**STM32F401RE: SPI1 Sawtooth Waveform Generation (16-bit) for LTC1661 DAC**

**Overview**

This project configures **SPI1 as a Master** on the **STM32F401RE**, generating a **sawtooth waveform** using the **LTC1661 DAC** via **SPI1 communication** at **1 MHz**.

**Features**

* **SPI1 Configured in Master Mode (Mode 0, CPOL = 0, CPHA = 0)**
* **Sends 10-bit Sawtooth Waveform Data over SPI1 (16-bit frame format)**
* **Baud Rate: 1 MHz (fPCLK / 16)**
* **Manual SS Control for Stable Communication**

**Hardware Setup**

**SPI1 Pin Configuration**

| **Signal** | **STM32F401RE Pin** | **Description** |
| --- | --- | --- |
| **SPI1 SCK (Clock)** | **PA5** | SPI1 Clock Output |
| **SPI1 MOSI (Master Out, Slave In)** | **PA7** | SPI1 Data Output |
| **SPI1 SS (Slave Select)** | **PA4** | Manual Control (GPIO Output) |

**Wiring for SPI1 to LTC1661 DAC**

| **STM32F401RE (Master)** | **LTC1661 DAC (Slave)** |
| --- | --- |
| **PA5 (SCK)** | **SCK** |
| **PA7 (MOSI)** | **DIN** |
| **PA4 (SS)** | **CS/SS** |
| **GND** | **GND** |

**Software Explanation**

**SPI1 Master Initialization**

* **PA5 (SCK) and PA7 (MOSI) configured as SPI Alternate Function (AF5).**
* **PA4 (SS) manually controlled as GPIO Output.**
* **SPI Mode 0 (CPOL = 0, CPHA = 0) ensures correct timing.**
* **Baud rate set to 1 MHz using fPCLK / 16.**
* **16-bit data format is enabled for proper transmission.**

**SPI1 Data Transmission**

* **SS (PA4) goes LOW before transmitting a character.**
* **Data is written to SPI1->DR, and we wait for the TX buffer to be empty.**
* **SS (PA4) goes HIGH after transmission is complete.**
* **A sawtooth waveform is continuously sent with a delay of 1ms.**

**Sawtooth Waveform Generation**

* **Iterates through 1024 values (10-bit resolution) to create a smooth waveform.**
* **Ensures proper bit shifting for the LTC1661 DAC.**
* **Includes a delayMs(1); to control waveform frequency.**

**Project Structure**

.

├── Inc/

│ └── stm32f4xx.h // CMSIS/Device headers

├── Src/

│ └── main.c // Contains SPI1 communication & waveform logic

└── README.md // This file

**Building and Uploading**

**Using Keil uVision / STM32CubeIDE**

1. **Open Keil uVision or STM32CubeIDE**.
2. **Create a new project** for STM32F401RE.
3. **Copy main.c into the Src/ directory**.
4. **Compile and Flash the project** to **Nucleo-F401RE**.

**Usage**

**1️⃣ Flash the code to the STM32F401RE**

* Use **Keil uVision** or **STM32CubeIDE**.

**2️⃣ Connect SPI1 to LTC1661 DAC**

* **Wire PA5, PA7, and PA4 to the SPI slave device.**
* Ensure the **GNDs are connected**.

**3️⃣ Observe Sawtooth Waveform**

* Use an **oscilloscope** to check the **DAC output voltage.**
* A **rising sawtooth waveform** should be visible.

**Troubleshooting**

**🔴 No SPI Transmission?**

✅ **Ensure PA5, PA7, and PA4 are correctly connected.**  
✅ **Confirm SPI1 is enabled (SPI1->CR1 |= (1U << 6);).**  
✅ **Check that SS (PA4) is LOW during transmission.**

**⚠️ Wrong DAC Output?**

✅ **Ensure SPI settings match between master & LTC1661 DAC.**  
✅ **Check SPI Clock Polarity/Phase (CPOL = 0, CPHA = 0).**

**⏳ Want Faster SPI Speed?**

✅ Change **Baud Rate Prescaler** (SPI1->CR1 |= (0U << 3);) for fPCLK/2.  
✅ Reduce **delay between values (delayMs(1);).**

**License**

This project is licensed under the **MIT License**.  
You are free to **modify, distribute, and use** the code.

**References**

📄 [STM32F401RE Datasheet](https://www.st.com/en/microcontrollers-microprocessors/stm32f401re.html)  
📄 [STM32 Reference Manual (RM0368)](https://www.st.com/resource/en/reference_manual/dm00096844.pdf)  
📄 [SPI Programming Guide](https://www.st.com/resource/en/programming_manual/dm00245755.pdf)

**🚀 Summary**

✅ **SPI1 configured as Master (PA5 SCK, PA7 MOSI, PA4 SS).**  
✅ **Sends 10-bit sawtooth values to LTC1661 DAC over SPI1 (16-bit frame format).**  
✅ **Manually controls SS (PA4) for stable SPI communication.**  
✅ **Generates a smooth analog waveform using SPI.**

📡 **Enjoy sawtooth waveform generation with STM32F401RE! 🎯**