

Attaching a Real Time Clock to Hamstack

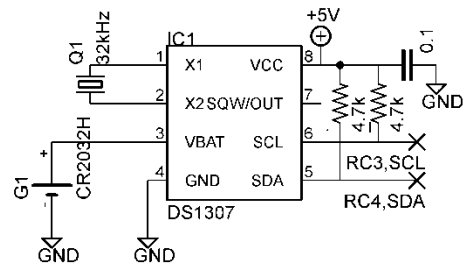
J. S. Best
May 29, 2012

This short document shows how to attach a Maxim DS1307 real time clock to the Hamstack CPU board. The DS1307 is an 8 pin device which requires only a 32kHz crystal and a lithium coin cell battery as external components to provide real time clock function with battery backup. This can be used to set a software clock function running in the Hamstack CPU on startup. This chip is controlled from the Hamstack CPU via the two pin I2C serial interface.

Hardware

The components used are as follows:

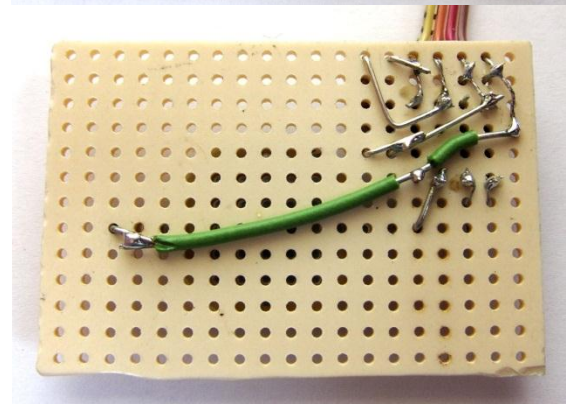
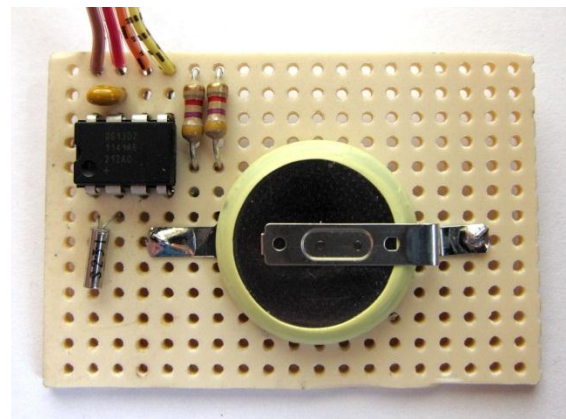
| Part | Digikey part number |
|--------------------------|--------------------------------|
| DS1307, 8 pin DIP | DS1307+-ND |
| 32.768 Hz 5ppm crystal | 300-8763-ND |
| CR2032 battery with tabs | P223-ND |
| 4.7k Ohm I2C pullups | Only 1 set required on I2C bus |
| 0.1uF dipped ceramic cap | |



The components were assembled on a small piece of perf-board and insulated with electrical tape. Connections were made with a short piece of color ribbon wire. In the photos, the connections vs. wire color are as follows:

| | |
|--------|--------|
| Red | +5V |
| Brown | GND |
| Orange | C3/CLK |
| Yellow | C4/SDI |

The real time clock wires can be hard soldered to the bottom of the CPU board or attached via connectors for experimentation (shown below).



Software

Simple library functions to set and read the DS1307 date and time are provided with the Hamstack C library, starting with version 1.11.

The declarations to use these functions are included in *hamstack.h* if the `DS1307_ENABLED` is defined in *user_options.h*. `DS1307_ENABLED` and `TIMEDATE_ENABLED` should be defined in *user_options.h* to use these functions.

```
extern unsigned char ds1307present;
```

These flags are set by the real time clock functions below. Bits are as follows: 1:present, 2:date set, 4:time set. This will be 0 if the ds1307 is not detected when calling one of the functions below.

```
extern unsigned char ds1307ctl;
```

This is the control byte for the square wave output pin on the ds1307. It defaults to 0x13, which enables the square wave output at 32768 Hz. 1 in the first hex digit enables the output. The 3 sets to 32768 Hz. Other options are 0x00: no output on the square wave output pin, 0x10: 1pps, 0x11: 4.096kHz, and 0x12: 8.192 KHz.

```
char ds1307_setcal(struct date *clock);
```

Set the time in the ds1307 to the calendar information provided in the clock parameter. See *time.h* for a definition of the fields in the date structure. Returns 0 if the clock is not present, 1 if the clock is present and successfully set.

```
char ds1307_settime(long time);
```

Set the time in the ds1307 to the values corresponding to the time parameter. As described in *time.h*, the time parameter is the number of seconds since midnight January 1, 2000. Returns 0 if the clock is not present, 1 if the clock is present and successfully set.

```
char ds1307_getcal(struct date *clock);
```

Get the time from the ds1307 and set the clock parameter accordingly. Returns 0 if the clock is not present, -1 if the clock is present, but has not been set, and 1 if the lock is present and has been set.

```
long ds1307_gettime(void);
```

Returns the time in seconds since midnight January 1, 2000. Returns 0 if the time has not been set.

The following are provided by *ds1307.h* as aliases to the functions above. Library support for other real time clock chips will define the same macros.

```
#define rtclock_setcal(d) ds1307_setcal(d)
#define rtclock_settime(t) ds1307_settime(t)
#define rtclock_getcal(d) ds1307_getcal(d)
#define rtclock_gettime() ds1307_gettime()
```

A complete example, using the DS1307 inc onjunction with the Hamstack time and date and GPS functions, is provided in *user_code_gpsclock5.c*.