

# Software Synchronization of DOR Clocks to UTC

Version 1.0

December 11, 2003

**John Jacobsen**

John Jacobsen IT Services  
for LBNL and the IceCube Collaboration

**John Jacobsen**  
Information Technology Services



1420 W. Edgewater Ave., Suite 3, Chicago, IL 60660

Phone: +1 (773) 769 0522 FAX: +1 (775) 254 5992

E-mail: [john@johnj.com](mailto:john@johnj.com)

Web Site: [it.johnj.com](http://it.johnj.com)

Available from <ftp://it.johnj.com/pub/icecube/domhub/driver/doc> or from the IceCube Docushare repository.

## Contents

Introduction .....	1
Instructions .....	1
Prerequisites .....	1
Procedure .....	2
Details and Potential Issues .....	3
References .....	4

## Introduction

This document describes a procedure to perform the absolute time synchronization of the DOM Readout (DOR) cards for the IceCube experiment at the South Pole. The goal is to get the local clock counters within the DOR cards to match universal coordinated time (UTC), in units of DOR clock ticks since January 0\* of the current year. For example, if the DOR clock speed is `CLOCK_SPEED` cycles per second, then a DOR timer value of `3600 * CLOCK_SPEED` represents exactly one hour into the new year.

The system relies on a TrueTime GPS clock as a source of both a 1 pps time signal and an ASCII representation of the current second. It also relies on clock distribution hardware for sending the clock's signals to all the DOR cards.

The software implementation includes functionality within the Linux device driver[1] for the DOR cards, as well as a standalone program (`dortimer`) for receiving the GPS time string and for carrying out the appropriate DOR driver operations. The driver functionality used in this procedure obviously depends heavily on the firmware design by Karl-Heinz Sulanke [2]. It is proposed by Sulanke to eventually carry out all the time synchronization in hardware.

## Instructions

### Prerequisites

- A PC consisting of the “approved” DOM Hub Configuration;
- A TrueTime GPS clock whose outputs are cabled to Sulanke’s clock distribution system, which in turn is cabled to the DOR cards. The RS-232 time string output must be cabled to the first serial port on the DOM Hub, `/dev/ttyS0`. The TrueTime RS-232 settings should be 9600,8,N,1 (per Serap Tilav. Note that these are NOT the default settings!);
- At least one DOR card installed in the DOM Hub;
- The latest dor-driver project installed on the DOM Hub;

```
% cvs checkout -d dor-driver
```

---

\* Jan. 0 refers to the exact moment when Dec. 31 becomes Jan. 1 of the new year.

```
% cd dor-driver/driver
% make
% su
# make install
# make chkconfig
# insmod dh.o          (if driver is not already running)
# make installfw
```

## Procedure

- 1) **Reboot.** Log in and make sure the DOR driver loaded automatically:

```
% /usr/local/bin/fvers.pl
Driver $Revision: 1.133 $
/proc/driver/domhub/card0 -- 005h
```

If instead you see:

```
% /usr/local/bin/fvers.pl
No driver present!
```

stop here and call technical support (John J.).

You don't have to reboot every time you synchronize the DOR cards, but you should reboot after installing the DOR driver for the first time.

- 2) **Install the most recent working IceTop firmware**, in the "009" series. The current version as of this writing is DC\_009b.rbf; ask Kalle Sulanke for the correct version (or refer to ICETOP\_FIRMWARE in the Makefile in the dor-driver project). Put the firmware on flash page 2 using the "fpga" script (installed with the dor-driver project):

```
% su
# /usr/local/bin/fpga.pl 0 p 2 /usr/local/dor/fpga/DC_009b.rbf
# /usr/local/bin/fpga.pl 0 f 2
# exit
% /usr/local/bin/fvers.pl
Driver $Revision: 1.133 $
/proc/driver/domhub/card0 -- 009b
```

The last step verifies that firmware version 009b is actually running. Note that in this example the DOR card ID was 0; substitute as appropriate for other card IDs.

- 3) **Run the dortimer program** to synchronize the DOR clock to UTC:

```
% /usr/local/bin/dortimer /dev/ttyS0 0 init
```

```
DOR TIMER INIT ... by J. Jacobsen, http://it.johnj.com
Waiting for GPS time string data on serial port...
Got 16 byte time string from /dev/ttyS0: 345:10:17:31
```

```
Loading DOR firmware with day=344, hr=10, min=17, sec=31.
Will load the following clock value into DOR card 0:
0x21d4e85a10c00
Done.
```

The second argument, “0”, is the DOR card ID.

The astute reader will notice that the day value in the GPS time string differs from value loaded into the firmware by 1. This is because the TrueTime GPS clock represents Jan. 1<sup>st</sup> as day 1 rather than day 0.

4) **Make sure it worked:**

```
% /usr/local/bin/dortimer /dev/ttyS0 0 check
```

```
DOR TIMER CHECK ... by J. Jacobsen, http://it.johnj.com
Waiting for GPS time string data on serial port...
Got 16 byte time string from /dev/ttyS0: 345:10:19:45
```

```
Congratulations, the DOR timer value matches the GPS clock time string.
Done.
```

5) Repeat steps 3 and 4 for all DOR cards installed in the Hub, changing the DOR card ID argument to `dortimer` as appropriate.

After the above time synchronization procedure has been carried out, the 64-bit DOR time stamps read out during DOR-DOM time calibrations (RAPCals) should reflect the current UTC in units of DOR clock cycles.

## Details and Potential Issues

The `dortimer` program uses `/proc` files in the driver to load the UTC value into the DOR cards and to retrieve the UTC value in “check” mode). The relevant details of the implementation in the driver are described in the most recent version of the driver documentation[1]. Great care is taken in the driver and firmware to avoid race conditions during the synchronization procedure; see Refs.[1,2].

The software was developed on a configuration at LBNL without an actual GPS clock available for testing. Instead, a simulation of the GPS time string was provided using a program called `simgps` (`simgps.c` in the `dor-driver/driver` directory). `Simgps` creates a time string in the appropriate format based on the PC’s system time. When two serial ports on a PC are cabled together, the time string from `simgps` is sent to `/dev/ttyS1` and can then be read from `/dev/ttyS0` by `dortimer`.

If for some reason `dortimer` has trouble reading the time string (this was occasionally seen to happen during testing at LBNL), the program can simply be run again.

The software was written under the assumption that the serial RS-232 time string arrives at the DOR card just after a one-pulse-per-second (1pps) time signal. This will be the case in the setup at South Pole during the '03/'04 season. Due to the absence of a suitable clock in our test setup, the 1pps signal and the RS-232 time string were not precisely in phase. Running the “check” mode of `dortimer` sometimes showed an exact 1 second (20,000,000 clock cycles) difference between UTC and the DOR clock, presumably because of this limitation of our test setup. In other words, our setup did not allow us to verify that the clock synchronization was always exact, though it did seem to match exactly at the sub-second level. The caveat here is that one should take extra care to watch for such a 1 second offset in the data taken with the final setup at South Pole.

## References

1. J. Jacobsen, *The DOM Hub Device Driver – Function and Design*, 2003. Available on the IceCube DocuShare repository or at <ftp://it.johnj.com/pub/icecube/domhub/driver/doc>.
2. K. Sulanke, *DOR-API Description*, 2003. Description of the FPGA registers used to control the DOR hardware. Contact [sulanke@ifh.de](mailto:sulanke@ifh.de) for details.