Antoine Wajntraub (/s/profile/0050X000007voE9QAI) (Customer) to ST Community (/s/profile/0050X000007vh8GQAQ) (Employee): asked a question.
Edited by STM Community July 31, 2018 at 4:40 PM (/s/question/0D50X00009Xke2hSAB/stm32f417-xpt2046-touch-screen)

STM32F417 + XPT2046 (touch screen)

Posted on February 23, 2017 at 22:57

Hello everyone !
To give you some context : I have a entire project generated by CubeMX on STM32F417, and now, I'm trying to make my touch screen work with it, trough SPI. So far, I can get the IRQ signal, so I detect when I have a touch, but I'm also trying to have the position of the touch, and this is the moment I have a problem.
Every pins are initialized by the code generated by CubeMX, and it's the same for my SPI_Handle.
I saw 'many' examples on the web, most of them they communicate with the functions SPI_I2S_Send/ReceiveData that is normally in the STM32F4XX_StdPeriph_Driver library. But CubeMX didn't generated this library, instead I have the HAL library, with the SPI file, and as I think that CubeMX is not bad, I think that if I don't have the first one, it's because I don't need it. Anyway, I'm trying to make it work with what I have, but I always got the same data, which is not good... Here is the interesting part of my code :

```
void Touch_GetTouch(uint16_t * xPos, uint16_t * yPos)

{

   uint32_t avgX = 0;

   uint32_t avgY = 0;

   float calX, calY;

   TOUCH_CS_RESET;

   avgX = Touch_Read(CHX); //0x90

   avgY = Touch_ReadY(CHY); //0xD0

   TOUCH_CS_SET;


   calX = aX * avgX + bX * avgY + dX;

   calY = aY * avgX + bY * avgY + dY;

   *xPos = calX;

   *yPos = calY;

}
uint16_t Touch_Read(uint8_t cmd)

{

   Touch_Lock();

   Touch_Write(cmd);

   LCD_Delay(10);

   Touch_Unlock();

   return Touch_Read();

}
uint16_t Touch_Read(void)

{

   uint32_t buf;

   HAL_StatusTypeDef res;

   uint8_t temp;
```

```
    uint8_t data = 0;

    /* Wait for SPI3 Tx buffer empty */
    while (hspi2.State != HAL_SPI_STATE_READY);

    res = HAL_SPI_Transmit(&hspi2, &data, 1, 500);

    if (res == HAL_OK) {
        while (hspi2.State != HAL_SPI_STATE_READY);
        res = HAL_SPI_Receive(&hspi2, &temp, 1, 500);
    }

    if (res == HAL_OK) {
        buf=temp<<8;
        LCD_Delay(1);
        while (hspi2.State != HAL_SPI_STATE_READY);
        res = HAL_SPI_Transmit(&hspi2, &data, 1, 500);
    }

    if (res == HAL_OK) {
        while (hspi2.State != HAL_SPI_STATE_READY);
        res = HAL_SPI_Receive(&hspi2, &temp, 1, 1000);
    }

    if (res == HAL_OK) {
        buf |= temp;
        buf>>=3;
        buf&=0xfff;
        return buf;
    }

    return buf;
}
void Touch_Write(uint8_t out)
{
    while (hspi2.State != HAL_SPI_STATE_READY);
    HAL_SPI_Transmit(&hspi2, &out, 1, 500);
    while (hspi2.State != HAL_SPI_STATE_READY);
    HAL_SPI_Receive(&hspi2, &out, 1, 500);
}
```

The Touch_Write and Touch_Read are inspired from some projects I saw, like this one :

http://fabioangeletti.altervista.org/blog/stm32f4-discovery-lcd-touchscreen/ (http://fabioangeletti.altervista.org/blog/stm32f4-discovery-lcd-touchscreen/)

I'm a beginner with the SPI, and most of the STM libraries, so I'm probably missing something stupid and simple, but after two days of trying to find that positions, I can't find out what it is.
Thanks for your help, if you need any infos, no problem !
Antoine

#xpt2046 #stm32f4

fabioangeletti.altervista.org (http://fabioangeletti.altervista.org/blog/stm32f4-discovery-lcd-touchscreen/)
http://fabioangeletti.altervista.org/blog/stm32f4-discovery-lcd-touchscreen/ (http://fabioangeletti.altervista.org/blog/stm32f4-discovery-lcd-touchscreen/)

STM32 MCUs (/s/topic/0TO0X000000BSqSWAW/) STM32F4 (/s/topic/0TO0X000000BTnoWAG/)

2.16K views

Like Share

This question is closed.