
    clr TX_LED
    setb RX_LED
    mov r2,#1
GET_DATA_1:
    mov r3,#1
GET_DATA_2:
    lcall RX_MODE_900C
    lcall RECV_DATA
    clr flag
    djnz r3,GET_DATA_2
    djnz r2,GET_DATA_1
    clr RX_LED
    setb TX_LED

```

```

ret
/*****
//函數名      : SEND_DATA
//功能描述    : 將資料送進TRW-900C，並讓其發射出去
//函數說明    : 以可變封包為例
//調用函數    : W_BYTE_900C，SEND_FIFO_POINTER
//傳入參數    : 封包長度。通過TR_NUM傳遞
//返回值      : 無
*****/
SEND_DATA:
    clr CS
    jb MISO,$
    mov a,#7fh
    lcall W_BYTE_900C
    mov a,TR_NUM
    lcall W_BYTE_900C
    mov a,TR_NUM
    clr c
    subb a,#64
    jnc SEND_DATA_1
    mov r7,TR_NUM
SEND_DATA_0:                                //發射個數小於 64
    mov a,#88h
    lcall W_BYTE_900C
    djnz r7,SEND_DATA_0
    ajmp SEND_END
SEND_DATA_1:                                //發射個數大於 64
    mov r7,#64
SEND_DATA_2:
    clr c
    mov a,#88h
    lcall W_BYTE_900C
    djnz r7,SEND_DATA_2
    setb CS                                //等待資料發送；這裏要注意：也許程式運行到此處時
    jnb DR,$                                //資料已經發送完畢，則程式將卡在此處
    lcall SEND_FIFO_POINTER                //監視發送資料個數
    mov a,TR_NUM
    clr c
    subb a,#64                                //將發送過的資料個數減掉
    mov TR_NUM,a
    clr CS
    jb MISO,$
    mov a,#7fh
    lcall W_BYTE_900C

```

```

SEND_DATA_3:
    mov r7,TR_NUM
SEND_DATA_5:
    mov a,#88h
    lcall W_BYTE_900C
    djnz r7,SEND_DATA_5
SEND_END:
    setb CS
    jnb DR,$                //等待發送
    jb DR,$                //等待發送完畢
    ret
//*****
//函數名      : RECV_DATA
//功能描述    : 從TRW-900C中讀取接收到的資料，並存於80h開始的單元中
//函數說明    : 以可變封包為例
//調用函數    : W_BYTE_900C , R_BYTE_900C , RECV_FIFO_POINTER
//傳入參數    : 無
//返回值      : 無
//*****
RECV_DATA:
    lcall DR_TEST
; jnb dr,$
    jb NULL,REC            //檢測接收是否為空，NULL 為空標誌
    mov r0,#80h
    lcall RECV_FIFO_POINTER
    jnb flag,RECV_DATA_1   //flag 標誌，中途退出
    clr CS
    jb MISO,$
    mov a,#0ffh
    lcall W_BYTE_900C
    lcall R_BYTE_900C
    mov TR_NUM,a
    mov r7,TR_NUM
RECV_DATA_0:              //本段處理位組長度小於 50Byte 的封包
    lcall R_BYTE_900C
    mov @r0,a
    inc r0
    djnz r7,RECV_DATA_0
    lcall END_CHK
REC:
    ajmp RECV_DATA_5
RECV_DATA_1:
    clr CS
    jb MISO,$

```

```

    mov a,#0ffh
    lcall W_BYTE_900C
    lcall R_BYTE_900C
    mov TR_NUM,a
    mov r7,#50
RECV_DATA_2:                                //本段處理位元組長度大於 50Byte 的封包
    lcall R_BYTE_900C
    mov @r0,a
    inc r0
    djnz r7,RECV_DATA_2
    setb CS
    mov a,TR_NUM
    clr c
    subb a,#50
    mov TR_NUM,a
    lcall RECV_FIFO_POINTER
    jnb flag,RECV_DATA_4
    jb DR,$
    clr CS
    jb MISO,$
    mov a,#0ffh
    lcall W_BYTE_900C
    mov r7,TR_NUM
RECV_DATA_3:
    lcall R_BYTE_900C
    mov @r0,a
    inc r0
    djnz r7,RECV_DATA_3
    lcall END_CHK
    ajmp RECV_DATA_5
RECV_DATA_4:
    clr CS
    jb MISO,$
    mov a,#0ffh
    lcall W_BYTE_900C
    mov r7,#50
    ajmp RECV_DATA_2
RECV_DATA_5:
    setb CS
    ret
//*****
//函數名      : RECV_FIFO_POINTER
//功能描述    : 讀900C寄存器FB的值，監視接收資料的個數（50個）
//函數說明    : 監視接收到51個Bytes資料時，退出

```

```

//調用函數   : R_CONFIG_BYTE_900C
//傳入參數   : 無
//返回值     : flag 為1，則接收資料未到50就接收結束，為0，則接收資料到50
//*****

```

```

RECV_FIFO_POINTER:
    clr flag
    mov addr,#0fbh
    lcall R_CONFIG_BYTE_900C
    jnb DR,RECV_FIFO_POINTER_1
    clr c
    subb a,#50
    jc RECV_FIFO_POINTER
    ajmp RECV_FIFO_POINTER_2

```

```

RECV_FIFO_POINTER_1:
    setb flag

```

```

RECV_FIFO_POINTER_2:
    ret

```

```

//*****

```

```

//函數名     : END_CHK
//功能描述   : 讀接收資料最後的RSSI值和CRC值
//調用函數   : R_CONFIG_BYTE_900C
//傳入參數   : 無
//返回值     : error，為0則接收資料無誤，為1則接收資料有誤
//*****

```

```

END_CHK:
    lcall R_BYTE_900C
    mov RSSI,a
    lcall R_BYTE_900C
    mov CRC,a
    jb acc.7,right
    setb error
    ret

```

```

right:
    clr error
    ret

```

```

delay:
    mov r7,delh
delay_0:
    mov r6,dell
    djnz r6,$
    djnz r7,delay_0
    ret

```

```

//*****

```

```

//函數名      : RESET_TR_MODE
//功能描述    : 復位900C模組，寫復位時序以及復位參數
//調用函數    : W_BYTE_900C
//傳入參數    : 無
//返回值      : 無
//*****

```

RESET\_TR\_MODE:

```

    setb CS
    mov delh,#10
    mov dell,#10
    lcall delay
    clr CS
    mov delh,#20
    mov dell,#10
    lcall delay
    setb CS
    mov delh,#20
    mov dell,#10
    lcall delay
    clr CS
    jb MISO,$
    mov a,#30h
    lcall W_BYTE_900C
    jnb MISO,$
    setb CS
    ret

```

//\*\*\*\*\*

```

//函數名      : W_BYTE_900C
//功能描述    : 向900C模組寫1Byte資料
//調用函數    : 無
//傳入參數    : 傳入要寫的內容，由A傳遞
//返回值      : 無

```

//\*\*\*\*\*

W\_BYTE\_900C:

```

    push 05h
    mov r5,#08h
    clr c

```

W\_BYTE\_900C\_1:

```

    clr CLK
    rlc a
    mov MOSI,c
    setb CLK
    nop
    djnz r5,W_BYTE_900C_1

```

```

    clr CLK
    pop 05h
    ret
//*****
//函數名      : R_BYTE_900C
//功能描述    : 從900C模組中讀1Byte資料
//調用函數    : 無
//傳入參數    : 無
//返回值      : 讀取的內容，在A中
//*****
R_BYTE_900C:
    push 05h
    mov r5,#08h
    clr a
    clr c
R_BYTE_900C_1:
    clr CLK
    mov c,MISO
    rlc a
    setb CLK
    nop
    djnz r5,R_BYTE_900C_1
    clr CLK
    pop 05h
    ret
//*****
//函數名      : W_CONFIG_BYTE_900C
//功能描述    : 向900C模組寫配置資訊
//調用函數    : W_BYTE_900C
//傳入參數    : 1、位址，通過addr傳送。2、該位址的內容，通過wdata傳送
//返回值      : 無
//*****
W_CONFIG_BYTE_900C:
    clr CS
    jb MISO,$           //寫的時候必須判斷 MISO 為低電平時，才能寫
    mov a,addr
    lcall W_BYTE_900C
    nop
    mov a,wdata
    lcall W_BYTE_900C
    setb CS
    ret
//*****
//函數名      : R_CONFIG_BYTE_900C

```

```

//功能描述    : 讀配置資訊
//調用函數    : W_BYTE_900C , R_BYTE_900C
//傳入參數    : 位址，通過addr傳送
//返回值      : 讀出的資料存在A中
//*****
R_CONFIG_BYTE_900C:
    clr CS
    jb MISO,$
    mov a,addr
    lcall W_BYTE_900C
    nop
    nop
    lcall R_BYTE_900C
    nop
    setb CS
    ret
//*****
//函數名      : POWER_SET
//功能描述    : 配置900C的功率參數
//調用函數    : W_BYTE_900C , R_BYTE_900C
//傳入參數    : 功率值，通過power傳送
//返回值      : 無
//*****
POWER_SET:
    clr CS
    jb MISO,$
    mov a,#7Eh
    lcall W_BYTE_900C
    mov r5,#08h
power_1:
    mov a,power
    lcall W_BYTE_900C
    djnz r5,power_1
    setb CS
    clr CS
    nop
    mov r5,#08h
    jb MISO,$
    mov a,#0FEh
    lcall W_BYTE_900C
    nop
power_2:
    lcall R_BYTE_900C                //功率參數設定完後，對其進行讀出核對
    cjne a,power,POWER_SET

```

```

    djnz r5,power_2
    setb CS
    ret
//*****
//函數名      : SEND_FIFO_POINTER
//功能描述    : 讀900C寄存器FA的值，監視發射資料的個數
//調用函數    : W_BYTE_900C，R_BYTE_900C
//傳入參數    : 無
//返回值      : 無
//*****
SEND_FIFO_POINTER:
    mov addr,#0fah
    lcall R_CONFIG_BYTE_900C
    clr c
    subb a,#05h
    jnc SEND_FIFO_POINTER
    ret
//*****
//函數名      : CONFIG_900C
//功能描述    : 對900C配置工作頻率，發射功率等參數
//調用函數    : W_CONFIG_BYTE_900C，R_CONFIG_BYTE_900C，POWER_SET
//傳入參數    : 無
//返回值      : 無
//*****
CONFIG_900C:
    mov dptr,#CONFIG_TABLE
    mov r7,#34
    mov r4,#00h
CONFIG_900C_1:
    mov a,r4
    movc a,@a+dptr
    mov addr,a
    inc r4
    mov a,r4
    movc a,@a+dptr
    mov wdata,a
    inc r4
    lcall W_CONFIG_BYTE_900C
    djnz r7,CONFIG_900C_1
    setb CS
    clr CS
    mov power,#0c3h           //功率設定到最高發射
    lcall POWER_SET
    clr CS

```

```

jb MISO,$
mov a,#33h
lcall W_BYTE_900C
setb CS
clr CS
jb MISO,$
mov a,#37h
lcall W_BYTE_900C
setb CS
mov delh,#20
mov dell,#20
lcall delay
mov addr,#86h
lcall R_CONFIG_BYTE_900C
cjne a,#0ffh,CONFIG_900C
nop
ret

```

\*\*\*\*\*

```

//函數名      : TX_MODE_900C
//功能描述    : 將900C配置在發射模式
//調用函數    : W_BYTE_900C
//傳入參數    : 無
//返回值      : 無

```

\*\*\*\*\*

TX\_MODE\_900C:

```

clr CS
jb MISO,$
mov a,#36h
lcall W_BYTE_900C
setb CS
clr CS
jb MISO,$
mov a,#3bh
lcall W_BYTE_900C
setb CS
clr CS
jb MISO,$
mov a,#35h
lcall W_BYTE_900C
setb CS
ret

```

\*\*\*\*\*

```

//函數名      : RX_MODE_900C
//功能描述    : 將900C配置在接收模式

```

```
//調用函數   : W_BYTE_900C
//傳入參數   : 無
//返回值     : 無
//*****
```

```
RX_MODE_900C:
    clr CS
    jb MISO,$
    mov a,#36h
    lcall W_BYTE_900C
    setb CS
    clr CS
    jb MISO,$
    mov a,#3ah
    lcall W_BYTE_900C
    setb CS
    clr CS
    jb MISO,$
    mov a,#34h
    lcall W_BYTE_900C
    setb CS
    ret
```

```
//*****
//函數名     : IDLE_MODE_900C
//功能描述   : 將900C配置在空閒模式
//調用函數   : W_BYTE_900C
//傳入參數   : 無
//返回值     : 無
//*****
```

```
IDLE_MODE_900C:
    clr cs
    jb MISO,$
    mov a,#36h
    lcall W_BYTE_900C
    setb CS
    ret
```

```
//*****
//函數名     : DR_TEST
//功能描述   : 延時檢測900C的資料接收情況
//函數說明   :
//當900C處於接收模式時，如果超出一段時間沒有接收到資料（即DR為低）將設定接收
//空標誌位元，可以根據此標誌位元作程式跳轉，如果資料包太長或工作頻率大於12MHz
//此中參數可能要更改才能正常接收資料
//調用函數   : 無
//傳入參數   : 無
```

//返回值 : NULL, 置 1 表示在這一段時間內沒有收到資料

\*\*\*\*\*

DR\_TEST:

```
push 06h
push 07h
clr NULL
mov r7,#250
```

DR\_TEST\_1:

```
mov r6,#250
```

DR\_TEST\_2:

```
jb DR,DR_T_ED_1
mov r4,#10
djnz r4,$
djnz r6,DR_TEST_2
djnz r7,DR_TEST_1
```

DR\_T\_ED:

```
setb NULL
```

DR\_T\_ED\_1:

```
pop 07h
pop 06h
ret
```

\*\*\*\*\*

```
//函數名 : init_mcu
//功能描述 : 配置MCU
//函數說明 : 本配置視具體MCU而定
//調用函數 : 無
//傳入參數 : 無
//返回值 : 無
```

\*\*\*\*\*

init\_mcu:

```
mov xbr1,#40h
mov PCA0MD,#00h
mov P2MDOUT,#0eh
mov P1MDOUT,#4bh
mov P0MDOUT,#0e0h
mov OSCICN,#83h //時鐘基準確定為 12M
mov p2,#01h
mov p0,#20h
mov p1,#0f4h
ret
```

\*\*\*\*\*

```
// CONFIG_TABLE : 900C 配置參數表
// 每一排資料前一位元組為900C寄存器位址
// 後一位元組為寫入該寄存器的參數
```

```
//工作頻率為：868.3MHz 速率 250Kbps 頻道間隔 250k
//*****
```

```
CONFIG_TABLE:
```

- DB 0x0D,0x21
- DB 0x0E,0xc7
- DB 0x0F,0xe0
- DB 0x0A,0x00
- DB 0x0B,0x0b
- DB 0x0C,0x00
- DB 0x10,0x2d
- DB 0x11,0x3b
- DB 0x12,0x73
- DB 0x13,0x23
- DB 0x14,0x3B
- DB 0x15,0x00
- DB 0x22,0x10
- DB 0x21,0xb6
- DB 0x18,0x08
- DB 0x19,0x1d
- DB 0x1A,0x1C
- DB 0x1B,0xc7
- DB 0x1C,0x00
- DB 0x1D,0xb2
- DB 0x23,0xEA
- DB 0x24,0x2A
- DB 0x25,0x00
- DB 0x26,0x11
- DB 0x29,0x59
- DB 0x2C,0x88
- DB 0x2D,0x31
- DB 0x2E,0x0B
- DB 0x08,0x05
- DB 0x07,0x04
- DB 0x02,0x06
- DB 0x00,0x1B
- DB 0x09,0x00
- DB 0x06,0xFF

```
//*****
```

```
//基頻 906.000MHZ 工作速率 10Kbps 頻道間隔 100K
//*****
```

```
CONFIG_TABLE:
```

- DB 0x0D,0x22
- DB 0x0E,0xd8

```
DB 0x0F,0x9d //基頻 906.000 MHZ
DB 0x0A,0x00
DB 0x0B,0x06
DB 0x0C,0x00
DB 0x10,0xc8 //
DB 0x11,0x93 //工作速率 10K
DB 0x12,0x03
DB 0x13,0x21 //
DB 0x14,0xf8 //頻道間隔 100K
DB 0x15,0x34
DB 0x22,0x10
DB 0x21,0x56
DB 0x18,0x08
DB 0x19,0x16
DB 0x1A,0x6C
DB 0x1B,0x43
DB 0x1C,0x40
DB 0x1D,0x91
DB 0x23,0xa9
DB 0x24,0x2a
DB 0x25,0x00
DB 0x26,0x11
DB 0x29,0x59
DB 0x2C,0x81
DB 0x2D,0x35
DB 0x2E,0x0B
DB 0x08,0x05
DB 0x07,0x04
DB 0x02,0x06
DB 0x00,0x1B
DB 0x09,0x00
DB 0x06,0xFF
end
```