```
/* p8_1.c: Using SPI1 to send A to Z characters
 *
 * SPI1 is configured as master with software slave select.
 * Clock rate is set to 1 MHz. Polarity/Phase are 0, 0
 *
 * PA7  MOSI
 * PA5  SCK
 * PA4  SS
 *
 * This program was tested with Keil uVision v5.24a with DFP v2.11.0
 */

#include "stm32f4xx.h"

void delayMs(int n);
void SPI1_init(void);
void SPI1_write(unsigned char data);

int main(void) {
    char c;

    SPI1_init();

    while(1) {
        for (c = 'A'; c <= 'Z'; c++) {
            SPI1_write(c);      /* write the letter through SPI1 */
            delayMs(100);
        }
    }
}

/* enable SPI1 and associated GPIO pins */
void SPI1_init(void) {
    RCC->AHB1ENR |= 1;              /* enable GPIOA clock */
    RCC->APB2ENR |= 0x1000;         /* enable SPI1 clock */
```

```
    /* PORTA 5, 7 for SPI1 MOSI and SCLK */
    GPIOA->MODER &= ~0x0000CC00;    /* clear pin mode */
    GPIOA->MODER |=  0x00008800;    /* set pin alternate mode */
    GPIOA->AFR[0] &= ~0xF0F00000;   /* clear alt mode */
    GPIOA->AFR[0] |=  0x50500000;   /* set alt mode SPI1 */

    /* PORTA4 as GPIO output for SPI slave select */
    GPIOA->MODER &= ~0x00000300;    /* clear pin mode */
    GPIOA->MODER |=  0x00000100;    /* set pin output mode */

    SPI1->CR1 = 0x31C;             /* set the Baud rate, 8-bit data frame */
    SPI1->CR2 = 0;
    SPI1->CR1 |= 0x40;              /* enable SPI1 module */
}

/* This function enables slave select, writes one byte to SPI1,
   wait for transmission complete and deassert slave select. */
void SPI1_write(unsigned char data) {
    while (!(SPI1->SR & 2)) {}      /* wait until Transfer buffer Empty */
    GPIOA->BSRR = 0x00100000;       /* assert slave select */
    SPI1->DR = data;                /* write data */
    while (SPI1->SR & 0x80) {}      /* wait for transmission done */
    GPIOA->BSRR = 0x00000010;       /* deassert slave select */
}

/* 16 MHz SYSCLK */
void delayMs(int n) {
    int i;
    for (; n > 0; n--)
        for (i = 0; i < 3195; i++) ;
}
```