

ใบงานครั้งที่ 10

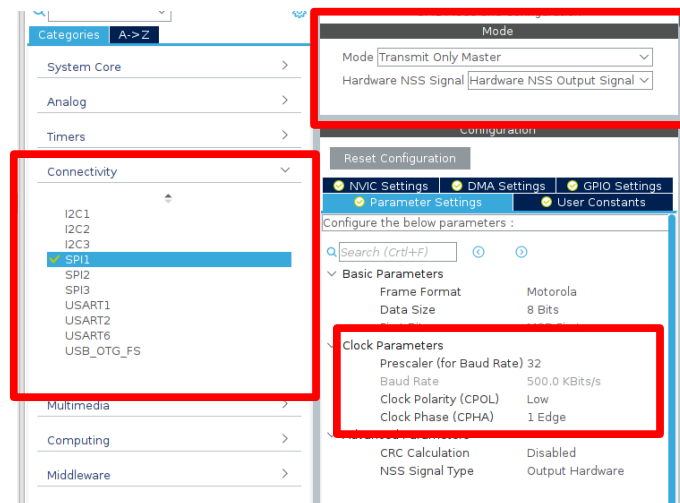
Serial Peripheral Interface (SPI)

อุปกรณ์ที่ใช้ในการทดลอง

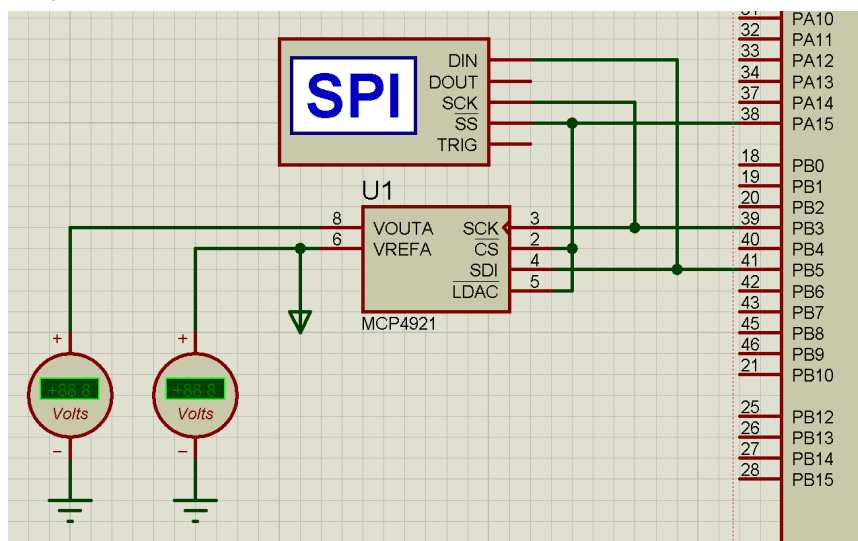
1. STM32CubeIDE
2. Proteus 8.9 SP 2

ขั้นตอนการทดลอง

1. ใน stm32cubeide ให้เลือกไปที่ connectivity → SPI1 → เลือก
 - Mode เป็น Transmit Only Master เนื่องจากในการทดลองนี้จะทำการส่งข้อมูลจาก STM32F401 ไปที่ MCP4921 เพียงอย่างเดียวไม่มีการอ่านข้อมูลกลับเข้ามา
 - Hardware NSS Signal เลือกเป็น Hardware NSS Output Signal
 - ในส่วนของ Prescaler (for Baud Rate) เลือกค่าเป็น 32 เพื่อทำการหาร clock ให้เหลือความเร็วในการรับส่งข้อมูลที่ 500KBits/s



2. ต่อดิจตามรูปด้านล่าง



3. โค้ดของโปรแกรมที่สร้างมาอัตโนมัติในส่วนที่เกี่ยวข้องกับ SPI

```

157  */
158  static void MX_SPI1_Init(void)
159  {
160
161      /* USER CODE BEGIN SPI1_Init 0 */
162
163      /* USER CODE END SPI1_Init 0 */
164
165      /* USER CODE BEGIN SPI1_Init 1 */
166
167      /* USER CODE END SPI1_Init 1 */
168      /* SPI parameter configuration*/
169      hspi1.Instance = SPI1;
170      hspi1.Init.Mode = SPI_MODE_MASTER;
171      hspi1.Init.Direction = SPI_DIRECTION_2LINES;
172      hspi1.Init.DataSize = SPI_DATASIZE_8BIT;
173      hspi1.Init.CLKPolarity = SPI_POLARITY_LOW;
174      hspi1.Init.CLKPhase = SPI_PHASE_1EDGE;
175      hspi1.Init.NSS = SPI_NSS_HARD_OUTPUT;
176      hspi1.Init.BaudRatePrescaler = SPI_BAUDRATEPRESCALER_32;
177      hspi1.Init.FirstBit = SPI_FIRSTBIT_MSB;
178      hspi1.Init.TIMode = SPI_TIMODE_DISABLE;
179      hspi1.Init.CRCCalculation = SPI_CRCCALCULATION_DISABLE;
180      hspi1.Init.CRCPolynomial = 10;
181      if (HAL_SPI_Init(&hspi1) != HAL_OK)
182      {
183          Error_Handler();
184      }
185      /* USER CODE BEGIN SPI1_Init 2 */
186
187      /* USER CODE END SPI1_Init 2 */
188
189  }
190

```

4. เพิ่มโค้ดในส่วนของตัวแปรเพื่อใช้ในการส่งข้อมูล

```

04  // Uncomment the application entry point.
05  * @retval int
06  */
07  int main(void)
08  {
09      /* USER CODE BEGIN 1 */
10      unsigned char data[2] = {0x38, 0x00};
11      unsigned int raw_dac = 0;
12
13      /* USER CODE END 1 */
14
15      /* MCU Configuration-----*/
16
17      /* Reset of all peripherals, Initializes the Flash interface and the Systick. */
18      HAL_Init();
19
20      /* USER CODE BEGIN Init */

```

5. โค้ดในส่วนของการส่งข้อมูลให้กับ MCP4921

```

99      /* USER CODE BEGIN WHILE */
100     while (1)
101     {
102         /* USER CODE END WHILE */
103
104         /* USER CODE BEGIN 3 */
105         raw_dac += 32;
106         if (raw_dac > 4096) raw_dac = 0;
107         data[0] = 0x30 | ((raw_dac >> 8) & 0x0F);
108         data[1] = raw_dac & 0xFF;
109         HAL_SPI_Transmit(&hspi1, data, 2, HAL_MAX_DELAY);
110         HAL_Delay(500);
111     }
112     /* USER CODE END 3 */

```

6. ข้อมูลเรจิสเตอร์ของ MAP4921 ที่เกี่ยวข้อง

REGISTER 5-1: WRITE COMMAND REGISTER

Upper Half:							
W-x	W-x	W-x	W-0	W-x	W-x	W-x	W-x
A/B	BUF	GA	SHDN	D11	D10	D9	D8
bit 15							

Lower Half:							
W-x	W-x	W-x	W-x	W-x	W-x	W-x	W-x
D7	D6	D5	D4	D3	D2	D1	D0
bit 7							

- bit 15 **A/B**: DAC_A or DAC_B Select bit
1 = Write to DAC_B
0 = Write to DAC_A
- bit 14 **BUF**: V_{REF} Input Buffer Control bit
1 = Buffered
0 = Unbuffered
- bit 13 **GA**: Output Gain Select bit
1 = 1x (V_{OUT} = V_{REF} * D/4096)
0 = 2x (V_{OUT} = 2 * V_{REF} * D/4096)
- bit 12 **SHDN**: Output Power Down Control bit
1 = Output Power Down Control bit
0 = Output buffer disabled, Output is high impedance
- bit 11-0 **D11:D0**: DAC Data bits
12 bit number "D" which sets the output value. Contains a value between 0 and 4095.

คำถามท้ายใบงาน

1. ให้นักเขียนโปรแกรมและเพิ่มอุปกรณ์เพื่อให้ได้ผลตามนี้

- มีปุ่มกด 2 ปุ่ม โดยให้ปุ่มที่ 1 เมื่อกดแล้วจะเพิ่มแรงดันไฟฟ้าที่ MCP4921 ครั้งละ 0.5V / ปุ่มที่ 2 เมื่อกดแล้วจะลดแรงดันไฟฟ้าครั้งละ 0.5 V

- ค่าแรงดันไฟฟ้าที่ MCP4921 ปลอยออกมาในช่วงเวลานั้นๆ จะถูกแสดงไปที่ Virtual Terminal เช่น

=> Current MCP4921 Voltage: 3.5 V