

ID49- ELECTRONIC SYSTEM FOR DRIFT CLOCK CALCULATION AND SYNCHRONIZATION FOR SEAFLOOR OBSERVATORY

UMBERTO APPONI¹⁸⁷, DAVIDE EMBRIACO¹³⁰, GIUDITTA MARINARO¹²⁹, VINCENZO ROMANO¹²⁸

Abstract –The paper describes a new electronic device that allows an easily measurement of the drift between a reference time source (usually GPS) and an atomic rubidium clock which is normally used in seafloor observatories. The Rubidium clock is used in autonomous seafloor observatories to supply reference time for data acquisition with the precision of milliseconds. During the deployment of seafloor observatory the clock is synchronized with GPS. It is critical to evaluate the time drift between the clock and the GPS, when the observatory is recovered. In fact, thanks to an accurate drift measurement it's possible to have a correct timestamp for data series collected by seafloor observatory's instruments. The device described in this paper is composed by an Arduino mega shield integrated with other electronic circuits. The device is easily customizable for different clocks in fact Arduino IDE allows development of the desired features for the rubidium clock used in the specific application.

Keywords: clock drift, Arduino, GPS, seafloor observatory

I. INTRODUCTION

The device described in this paper has been realized in the framework of the EU project EMSODEV [1] (EMSO implementation and operation: DEvelopment of instrument module) whose objective is to catalyse the full implementation and operation of the EMSO (European Multidisciplinary Seafloor & water-column Observatory) distributed Research Infrastructure (RI), through the development, testing and deployment of an EMSO Generic Instrument Module (EGIM). The stand-alone prototype of EGIM will host a precise internal clock to synchronize all the data acquired by the observatory. Usually the clock is synchronized with a GPS before the deployment and immediately after the recovery of the seafloor observatory it is necessary to measure the clock drift.

The device was developed to measure the clock drift between the GPS and an atomic clock. It is designed for seafloor stand-alone observatory, that requires a precise time synchronization between GPS and observatory's clock. This goal is obtained measuring the drift between these signals, that are synchronized before the observatory deployment. After the recovery it's important to evaluate this measure to have a precise timestamp for instrument's data. The device is fully customized as the firmware is written in C code with Arduino IDE. So, it's possible to modify the code to adapt it to a different clock. Also, it's possible to modify the baud rate and serial protocol for serial GPS and clock.

II. ELECTRONICS

A device block diagrams is represented in Figure 1

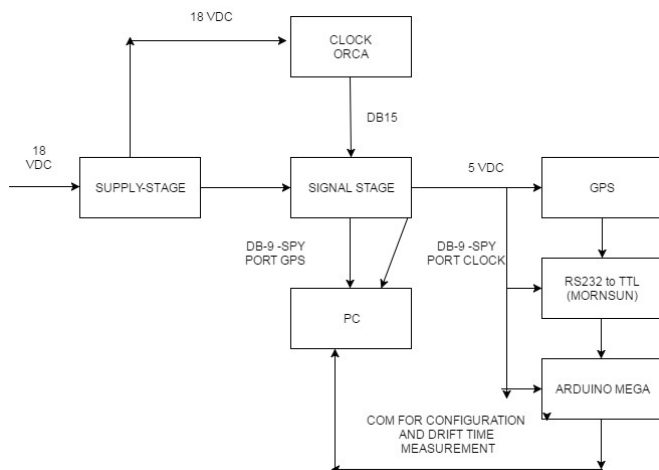


Fig. 1: Block diagram

In this figure you can identify the following blocks:

- GPS: It generates NMEA strings[2] on RS-232 serial port with DB9 connector (9600-8N1). PPS signal output on a BNC connector will be generated only if there is an identification of the position.
- ORCA Clock[3]: The rubidium clock provides the time. It has a DB15 connector that is connected to an external device called SIGNAL STAGE (Figure 1), which is responsible for distributing the TOP SEC signal (clock PPS) to a microcontroller and a BNC. This stage also allows you to bring the signals to a DB-9 (RS-232 serial port in configuration 7N1), connected to the microcontroller.
- Arduino Mega 2560 board[4]: This is the heart of the implemented device. It allows you to configure the serial (one for GPS and the other for ORCA clock), measure the time drift between the PPS[5] of the GPS signal and TOP-SEC ORCA signal, sending this to devices.
- Power Stage: This provides power to the various elements of the circuit. In particular, supply voltage of 18 VDC is sent directly to the ORCA input and, with a series of voltage regulators, it splits the power on each regulator to generate the following voltages 15VDC, 12VDC, 8VDC and 5 VDC, which powers the GPS.

III. DEVICES OPERATIONS

In order to make the measurement of clock drift and verify the calibration of the rubidium clock with GPS time the following steps are performed by device:

- Check GPS NMEA output string.
- Measure time drift between Pulse per second (PPS) GPS signal with PPS clock ORCA (TOP_SEC signal).
- Control if GPS PPS signal and TOP_SEC signal are referred to the same temporal instant.

All the operation are managed by mega microcontroller. In fact, it performs the following operations:

- Configure the BAUD RATE and serial transmission protocol. The command that it implements is:
ABCDE

Where ABCDE is a string of 5 character that the user passes to Arduino Mega serial 0 (9600 8N1).

A: ascii character that identifies the transmission rate in baud (from 300 to 115200) of the first serial (Serial 1) between 0 and 9

B: ascii character that identifies transmission protocol used the first serial.

C: ascii character that identifies the transmission rate in baud of the second serial (Serial 2).

D: ascii character that identifies the transmission protocol for the second serial

E: ascii character that identifies the selected function for the operation of the clock . If it is 1 ORCA clock function can be used, if 2 another user defined function will be used.

· Wait until the GPS is connected correctly. The microcontroller wait until GPS is locked, when NMEA string is so similar to:

```
$GPRMC,045200.000,A,3014.3820,N,09748.9514,W,36.88,65.02,030913,,,A*77\r\n
```

· Measure the clock drift between the PPS signals and TOP_SEC. To do this, the Arduino uses interrupts (Figure 2): one on the rising edge (GPS_PPS signal) and one on the falling edge (TOP_SEC signal).

· Start ORCA clock, sending correct address to it, reading the resulting time to check if this is referred to the same instant time of GPS, using this formula (Figure 3)

$$T_{pps} = (T@ - 1000000us - t_2) + (t_2 - State) < 1000000 us$$

Where the time correspond to:

Tint: enabling of the interrupts for signals with frequency of 1 Hz (GPS and TOP_SEC).

State: rising front of the GPS PPS.

State2: falling edge of the clock signal TOP_SEC of ORCA.

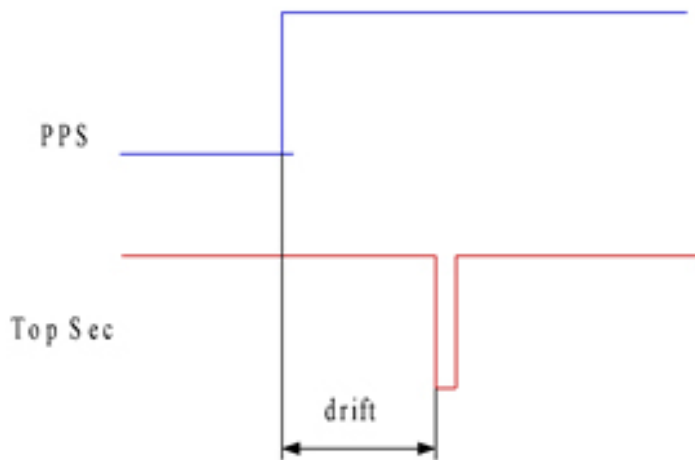


Fig 2. Drift between PPS and clock signal

$T_2 = T_{int} + 1s$
 State and State2 are always between T_{int} and T_2 .
 Time clock: time instant when the clock gives to Arduino the time.
 $T@$: Time clock + 1 second when the "@" ascii character is send to Arduino from Clock ORCA.

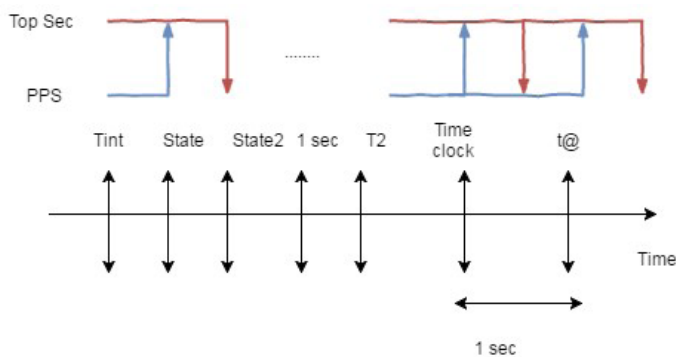


Fig 3. Time manage by Arduino mega microcontroller

IV. VALIDATION AND TESTS

The testing procedure for the device has requested these connections:

- Power for the device to 18 VDC
- Receiver antenna to GPS
- DB15 for electronic device and clock ORCA
- BNC PPS with oscilloscope CH1
- BNC oscilloscope CH2 to TOP SEC signal
- Spy -port DB9 LPHR to the PC terminal (9600,7,1)
- Spy -port DB9 GPS to the PC terminal (9600,8,1)
- USB Arduino port onto PC

Two tests are made: the first with an oscilloscope, that gives us a "qualitative" result (Figure 4); the second in order to measure the crystal oscillator accuracy mounted on microcontroller shield. With oscilloscope, we measure the temporal drift as $550000 \pm 50000 \text{ us}$. With the implemented procedure, using Arduino, the obtained drift is $546524 \pm 2 \text{ us}$ (Figure 4). The resolution time for Arduino is 1 us.

The second test shows that the device is almost insensitive to the accuracy of crystal oscillator mounted on the microcontroller, as we implemented a special function that allow the accurate measure of the period of the internal clock of Arduino with the GPS PPS signal. The reference period of PPS is 1 second and the Arduino result is 999928 us with a standard deviation of 1.24us on 150 samples.

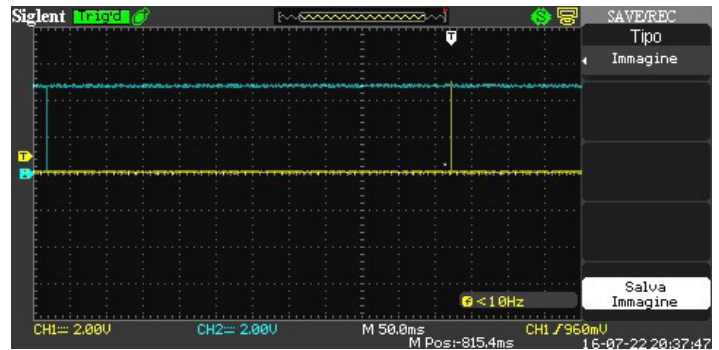


Fig 4. Drift between GPS PPS and clock signal TOP_SEC



Fig 5. The Electronic system

V. CONCLUSIONS

The device presented in this paper allows the time drift measurement between the PPS GPS signal and the 1 Hz output of seafloor observatory rubidium clock. The device has a time resolution of 1 us. Such result is innovative because the obtained resolution is four time improved respect to the one obtained by Arduino Mega standard shield. Moreover the firmware code can be implemented using Arduino IDE making the electronics easy customizable. It allows to write in C language a management function for every rubidium clock used in the seafloor observatory. The device also evaluates the time drift between GPS signal and Arduino internal clock, in order to be insensitive to the accuracy of the crystal oscillator mounted with microcontroller.

REFERENCES

- [1] <http://www.emsodev.eu/>.
- [2] VV.AA, NMEA reference manual, SIRF Technology Inc., San Jose (CA), 2005.
- [3] VV.AA, LPHS/R CLOCK User's manual, ORCA instrumentation., Brest, 2002.
- [4] Mark Schmidt, Arduino's manual, Apogeo s.r.l., Italy, 2010.
- [5] VV.AA, GLOBAL POSITIONING SYSTEM PRECISE POSITIONING SERVICE PERFORMANCE STANDARDLOCK, U.S department of defence, Washington D.C., 2007.