

G1 Programming A DS1077 Oscillator

Jesse Larson, Jing Lu, and Qingqing Liu

The goal of this app note:

To give you some help to successfully integrate this chip easily into your project

Introduction:

This chip can generate a Digital clock signal from 8.1 KHz to 133 MHz these values depend on the base clock frequency you order on the chip. It is available in 3.3 volts and 5 volts, and must be programmed via I2C. I got my chip on a break out board from sparkfun.

Check if your desired frequency can be made:

Attached is a excel spread sheet provided my maxim integrated it is called DS10775VFREQ.xls if you can generate a frequency close enough to the one you desire go ahead and order the chip, with the base clk frequency you want, however only order it if you have a chip that can communicate with it via 5v I2C, I used a 5v arduino for my oscillator. There are ways to get a 3.3 volt chip to talk to this 5v I2C chip using a level shifter made out of some n enhancement mosfets in a pinch. But I still recommend using a 5 volt arduino instead.

Programming the oscillator

Things you need

- Arduino software
- Arduino chip
- Your DS1077 soldered and connected via SDA, and SCL lines to your arduino

Grab the code in from app note and move it into your arduino project

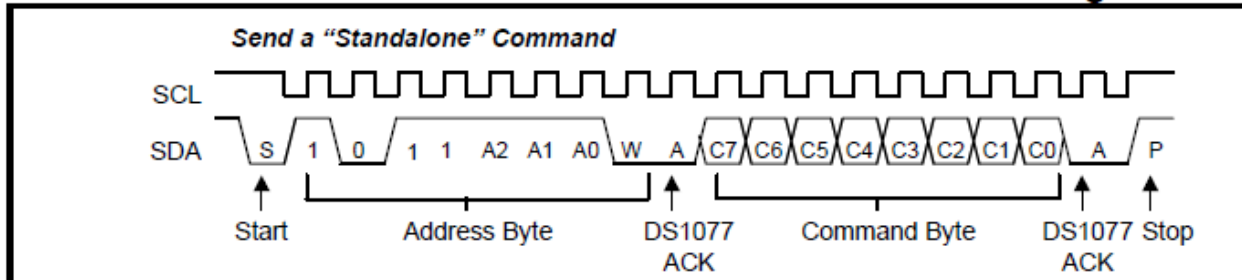
The code was taken from this website

<http://jondontdoit.blogspot.ca/2012/08/using-ds1077-programmable-oscillator.html>

The default Address of the Device

On pages 11 and 12 of the DS1077 they give you the following guidelines for communicating with the DS1077

2-WIRE SERIAL COMMUNICATION WITH DS1077 Figure 3



However sadly this address byte they use is wrong for these 2 pages. In fact the A2, A1, and A0 are zero by default, and this is mentioned once earlier in the data sheet.

So your address byte should always be **0xB0**

Setting the M pre scalar divisor

The data sheet gives you the following guide.

PRESCALER DIVISOR M SETTINGS Table 4

0M1	0M0	Prescaler P0 Divisor "M"	1M1	1M0	Prescaler P1 Divisor "M"
0	0	1**	0	0	1**
0	1	2	0	1	2
1	0	4	1	0	4
1	1	8	1	1	8

**Default Condition

This is the register you program

		MSB						LSB		MSB						LSB							
Name	*	PDN1	PDN0	SEL0	EN0	0M1	0M0	1M1	1M0	DIV1	-	-	-	-	-	-	-	-	-	-	-	-	-
Default setting	0	0	0	1	1	0	0	0	0	0	x	x	x	x	x	x	x	x	x	x	x	x	

first data byte

second data byte

*This bit must be set to zero.

However some of these values in this register are actually flipped. The placement of 0M0 and 0M1 needs to be switched, as well as the placement of 1M0 and 1M1.

You program this register by sending the following command.

```
i2c_write(ds_address, 0x02, 0x__, 0x_0)
```

Setting the N divisor value

Take the N value from the spread sheet provided by maxim and input this value into the DSregister.xlsx file. This file will give you the hex values you need to send to the DS1077 to program it properly

You program this register by the following command

```
I2c_write(ds_address, 0x01, 0x__, 0x_0)
```

General tips

I sent only one command at a time as it can be rather baffling what this chip is actually doing in response to your commands. I recommend you do the same. The chip works like a dream once you have it programmed, despite the pain of programming it, it is a great chip. It does get rather warm during operation so don't touch it. Due to the power draw fluctuations of the chip as it is an oscillator, decoupling the power supply is highly advisable to avoid supply ripples to the rest of your circuit. The data sheet recommends two ceramic capacitors, 0.01 uF and .1 uF.

Lastly before you wire it into your circuit:

Don't be confused by the whole out1 and out2, out2 is actually out0. Sparkfun apparently cannot name things, perhaps they will fix this by the time you get your break out board. Also confirm this via the oscilloscope, just to be extra sure of which pin is actually which.

Cheers

Jesse Larson, Jing Lu, and Qingqing Liu.