

# Zegar czasu rzeczywistego DS1307(I2C)

Wyciąg z noty katalogowej

Polecenia do wykonania w laboratorium

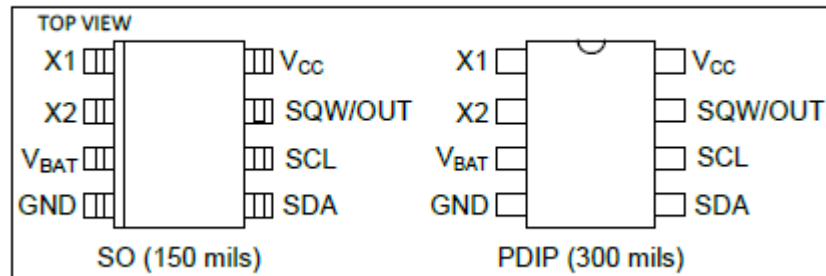
# DS1307

## 64 x 8 Serial Real-Time Clock

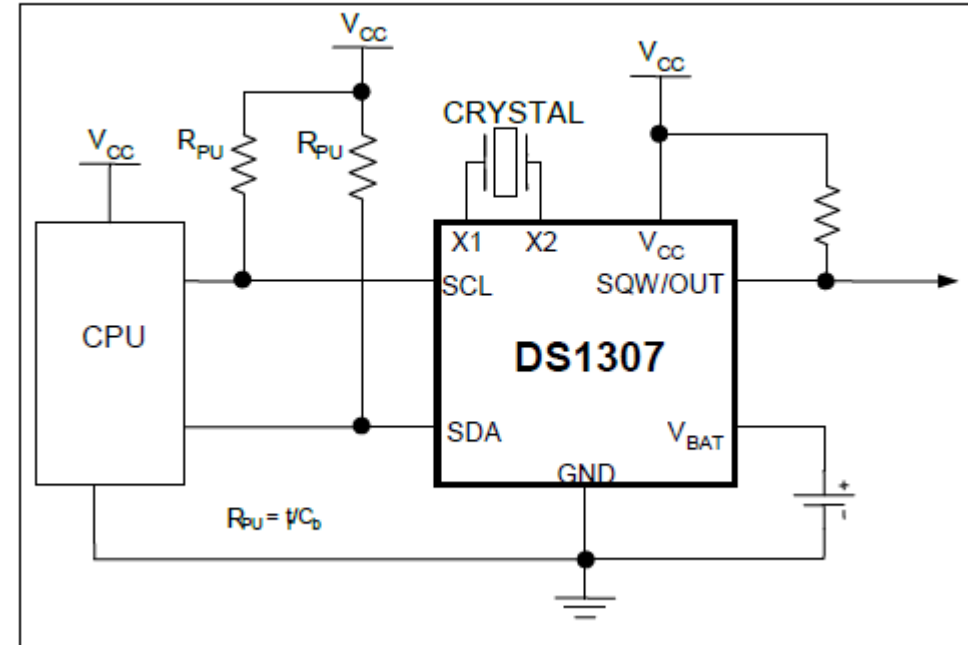
### GENERAL DESCRIPTION

The DS1307 serial real-time clock (RTC) is a low-power, full binary-coded decimal (BCD) clock/calendar plus 56 bytes of NV SRAM. Address and data are transferred serially through an I<sup>2</sup>C, bidirectional bus. The clock/calendar provides seconds, minutes, hours, day, date, month, and year information. The end of the month date is automatically adjusted for months with fewer than 31 days, including corrections for leap year. The clock operates in either the 24-hour or 12-hour format with AM/PM indicator. The DS1307 has a built-in power-sense circuit that detects power failures and automatically switches to the backup supply. Timekeeping operation continues while the part operates from the backup supply.

### PIN CONFIGURATIONS



### TYPICAL OPERATING CIRCUIT



### PIN DESCRIPTION

- |                  |                                |
|------------------|--------------------------------|
| V <sub>CC</sub>  | - Primary Power Supply         |
| X1, X2           | - 32.768kHz Crystal Connection |
| V <sub>BAT</sub> | - +3V Battery Input            |
| GND              | - Ground                       |
| SDA              | - Serial Data                  |
| SCL              | - Serial Clock                 |
| SQW/OUT          | - Square Wave/Output Driver    |

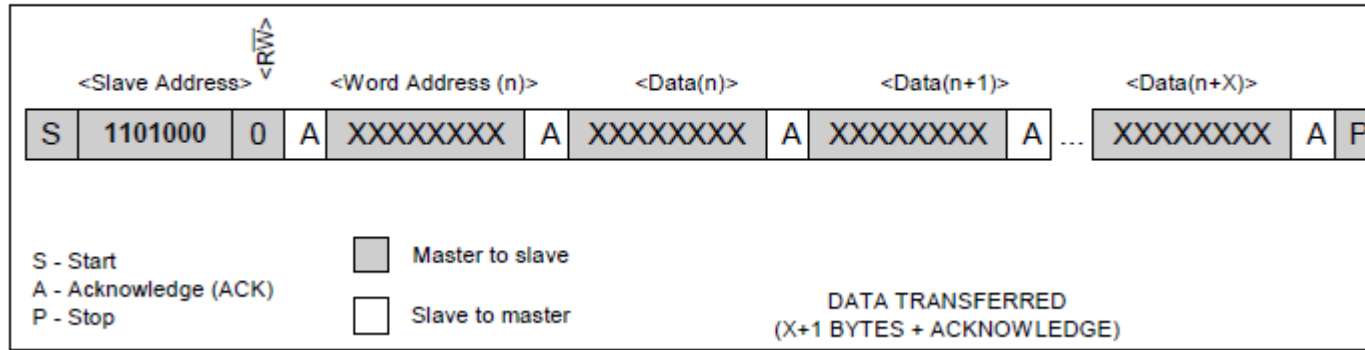
## CLOCK AND CALENDAR

Bit 7 of Register 0 is the clock halt (CH) bit. When this bit is set to 1, the oscillator is disabled. When cleared to 0, the oscillator is enabled. On first application of power to the device the time and date registers are typically reset to 01/01/00 01 00:00:00 (MM/DD/YY DOW HH:MM:SS). The CH bit in the seconds register will be set to a 1.

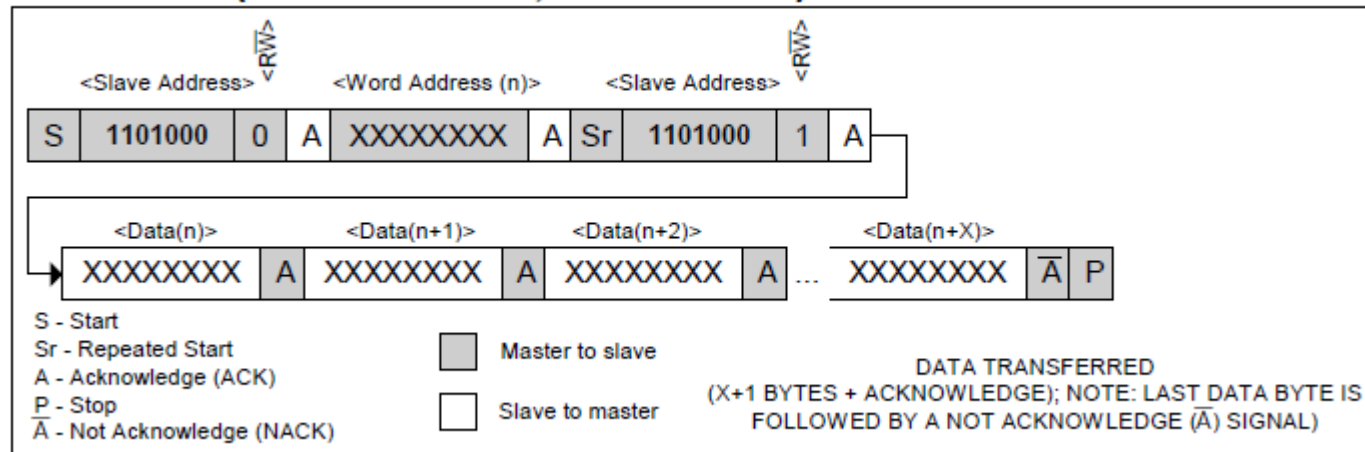
### Timekeeper Registers

ADDRESS	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0	FUNCTION	RANGE
00h	CH	10 Seconds			Seconds				Seconds	00–59
01h	0	10 Minutes			Minutes				Minutes	00–59
02h	0	12	10 Hour	10 Hour	Hours				Hours	1–12 +AM/PM 00–23
		24	PM/ AM							
03h	0	0	0	0	0	DAY			Day	01–07
04h	0	0	10 Date		Date				Date	01–31
05h	0	0	0	10 Month	Month				Month	01–12
06h	10 Year				Year				Year	00–99
07h	OUT	0	0	SQWE	0	0	RS1	RS0	Control	—
08h–3Fh									RAM 56 x 8	00h–FFh

## Data Write—Slave Receiver Mode



## Data Read (Write Pointer, Then Read)—Slave Receive and Transmit



## Polecenia:

1. Skonfiguruj interfejs I2C w środowisku STM32CubeIDE.
2. Podłącz linie SCL, SDA, VCC, GND. Nie podłączaj linii SQW (wyprowadzenie przebiegu prostokątnego) ani zasilania VBAT (brak baterii). Na wyprowadzeniach X1, X2 został już podłączony kwarc zegarkowy 32,768 kHz umożliwiający dokładne odliczanie czasu rzeczywistego.
3. Uruchom oscylator zegarkowy. W tym celu wyzeruj najstarszy bit w rejestrze 0 (bit CH)
4. Odczytuj w pętli rejestr sekund i odczytaną wartość obserwuj w Live Expressions STM32CubeIDE. Odczytana wartość ma format BCD, więc wyświetlane znaki nie odpowiadają kolejnym wartościom sekund od 0 do 59.
5. Zamień odczytane sekundy z formatu BCD na format ASCII. Ponownie obserwuj zmienność sekund – tym razem prawidłową prezentację od 0 do 59.
6. Do pętli odczytującej dodawaj po kolei: minuty, godziny, dni miesiąca, miesiące i lata – wyświetlaj godzinę i datę w postaci sformatowanej w interfejsie Serial Wire Viewer STM32CubeIDE.
7. Zapisz w rejestrach daty i czasu dane początkowe odpowiadające aktualnej dacie i czasowi.

Do zapisu rejestrów zegara można użyć funkcji z biblioteki HAL:

```
HAL_I2C_Master_Transmit (.....)
```

lub

```
HAL_I2C_Memory_Write(....)
```

Do odczytu rejestrów zegara można użyć funkcji z biblioteki HAL:

```
HAL_I2C_Master_Transmit(przełącz 2 bajty jako argumenty) + HAL_I2C_Master_Receive(..)
```

lub

```
HAL_I2C_Memory_Read(....).
```