

Hi, I just received my STM32F429 discovery board and I tried to implement basic SPI but I'm stumped as to why the logic analyzer output was different. Can anybody point out where I was wrong. Here is the code.

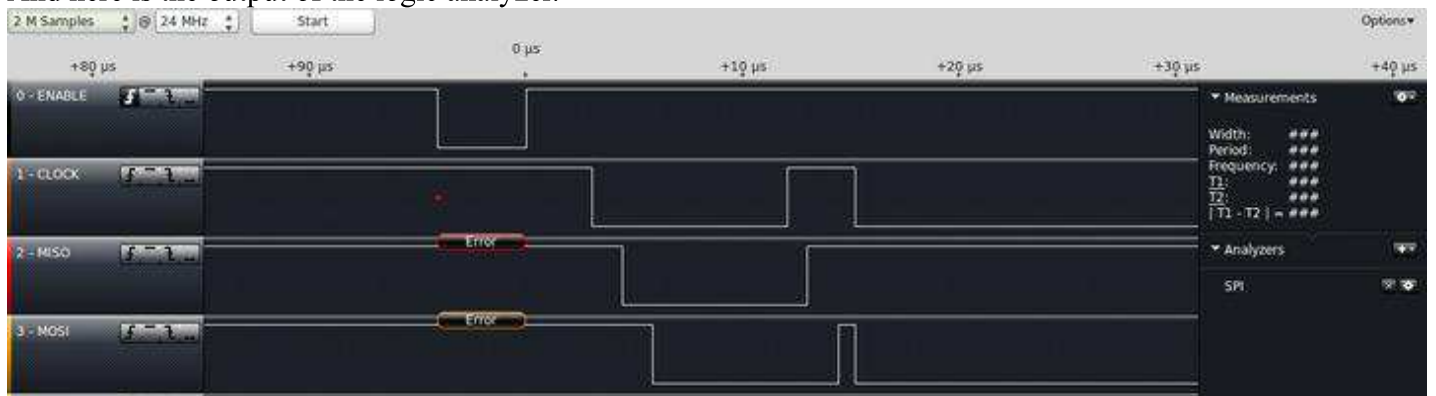
```
001. #include <stm32f4xx.h>
002. #include <stm32f4xx_gpio.h>
003. #include <stm32f4xx_rcc.h>
004. #include <stm32f4xx_spi.h>
005.
006.
007. void spi1_init(void);
008. void spiSend(uint8_t data);
009. uint8_t spiRead(void);
010. void csInit(void);
011.
012. uint8_t txBuffer[4] = {0x00, 0x01, 0x02, 0x03};
013. uint8_t rxBuffer[4];
014.
015. void Delay(__IO uint32_t nCount){
016.     while(nCount-->0)
017.     {
018.     }
019. }
020.
021. int main(void){
022.
023.     int i,j;
024.
025.     csInit();
026.     spi1_init();
027.
028.     for(i = 0; i < 4; i++){
029.         GPIO_WriteBit(GPIOE, GPIO_Pin_3, 0);
030.         spiSend(txBuffer[i]);
031.         rxBuffer[i] = spiRead();
032.         GPIO_WriteBit(GPIOE, GPIO_Pin_3, 1);
033.     }
034.
035.     while(1){
036.     }
037.
038. }
039.
040. void csInit(void){
041.
042.     GPIO_InitTypeDef GPIO_InitStructure;
043.     RCC_AHB1PeriphClockCmd(RCC_AHB1Periph_GPIOE, ENABLE);
044.
045.     /* Configure chip select pin to PE3 */
046.     GPIO_InitStructure.GPIO_Pin = GPIO_Pin_3;
047.     GPIO_InitStructure.GPIO_Mode = GPIO_Mode_OUT;
048.     GPIO_Init(GPIOE, &GPIO_InitStructure);
049.
050.     GPIO_WriteBit(GPIOE, GPIO_Pin_3, 1); /* Set chip select high */
051. }
052. void spi1_init(void){
053.     GPIO_InitTypeDef GPIO_InitStructure;
054.     SPI_InitTypeDef SPI_InitStructure;
055.
056.     RCC_AHB1PeriphClockCmd(RCC_AHB1Periph_GPIOA, ENABLE);
057.     RCC_APB2PeriphClockCmd(RCC_APB2Periph_SPI1, ENABLE);
058.
059.     /* Configure GPIO pins for SPI1
060.      * PA5 = SCK
061.      * PA6 = MISO
```

```

062.     * PA7 = MOSI
063.     */
064.
065.     GPIO_InitStructure.GPIO_Pin = GPIO_Pin_7 | GPIO_Pin_6 | GPIO_Pin_5;
066.     GPIO_InitStructure.GPIO_Mode = GPIO_Mode_AF;
067.     GPIO_InitStructure.GPIO_OType = GPIO_OType_PP;
068.     GPIO_InitStructure.GPIO_Speed = GPIO_Speed_50MHz;
069.     GPIO_InitStructure.GPIO_PuPd = GPIO_PuPd_NOPULL;
070.     GPIO_Init(GPIOA, &GPIO_InitStructure);
071.
072.     /* Connect SPI1 pins to SPI alternate function */
073.     GPIO_PinAFConfig(GPIOA, GPIO_PinSource5, GPIO_AF_SPI1);
074.     GPIO_PinAFConfig(GPIOA, GPIO_PinSource6, GPIO_AF_SPI1);
075.     GPIO_PinAFConfig(GPIOA, GPIO_PinSource7, GPIO_AF_SPI1);
076.
077.     /* Configure SPI1 */
078.     SPI_InitStructure.SPI_Direction = SPI_Direction_2Lines_FullDuplex;
079.     SPI_InitStructure.SPI_Mode = SPI_Mode_Master;
080.     SPI_InitStructure.SPI_DataSize = SPI_DataSize_8b;
081.     SPI_InitStructure.SPI_CPOL = SPI_CPOL_Low;
082.     SPI_InitStructure.SPI_CPHA = SPI_CPHA_1Edge;
083.     SPI_InitStructure.SPI_NSS = SPI_NSS_Soft;
084.     SPI_InitStructure.SPI_BaudRatePrescaler = SPI_BaudRatePrescaler_64;
085.     SPI_InitStructure.SPI_FirstBit = SPI_FirstBit_MSB;
086.     SPI_InitStructure.SPI_CRCPolynomial = 7;
087.
088.     SPI_Init(SPI1, &SPI_InitStructure);
089.     SPI_Cmd(SPI1, ENABLE);
090. }
091.
092. void spiSend(uint8_t data){
093.     while(!SPI_I2S_GetFlagStatus(SPI1, SPI_I2S_FLAG_TXE));
094.     SPI_I2S_SendData(SPI1, data);
095.
096. }
097.
098. uint8_t spiRead(void){
099.     while(SPI_I2S_GetFlagStatus(SPI1, SPI_I2S_FLAG_RXNE) == RESET);
100.     return SPI_I2S_ReceiveData(SPI1);
101.
102. }
103.
104. #ifdef USE_FULL_ASSERT
105. void assert_failed(uint8_t * file, uint32_t line){
106.     while(1)
107.     {
108.
109.     }
110. }
111. #endif

```

And here is the output of the logic analyzer.



Thank you very much!