FT62F21X

Application note

目录

FT62F21X SPI 应用

1 SPI 应用说明

SPI是串行外设接口（Serial Peripheral Interface）的缩写。SPI，是一种高速的，全双工，同步的通信总线，以主从方式工作，这种模式通常有一个主设备和一个或多个从设备，需要至少4根线，事实上3根也可以（单向传输时）。也是所有基于SPI的设备共有的，它们是：

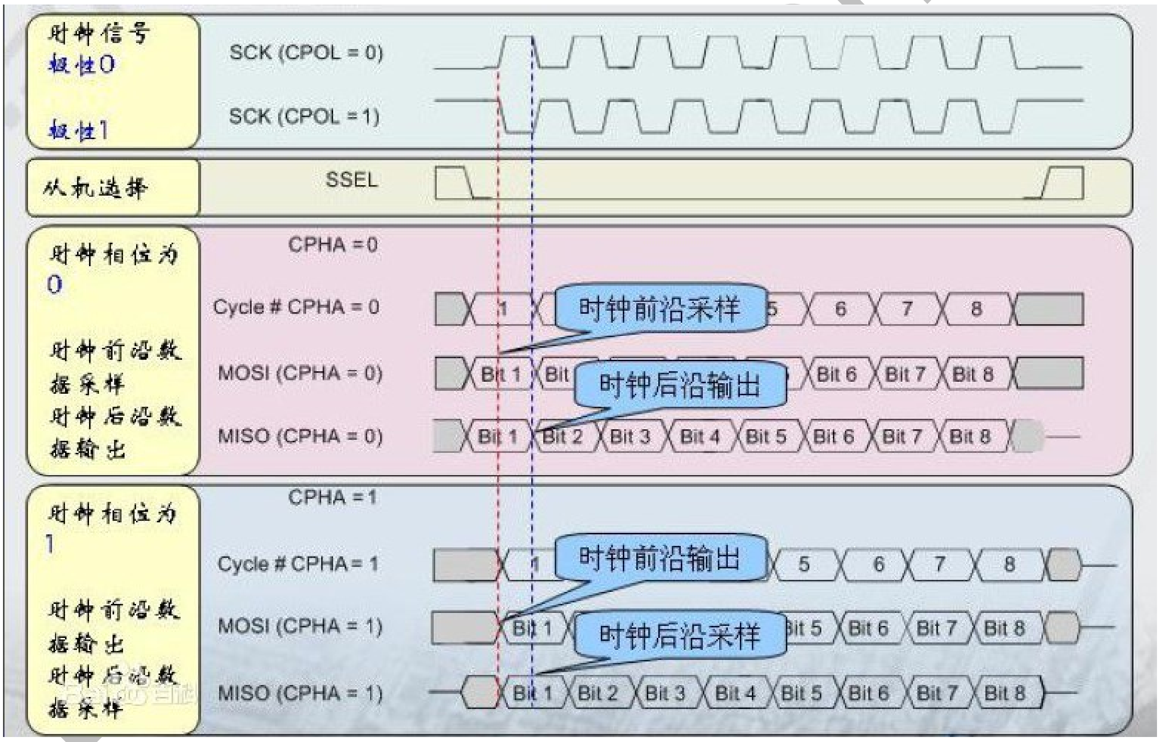
SDO/MOSI ------- 主设备数据输出，从设备数据输入;

SDI/MISO ------- 主设备数据输入，从设备数据输出;

SCLK ------------ 时钟信号，由主设备产生;

CS ---------------片选，从设备使能信号，由主设备控制。

SPI通信有4种不同的模式，不同的从设备可能在出厂是就是配置为某种模式，这是不能改变的；但我们的通信双方必须是工作在同一模式下，所以我们可以对我们的主设备的SPI模式进行配置，通过CPOL（时钟极性）和CPHA（时钟相位）来控制我们主设备的通信模式



Mode0：CPOL=0，CPHA=0

Mode1：CPOL=0，CPHA=1

Mode2：CPOL=1，CPHA=0

Mode3：CPOL=1，CPHA=1

本说明以IC FT62F21X与存储芯片25C64为示范。

有四种工作模式，本程序采用mode0的工作模式，四根数据线所对应的IO引脚：

```
#define   MISO            RA4

#define   MOSI            RA3

#define   SCK             RA2

#define   CS              RA1
```

2 应用范例

```
//=============================================================
;*   文件名：   ASM_FT62F21X_SPI.ASM
;*   功能：     FT62F21X_SPI 功能演示
;*   IC:        FT62F211        SOP8
;*   内部：     16M/4T
;*   说明：     该程序读取(25C64)0x12 地址的值,取反后存入 0x13 地址
;*
;*
;*                          FT62F211 SOP8
;*                         ----------------
;*   MISO---------|1(PA4)        (PA3)8|------------MOSI
;*   NC------------|2(TKCAP)    (PA0)7|------------NC
;*   VDD----------|3(VDD)        (PA1)6|------------CS
;*   GND----------|4(VSS)         (PA2)5|------------SCK
;*                         ----------------
//=============================================================
#INCLUDE    <FT62F21X.INC>;
;=============================================================
;RAM DEFINE
;=============================================================
    TEMP            EQU         0X40
    TEMP1           EQU         0X41
    TEMP2           EQU         0X42
    SPIDATTEMP      EQU         0X43
    count           EQU         0X44
    buff            EQU         0X45
    #define         f_ready     buff,0
    SPIDATA         EQU         0X46
    SPIADDRL        EQU         0X47
    SPIADDRH        EQU         0X48

    W_TMP           EQU         0X70
    S_TMP           EQU         0X71
;==============================================================
;CONSTANT DEFINE
;==============================================================
    INTCON_DEF      EQU     B'00000000'   ;禁止所有中断

    OSCCON_DEF      EQU     B'01110000'   ;16MHz

    WPUA_DEF        EQU     B'00001000'   ;弱上拉的开关，0-关，1-开

    TRISA_DEF       EQU     B'00001000'   ;输入输出设置，0-输出，1-输入

    PSRCA_DEF       EQU     B'00001111'   ;源电流设置最大

    PSINKA_DEF      EQU     B'00000011'   ;灌电流设置最大

    OPTION_DEF      EQU     B'00001000'   ;Bit3=1 WDT MODE,PS=000=1:1 WDT RATE
                                          ;Bit7(PAPU)=0 由 WPUA 决定是否上拉
;===============================================================
;USER DEFINE
;===============================================================
#DEFINE   MISO          PORTA,4
#DEFINE   MOSI          PORTA,3
#DEFINE   SCK           PORTA,2
#DEFINE   CS            PORTA,1
;==============================================================
;PROGRAM START
;==============================================================
    ORG         0x0000              ; 单片机复位向量入口
    LJUMP       RESTART             ; 跳转到主程序入口
    ORG         0x0004              ; 中断复位向量入口
    LJUMP       INT_PROGRAM
;==============================================================
;中断处理程序
;==============================================================
INT_PROGRAM:
    STR         W_TMP               ; 保存 W 寄存器
    SWAPR       STATUS,W            ; 保存 STATUS 寄存器
    STR         S_TMP

INT_RET:
    SWAPR       S_TMP,0
    STR         STATUS              ; 恢复 STATUS 寄存器
    SWAPR       W_TMP,1
```

```
        SWAPR       W_TMP,0             ; 恢复 W 寄存器
        RETI                            ; 中断返回
;==================================================================
;SYSTEM START
;==================================================================
RESTART:
        LCALL       INITIAL
        LCALL       init_25c64_io
;==================================================================
;主程序
;==================================================================
MAIN:
        LDWI        0X00
        BANKSEL     SPIADDRH
        STR         SPIADDRH
        LDWI        0X12
        BANKSEL     SPIADDRL
        STR         SPIADDRL
        LCALL       SPI_Read

        COMR        SPIDATTEMP,W
        STR         SPIDATA

        LDWI        0X00
        BANKSEL     SPIADDRH
        STR         SPIADDRH
        LDWI        0X13
        BANKSEL     SPIADDRL
        STR         SPIADDRL

        LCALL       SPI_Write

MAIN_LOOP:
        NOP
        LJUMP       MAIN_LOOP
;==================================================================
;系统初始化
;==================================================================
INITIAL:
        BANKSEL     OSCCON
        LDWI        OSCCON_DEF
        STR         OSCCON

        BANKSEL     INTCON
```

```
    LDWI        INTCON_DEF
    STR         INTCON

    BANKSEL     PORTA
    LDWI        0X10
    STR         PORTA

    BANKSEL     TRISA
    LDWI        TRISA_DEF
    STR         TRISA

    BANKSEL     WPUA
    LDWI        WPUA_DEF
    STR         WPUA

    BANKSEL     PSRCA
    LDWI        PSRCA_DEF
    STR         PSRCA

    BANKSEL     PSINKA
    LDWI        PSINKA_DEF
    STR         PSINKA

    BANKSEL     OPTION
    LDWI        OPTION_DEF
    STR         OPTION
;***************Clear   SRAM***************************************
    BCR         STATUS,PAGE
    LDWI        0X40
    STR         FSR
CLEAR_RAM_BANK0_LOOP:
    CLRR        INDF
    INCR        FSR,F
    LDWI        80H
    XORWR       FSR,W
    BTSS        STATUS,Z
    LJUMP       CLEAR_RAM_BANK0_LOOP
    RET
;=========================================================
;init_25c64_io
;=========================================================
init_25c64_io:
    BANKSEL     PORTA
    BSR         CS
```

```
        NOP
        BCR         SCK
        NOP
        BCR         MOSI
        RET
;=============================================================
;SPI_RW
;=============================================================
SPI_RW:
        CLRR        count
SPI_RW_LOOP:
        LDWI        0X08
        SUBWR       count,0
        BTSC        STATUS,0
        RET
        INCR        count,1
        BTSS        SPIDATTEMP,7
        LJUMP       $+3
        BSR         MOSI
        LJUMP       $+2
        BCR         MOSI

        NOP
        BCR         STATUS,0
        RLR         SPIDATTEMP,1
        BSR         SCK
        NOP
        NOP
        BTSS        MOSI
        LJUMP       $+3
        INCR        SPIDATTEMP,1
        LJUMP       $+2
        BCR         SPIDATTEMP,0
        NOP
        BCR         SCK
        LJUMP       SPI_RW_LOOP
;=============================================================
;WriteEnable
;=============================================================
WriteEnable:
        BANKSEL     PORTA
        BCR         CS
        LDWI        0X06
        STR         SPIDATTEMP
        LCALL       SPI_RW
        BSR         CS
        RET

;=====================================================================
;WriteDisable
;=====================================================================
WriteDisable:
        BANKSEL     PORTA
        BCR         CS
        LDWI        0X04
        STR         SPIDATTEMP
        LCALL       SPI_RW
        BSR         CS
        RET

;=====================================================================
;SPI_ReadStatus
;=====================================================================
SPI_ReadStatus:
        BCR         f_ready
        BCR         CS
        LDWI        0X05                    ;0x05 读取状态的命令字
        STR         SPIDATTEMP
        LCALL       SPI_RW
        LDWI        0X00
        STR         SPIDATTEMP
        LCALL       SPI_RW
        BSR         CS
        BTSC        SPIDATTEMP,0
        BSR         f_ready
        RET

;=====================================================================
;SPI_WriteStatus
;=====================================================================
SPI_WriteStatus:
        BCR         CS
        LDWI        0X01                    ;0X01 写入状态的命令字
        STR         SPIDATTEMP
        LCALL       SPI_RW
        LDWI        0X00
        STR         SPIDATTEMP
        LCALL       SPI_RW
        BSR         CS
        RET

;============================================================
;SPI_Read
;============================================================
SPI_Read:
        LCALL       SPI_ReadStatus
        BTSC        f_ready                 ;判断是否忙
        LJUMP       $-2
        BCR         CS

        LDWI        0X03                    ;发送读取命令
        STR         SPIDATTEMP
        LCALL       SPI_RW

        LDR         SPIADDRH,W              ;发送高地址
        STR         SPIDATTEMP
        LCALL       SPI_RW

        LDR         SPIADDRL,W              ;发送低地址
        STR         SPIDATTEMP
        LCALL       SPI_RW

        LDWI        0X00                    ;读出数据
        STR         SPIDATTEMP
        LCALL       SPI_RW
        BSR         CS
        RET

;============================================================
;SPI_Write
;============================================================
SPI_Write:
        LCALL       SPI_ReadStatus
        BTSC        f_ready                 ;判断是否忙
        LJUMP       $-2
        LCALL       WriteEnable
        BCR         CS

        LDWI        0X02                    ;发送写命令
        STR         SPIDATTEMP
        LCALL       SPI_RW
```

```
    LDR        SPIADDRH,W         ;发送高地址
    STR        SPIDATTEMP
    LCALL      SPI_RW

    LDR        SPIADDRL,W         ;发送低地址
    STR        SPIDATTEMP
    LCALL      SPI_RW

    LDR        SPIDATA,W          ;发送数据
    STR        SPIDATTEMP
    LCALL      SPI_RW

    LCALL      WriteDisable
    BSR        CS
    LCALL      SPI_ReadStatus
    BTSC       f_ready            ;判断是否忙
    LJUMP      $-2
    RET
;=================================================================
    END                           ; 汇编程序结束
```

Fremont Micro Devices (SZ) Limited

#5-8, 10/F, Changhong Building, Ke-Ji Nan 12 Road, Nanshan District, Shenzhen, Guangdong 518057
Tel: (86 755) 86117811
Fax: (86 755) 86117810

Fremont Micro Devices (Hong Kong) Limited

#16, 16/F, Blk B, Veristrong Industrial Centre, 34-36 Au Pui Wan Street, Fotan, Shatin, Hong Kong
Tel: (852) 27811186
Fax: (852) 27811144

Fremont Micro Devices (USA), Inc.

42982 Osgood Road Fremont, CA 94539
Tel: (1-510) 668-1321
Fax: (1-510) 226-9918

Web Site: http://www.fremontmicro.com/

* Information furnished is believed to be accurate and reliable. However, Fremont Micro Devices, Incorporated (BVI) assumes no responsibility for the consequences of use of such information or for any infringement of patents of other rights of third parties which may result from its use. No license is granted by implication or otherwise under any patent rights of Fremont Micro Devices, Incorporated (BVI). Specifications mentioned in this publication are subject to change without notice. This publication supersedes and replaces all information previously supplied. Fremont Micro Devices, Incorporated (BVI) products are not authorized for use as critical components in life support devices or systems without express written approval of Fremont Micro Devices, Incorporated (BVI). The FMD logo is a registered trademark of Fremont Micro Devices, Incorporated (BVI). All other names are the property of their respective own.