Tuning the T4 Science hydrogen maser

P. de Vicente, S. García-Espada, A. Barcia,
R. Bolaño, L. Barbas

Informe Técnico IT-OAN 2010-1

Revision history

Version	Date	Author	Updates
1.0	10-03-2010	P. de Vicente	First version

Contents

1 Introduction

A new hydrogen maser was rented from T4-Science and installed in Yebes on August 28th, 2009. The model is iMaser EFOS C, Serial Number: 37. This clock will stay in Yebes until a definitive maser from that same company is built and installed.

2 An overview on the installation

The hydrogen maser was installed in the maser room located in the basement of the 40 m tower building.

The maser can be monitored and remotely tuned through a serial port. T4 Science provided a DOS program which runs in Windows and allows to monitor a number of parameters and change the output frequency by tuning an internal synthesizer. That application was installed on a laptop from the OAN (Observatorio Astronómico Nacional). The laptop was placed on top of the maser connected to the maser serial port via a serial cable. The laptop is only powered up when needed.

Image 1 shows a photograph of the clock in the room with the laptop on top.

Figure 1: *EFOS Maser in its room*

This configuration was changed by a more flexible one. Currently the serial port is connected to a Lantronix Serial to Ethernet converter. A Python class which connects via sockets to the Lantronix device was developed by R. Bolaño and can be used from computer `meteomaser`. P de Vicente wrote a simple script to be used by users, which uses the previous class. It asks for the relative frequency error, shows the current maser status, computes the new synthesizer value and corrects it taking into account the supplied relative frequency error.

This maser provides several signals of which only 3 are being used now: 2 signals at 5 MHz and 1 signal at 100 MHz. The three are sent via RG-58 cables to the backends room. One 5 MHz signal and the 100 MHz signal are sent to a Quartzlock A5 distribution amplifier. The second 5 MHz signal is sent to the VLBI data acquisition terminal.

In order to measure the maser frequency, the same setup as described in report IT–OAN 2005-5 was used. This setup has been used since the Kvarz CH1-75 maser was installed in the 40 m building (see report IT–OAN 2007-3)

3 Tuning and drift since installation

Figure 3 shows the maser drift obtained by comparing 1 PPS from the maser and 1 PPS from the GPS. The 1 PPS from the maser is generated at the VLBI data acquisition Time Box. Both pulses are compared by an HP 53131A counter with 1 ps accuracy.

3.1 A review on PPS (Pulse Per Second) comparison

The 1 PPS from the GPS is received in connector 1 at the HP 51313A counter, the 1 PPS from the maser on connector 2. The counter measures the time ellapsed between the pulse on connector 1 and the pulse on connector 2. This value is displayed on the counter screen.

The VLBI data acquisition time box generates 1 PPS by counting $5\,10^6$ cycles from the 5 MHz signal. The count starts when the amplitude signal is zero. There are two possibilities for this event within a cycle: when the ramp is positive or when the ramp is negative. The alternative used depends upon the connection of an external synchronizing signal. The timing box has got two connectors for an external 1 PPS sync signal, which are labelled: `Input +` and `Input -`. If the sync signal is connected to `Input +` the count will start when the ramp is positive and the signal crosses zero. On the contrary if the sync signal is connected to `Input -` the count will start when the ramp is negative and the signal crosses zero. Currenlty only `Input` + is connected. The sync signal is 1 PPS from the GPS receiver.

Synchronization is achieved by pressing the black button in the Timing Box for more than a second. Once the button has been released the timing box will start counting immediately after it has received 1 PPS from the GPS **and** the 5 MHz amplitude signal is zero when the ramp is up. Hence the generated 1 PPS may differ from the sync signal at most one cycle period: 1/5 MHz $= 0.2\,\mu s$. However the 1 PPS signal from the maser still delays in the Time Box and in cables to the counter and this difference may increase up to 350 ns in the counter. That is the usual value seen on the HP51313A screen immediately after a synchronization.

If the GPS ticks before the maser the counter will display a small positive value. If the maser ticks faster than the GPS, and hence its frequency is higher than the GPS synthesized frequency, there will be a moment in which the maser will tick before the GPS. When this happens the counter will display a number close to 0.99999. The counter is read via GPIB by a computer which stores this information on a database. The recorded value is however the smallest difference between Time(GPS) - Time(Maser). Hence when the GPS ticks before the maser, Time(GPS) < Time(maser) and the value will be negative and close to zero. If the maser ticks before the GPS, Time(GPS) > Time(Maser), and the value will be positive and close to zero since only the smallest distance between pulses is recorded.

The difference between the GPS and maser pulses is recorded on a database as a function of time. When these values are represented on a graph, a straight line is seen for time intervals smaller than a month. The slope of the line will be the relative frequency error between the maser and the GPS synthesized frequency. If the time interval is larger, a parabolic behaviour will be seen. This means that the maser frequency is not stable and drifts slowly with time, probably due to cavity deformations. See figure 2 for an example. Here, the behaviour of the drift for the iMaser since November 2009 to March 2010 is depicted.

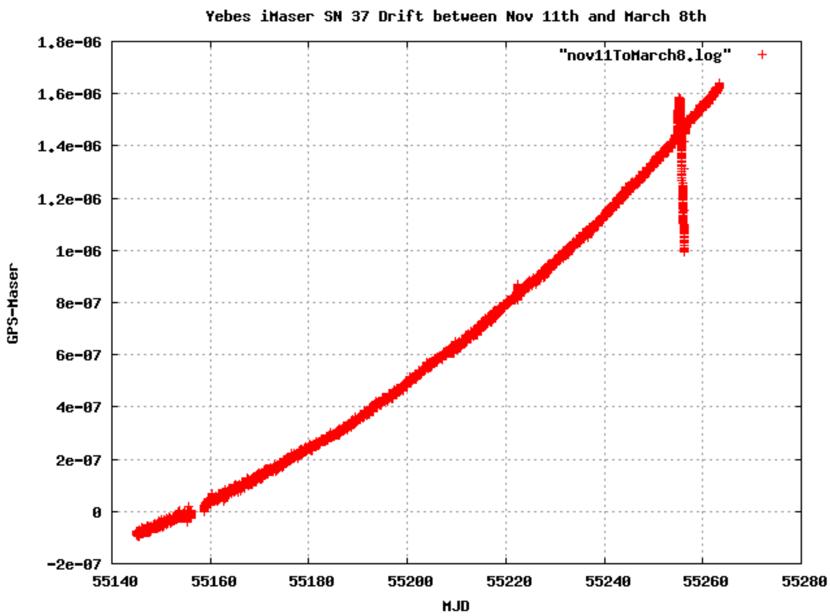

Figure 2: *Maser drift between November* 11^{th} *2009 and March* 8^{th} *2010. The time units are Modified Julian Days. The vertical line is due to an electrical power disruption which left the GPS antenna unavailable for 1 day and a half. The PPS comparison was trustless during that period*

3.2 Tuning history

5 tunings were performed since installation up to March 24^{th} 2010. Table 1 summarizes the date and time each tuning was performed, the measured drifts, the applied corrections and the application used for the correction. The measured drift is obtained from the slope of the graph.

The maser frequency was corrected using a Windows application running in the laptop up to March 8^{th} 2010. Since then this is achieved via a Linux Program as described below.

Date	Time	Meas. $\Delta f/f$	Appl. $\Delta f/f$	Soft. used
03/09/2009	09:20	$-4.50\,10^{-12}$	$-4.5\,10^{-12}$	Windows
07/09/2009	07:15	$-8.86\,10^{-12}$	$8.9\,10^{-12}$	Windows
09/09/2009	07:30	$-8.86\,10^{-12}$	$9\,10^{-12}$	Windows
14/09/2009	10:15	$8.52\,10^{-13}$	$-8\,10^{-13}$	Windows
08/03/2010	17:40	$2.80\,10^{-13}$	$8.4\,10^{-13}$	Linux

Table 1: *Date and time of the tuning. The measured drift is the drift up to the tuning date. The correction was applied in that date and time. There was an error in last correction and a different value from the measured relative error was applied.*

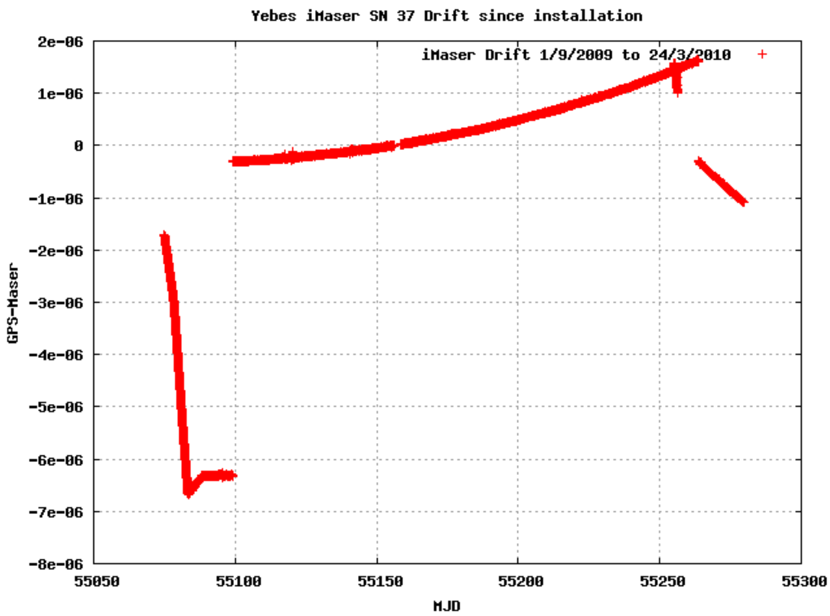

Figure 3: *Maser drift since installation to March 2010. The time units are Modified Julian Days. The discontinuities in the line indicate the tuning moments. On day 55100 (MJD) there was no tuning. The observed jump is only due to a synchronization, but not to a change of drift.*

3.3 Windows Tuning Application

The windows application, when opened, displays a window frame called ”Monitoring Parameters” which always remains opened and updates the value of some parameters. It is possible to correct the maser frequency by opening a new window called ”Frequency Synthesizer”. The Windows application shows a strange behaviour before and after correcting the maser frequency. The monitored maser frequency was read in different moments, before and after the tuning and from the main window and the ”Frequency Synthesizer” window. These tests are summarized in table 2. The conclusion is that the final cavity maser frequency after the correction is displayed in the ”Actual Freq. Synth.” window correctly once this window is closed and reopened again. The correct value is also displayed in the main window, after the ”Freq. Synth” window is closed for the second time, but it has less resolution than in the former window. These values have been highlighted in bold face in table 2.

The sign of the frequency correction was discovered by trial and error. The correct sign to be used is the opposite of the measured drift.

Date	Moment	DDS readout	”Actual Freq. Synth.” readout	Applied Correction
7/9/2009	Before OK	1420420877.00495	1420405750.287257	$8.9, 10^{-12}$
	After OK	1420420877.00495	1420420877.004959	
	After closing window ”Freq Synth” and reopening it	1420420877.00495	**1420405750.287257**	
	After closing window ”Freq Synth”	**1420405750.28825**	–	
9/9/2009	Initial monitor	1420420877.00495	–	
	Before OK	1420420877.00495	1420405750.287257	$9\,10^{-12}$
	After OK	1420420877.00495	1420420877.004959	
	After closing window ”Freq Synth” and reopening it	1420420877.00495	**1420405750.301044**	
	After closing window ”Freq Synth”	**1420405750.30104**	–	
14/9/2009	Initial monitor	1420420877.00495	–	$-8\,10^{-13}$
	Before OK	1420420877.00495	1420405750.301044	
	After OK	1420420877.00495	1420420877.004959	
	After closing window ”Freq Synth” and reopening it	1420420877.00495	**1420405750.299908**	
	After closing window ”Freq Synth”	**1420405750.29990**	–	

Table 2: *Windows appplication behaviour*

The windows application was used for the last time on September 14th 2009. After that date the Linux code was used.

4 Frequency Correction

The 5 MHz master oscillator (OCXO) is controlled by the hydrogen cabin signal ($\simeq 1420$ MHz) using a loop in which the latter signal is transformed to 405 KHz and compared to 405 KHz derived from the OCXO after multiplications (see iMaser manual).

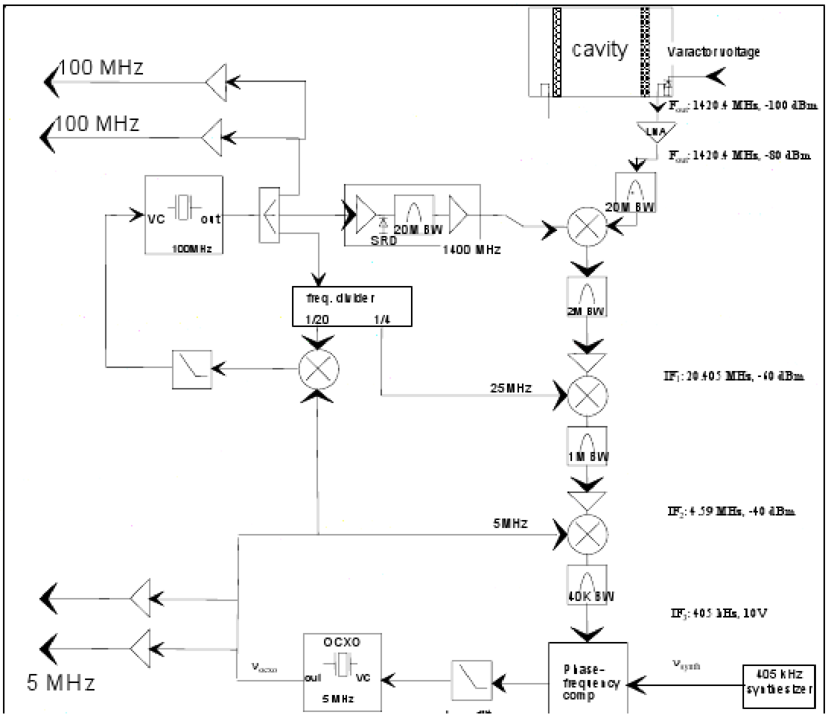

Figure 4: *PLL iMaser scheme*

The OCXO frequency adjustment is done tuning the frequency of the 405 KHz synthesizer. This procedure differs from that used with the KH-75 maser from Kvarz, in which the cavity was tuned using an electronic varactor.

Let f_H be the hydrogen frequency at the cavity output, f_o the OCXO frequency and f_s the 405 KHz synthesizer frequency. Then if the system is locked:

$$f_s = f_o - [5\, f_o\ - (f_H - 280\, f_o)] \tag{1}$$

where 280 and 5 are the multiplying factors to get the mixing frequencies at 1400 MHz and 25 MHz from f_0. Symplyfing the above equation we get:

$$f_s = f_H - 284\, f_o \tag{2}$$

f_s is obtained from f_o applying linear operations. So let us write:

$$f_s = k_s\, f_o \tag{3}$$

where k_s is a factor which can be changed to modify the 405 KHz signal, then:

$$f_H = f_s + \frac{284}{k_s}\, f_s = (1+k)\, f_s \tag{4}$$

Therefore the frequency at the cavity output can be considered proportional to the frequency from the frequency synthesizer, and also to the OCXO frequency.

From a practical point of view and according to the iMaser manual the hydrogen frequency at the cavity output is (f_H) is

$$f_H = f_{H0} + (FM - FM0)\, 9.09496\, 10^{-6} \tag{5}$$

where FM and $FM0$ are the frequency synthesizer values for the desired frequency and the reference maser frequency, f_{H0} (1420405751.0000 Hz) respectively. FM value can be modified by modifying four bytes of the register at a given address. $9.09496\, 10^{-6}$ corresponds to the frequency synthesizer resolution:

$$\frac{5\, 10^6}{2^{39}} = 9.09496\, 10^{-6} \tag{6}$$

Equation 5 relates the frequency synthesizer resolution with the frequency at the cavity output. Let us suppose we wish to modify the frequency to correct for a relative error frequency. The new frequency should be f'_H:

$$f'_H = f_H + \frac{\Delta f}{f}\, f_H \tag{7}$$

and hence:

$$f_{H0} + (FM - FM0)\, 9.09496\, 10^{-6} + \frac{\Delta f}{f}\, f_H = f_{H0} + (FM' - FM0)\, 9.09496\, 10^{-6} \tag{8}$$

and reordering terms:

$$FM - FM0 + \frac{\Delta f}{f}\, \frac{f_H}{9.09496\, 10^{-6}} = FM' - FM0 \tag{9}$$

and the new register should be:

$$FM' \;=\; FM + \frac{\Delta f}{f}\, \frac{f_H}{9.09496\, 10^{-6}} \tag{10}$$

$$\tag{11}$$

where f_H is the frequency at the cavity output before modifying the frequency synthesizer. Hence FM' can be computed from the current FM and f_H values provided by reading the current status maser.

5 Code in Linux and Database

R. Bolaño has written a Python class which connects via sockets to the Lantronix device and requests the maser its status. This class allows to write on some memory registers and modify the synthesizer frequency, and hence the output maser frequency (5 MHz). The serial communication between the Lantronix and the maser is achieved using a baud rate of 9600 bit/s, No parity, 8 data bits, 1 start bit and 1 stop bit.

Table 3 summarizes the monitored parameters. The list has been taken from the iMaser manual.

A total of eight commands may be used to obtain the maser status and modify its frequency. Here we include the 3 most used ones.

- `M (CRLF)` Used for reading the 40 analog channels and lock status. The string read has a length of 113 bytes and is composed of 3 bytes per channel (channels 1 to 32), 2 bytes per channel (channels 33 to 40) and 1 bit for the lock status (1 or 0) followed by a CRLF
- `RXX(CRLF)` Used for reading RAM at address XX. XX is 1 byte followed by CRLF. This command is used to read the synthesizer values on RAM addresses: 0E, 0F, 10 and 11.
- `WXXYY(CRLF)` Used for writing data YY at address XX in the RAM. This command is used to modify the synthesizer values and hence the hydrogen output frequency. RAM addresses written are 0E, 0F, 10 and 11.

The monitor parameters are read and transformed from hexadecimal to integer and multiplied by value in column 4 of table 3 to obtain the physical unit from the hexadecimal byte readout.

The following ”high level” functions have been implemented in the maser python class to monitor the status and modify some of its parameters:

- `readRAM(addr)` Function which read RAM registers and returns a two byte hexadecimal value
- `writeRAM(addr, value)`. Function which writes a two byte hexadecimal value on the RAM register.
- `swVersion()`. Returns a string with the software version.
- `activateSynth()`. Activates the synthesizer with new values.
- `readAnalogChannelsAndLockStatus()`. Returns a python list with the monitor values in meaningful physical units.
- `readSynthFreqAndACTRegisters()`. Returns a list with the frequency synthesizer and ACT registers. Used mainly to get the maser frequency and the current synthesizer frequency as an hexadecimal value.
- `computeFrequencyCorrection(df)`. Computes the new synthesizer frequency from the current status and a supplied relative error frequency.

Channel	Description	Name	[physical unit/LSB]
1	Battery voltage A	U batt.A [V]	2.441E-02
2	Battery current A	I batt. A [A]	1.221E-03
3	Battery voltage B	U batt.B [V]	2.441E-02
4	Battery current B	I batt. B [A]	1.221E-03
5	Hydrogen pressure setting	Set. H [V]	3.662E-03
6	Hydrogen pressure measurement	Meas. H [V]	1.221E-03
7	Purifier current	I purifier [A]	1.221E-03
8	Dissociator current	I dissociator [A]	1.221E-03
9	Dissociator light	H light [V]	1.221E-03
10	Internal top heater	IT heater [V]	4.883E-03
11	Internal bottom heater	IB heater [V]	4.883E-03
12	Internal side heater	IS heater [V]	4.883E-03
13	Thermal control unit heater	UTC heater [V]	4.883E-03
14	External side heater	ES heater [V]	4.883E-03
15	External bottom heater	EB heater [V]	4.883E-03
16	Isolator heater	I heater [V]	4.883E-03
17	Tube heater	T heater [V]	4.883E-03
18	Boxes temperature	Boxes temp. [C]	2.441E-02
19	Boxes current	I Boxes [A]	1.221E-03
20	Ambient temperature	Amb. Temp. [C]	1.221E-02
21	C-field voltage	C field [V]	2.441E-03
22	Varactor voltage	U varactor [V]	2.441E-03
23	external high voltage value	U HT ext. [Kv]	1.221E-03
24	external high voltage current	I HT ext. [uA]	1.221E-01
25	internal high voltage value	U HT int. [kV]	1.221E-03
26	internal high voltage current	I HT int. [uA]	1.221E-01
27	Hydrogen storage pressure	Sto. press. [bar]	4.883E-03
28	Hydrogen storage heater	Sto. heater [V]	6.104E-03
29	Pirani heater	Pir. heater [V]	6.104E-03
30	Unused	Unused []	0.000E+00
31	405 kHz Amplitude	U 405 kHz [V]	3.662E-03
32	OCXO varicap voltage	U ocxo [V]	2.441E-03
33	+24 V supply voltage	+24Vdc [V]	9.766E-02
34	+15 V supply voltage	+15Vdc [V]	7.813E-02
35	-15 V supply voltage	-15Vdc [V]	-7.813E-02
36	+5 V supply voltage	+5Vdc [V]	3.906E-02
37	-5 V supply voltage	-5Vdc [V]	-3.906E-02
38	+8 V supply voltage	+8Vdc [V]	3.906E-02
39	+18 V supply voltage	+18Vdc [V]	7.813E-02
40	Unused	Unused []	0.000E+00

Table 3: *List of iMaser monitor parameters. Taken from the imaser manual*

- `applyFrequencyCorrection(df).` Modifies the frequency synthesiser value applying a correction to compensate the relative error frequency.

P. de Vicente wrote a small script which calls the previous Python class and interacts with the user. It basically uses functions:

- `readAnalogChannelsAndLockStatus()`
- `readSynthFreqAndACTRegisters()`
- `computeFrequencyCorrection(df)` and
- `applyFrequencyCorrection(df).`

The user runs the script followed by a parameter which contains the relative frequency. This script allows to modify the maser frequency. The host, account and directory where this script can be run is known by the authors and is not included here for security reasons.

Below we include an usage example:

```
./correctMaser.py 8.4e-13
Connecting to the maser ...
Do you want to see all current parameters y/[n]) ? n
Current Frequency Synthesiser freq.: 63226438  (hex)
Current Maser frequency: 1420405750.2999072  Hz
New Frequency Synthesiser freq. to apply: 632264BB  (hex)
Do you want to apply this correction 8.4e-13 y/[n]) ? y
Applying the correction ...
  Frequency Synthesiser freq. after the correction: 632264BB  (hex)
  Maser frequency after the correction: 1420405750.298716  Hz
```

The applied correction should have the same sign as the measured drift. When the measured drift is positive, the maser frequency is higher than the GPS synthesized frequency. This should be checked after the correction has been applied. In the previous example a positive drift was measured and applied. The maser frequency before the tuning was 1420405750.2999072 Hz, and after the tuning 1420405750.298716 Hz. This means that, if the last value were closer to the optimal one, the maser frequency before the change was slightly higher than the correct one.

L. Barbas has created a database to store information on the maser status, a python script to fill it by using the previous Python class and a web page to plot any parameter stored in the database. The maser status is requested once per hour and stored in the database. The database contains the whole list of parameters in table 3 plus the maser frequency. The web page is located in `http://hera.oan.es/graficas_interactivas/` and can be reached only from inside the private OAN LAN.

Figure 5 shows a snapshot of the web page that allows to plot any of the previous maser parameters as a function of time.

The web form is programmed in PHP and starts a Python script that uses Matplotlib and MySql, and is integrated in the Apache server via the mod-python module. The web form allows to select the parameter from a combo box, the date/time interval from a calendar widget, the plot title, axis labels, and the X axis resolution. The plot will be shown on a separate tab in

Figure 5: *Web page at http://hera.oan.es/graficas_interactivas/ which allows to plot maser parameters as a function of time*

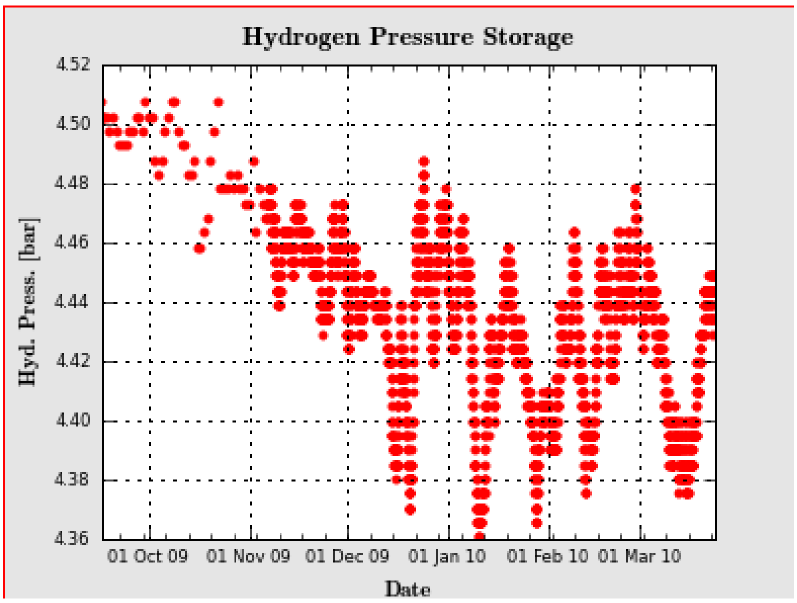

Figure 6: *Hydrogen storage pressure [bar] as a function of time. This graph is a snapshot obtained from the web form available at http://hera.oan.es*

the browser. If another parameter should be plotted, the web form needs to be reloaded and the new parameter and options selected. Data can also be extracted and stored in an ASCII local file with two columns: date and value.

Figure 6 shows the Hydrogen storage pressure as a function of time, as shown by the web page interface

References

[1] P. de Vicente, A. Garrigues. ”Monitorización del máser de hidrógeno del CAY”, IT–OAN 2005-5

[2] P. de Vicente, J.D. Gallego, R. Bolano, C. Almendros. ”Monitorización del máser de hidrógeno antes y después del traslado al radiotelescopio de 40m”, IT–OAN 2007-3

[3] T4 Science SA. ”Installation, Operation and Maintenance User Manual”, 2008