**STM32F401RE: I2C1 Byte Write to DS1337 RTC**

**Overview**

This project configures **I2C1 as a Master** on the **STM32F401RE**, communicating with the **DS1337 Real-Time Clock (RTC)** to enable the **1 Hz output signal**.

**Features**

* **I2C1 Configured in Master Mode (Standard Mode, 100 kHz)**
* **Sends a Byte to DS1337 RTC to Enable 1 Hz Output**
* **Uses Software-Generated Start and Stop Conditions**
* **PB8 (SCL) and PB9 (SDA) configured for I2C1**
* **Peripheral Clock: 16 MHz**

**Hardware Setup**

**I2C1 Pin Configuration**

| **Signal** | **STM32F401RE Pin** | **Description** |
| --- | --- | --- |
| **I2C1 SCL (Clock)** | **PB8** | I2C1 Clock Output |
| **I2C1 SDA (Data)** | **PB9** | I2C1 Data Line |

**Wiring for I2C1 to DS1337 RTC**

| **STM32F401RE (Master)** | **DS1337 RTC (Slave)** |
| --- | --- |
| **PB8 (SCL)** | **SCL** |
| **PB9 (SDA)** | **SDA** |
| **GND** | **GND** |
| **VCC (3.3V)** | **VCC** |

**Software Explanation**

**I2C1 Master Initialization**

* **PB8 (SCL) and PB9 (SDA) configured as Alternate Function (AF4).**
* **I2C1 set to Standard Mode (100 kHz) with Peripheral Clock = 16 MHz.**
* **I2C1 Software Reset and Proper Enable Sequence Followed.**

**I2C1 Data Transmission**

* **Sends Start Condition before transmitting data.**
* **Writes DS1337 RTC Slave Address (0x68) + Memory Register Address (0x0E).**
* **Transmits Data Byte (0x00) to Enable 1 Hz Output.**
* **Generates Stop Condition to complete the transaction.**

**Expected Behavior**

* **I2C1 writes to DS1337 RTC at address 0x68 via PB8 (SCL) & PB9 (SDA).**
* **Sends command to turn on the 1 Hz output at register 0x0E.**
* **No errors or acknowledgement checks implemented (simplified version).**

**Project Structure**

.

├── Inc/

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

├── Src/

│ └── main.c // Contains I2C1 communication 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 I2C1 to DS1337 RTC**

* **Wire PB8 (SCL) and PB9 (SDA) to the RTC.**
* Ensure the **GND and VCC (3.3V) connections** are correct.

**3️⃣ Check 1 Hz Signal Output**

* Connect an **oscilloscope** to the **1 Hz SQW/OUT pin** of DS1337.
* The signal should toggle at **1 Hz frequency** if the write was successful.

**Troubleshooting**

**🔴 No I2C Communication?**

✅ **Ensure PB8 (SCL) and PB9 (SDA) are correctly connected.**  
✅ **Confirm I2C1 is enabled (I2C1->CR1 |= (1U << 0);).**  
✅ **Check that pull-up resistors (4.7kΩ) are connected to SDA/SCL.**

**⚠️ RTC Not Responding?**

✅ **Ensure the correct I2C Slave Address (0x68) is used.**  
✅ **Check I2C Clock Polarity/Phase (Standard Mode).**

**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)  
📄 [I2C Programming Guide](https://www.st.com/resource/en/programming_manual/dm00245755.pdf)

**🚀 Summary**

✅ **I2C1 configured as Master (PB8 SCL, PB9 SDA).**  
✅ **Writes to DS1337 RTC using I2C1 at 100 kHz.**  
✅ **Proper Start/Stop condition generation.**  
✅ **Ensures stable I2C communication using correct flag checks.**

📡 **Enjoy I2C communication with STM32F401RE! 🎯**