

How do I configure NTP to use the PPS source?

Model:

Arc Series C Board (10G-PCIE2-8C2-2S-SYNC)

Software:

DBL and Sniffer.

Operating System:

Supports the Linux operating system.

Information:

When using Timesource (hardware) timestamping, if you would like to sync the host's clock to the Timesource, the software exports a PPS output to the system. This PPS output can be used as an input into ntp. The following instructions describe how to configure NTP to use the PPS source.

Below are the instructions for compiling NTP from source with PPS support:

1. Obtain a copy of the ntp source from the NTP website (www.ntp.org) must be at least version 4.2.6.
2. Obtain a copy of the timepps.h (it comes from the pps-tools package at [git://github.com/ago/pps-tools.git](https://github.com/ago/pps-tools.git)) and copy it to /usr/include/sys. The timepps.h file is also attached: [timepps.h](#)
3. Untar the ntp source. You may want to read over the documentation that comes with ntp about how to configure and make it. We will provide a very simple example in the following steps.
4. Configure the ntp source. If you don't want ntp to overwrite your system's ntp, please set the "--prefix=<path>" variable appropriately during the configure step for where you want to install ntp. An example of our configure step is:

```
$ configure --prefix=/opt/ntp
```

During the configure step you should see the following message: **"checking for sys/timepps.h...yes"** (It scrolls by quickly, if you miss it you can check in the config.log after configure is done).

5. Run 'make', then, optionally, 'make install' if you set the prefix variable during the configure step.
6. Edit the config file for ntp. For the sake of simplicity, you can copy the config file from your system's ntp and then edit that.

```
$ cp /etc/ntp.conf /opt/ntp  
$ vi /opt/ntp/ntp.conf
```

7. Add something similar to the following to ntp.conf (inserting your server's IP address):

```
# The minsane keeps ntp going even if the server the pps  
# is associated with drifts away from the pps resulting in ntp  
# rejecting it. Alternatively you can use the 'mindist .015' or  
# greater values to increase how much the server can drift from the pps.  
tos minsane 0  
  
# This server will be used to enumerate the seconds for the pps interface.  
server <your server ip> iburst prefer minpoll 4 maxpoll 4  
  
# The 127.127.22.X if the pps device and corresponds to /dev/ppsX.
```

```
server 127.127.22.0 minpoll 4 maxpoll 4
fudge 127.127.22.0 flag2 0
```

8. Make sure to kill your system's ntpd if it is already running and start the ntpd you built.

```
$ killall ntpd #or whatever is appropriate for your system
$ /opt/ntp/bin/ntpd -c /opt/ntp/ntp.conf
```

9. After about 30 seconds or so, ntp should be using the pps output from the adapter. You can check by running:

```
$ ntpq -p
      remote           refid      st t when poll reach  delay  offset  jitter
=====
*sw2.sw.myricom.com 70.86.250.6      3 u   -  16    3   0.322   1.067   0.109
oPPS(0)             .PPS.            0 l   1   16    1   0.000   0.904   0.000
```

10. NTP is quite conservative about changing system time, and it can take quite a while for it to slowly adjust the system time to get close to the time being provided by the pps output. This can take an hour or more. Depending on your ntp.conf you can have ntp output various statistics to output files. You may want to look over the ntp documentation and become acquainted with the loopstats file as a way of checking how ntp thinks it is performing. For example, to output the statistics to /var/log/ntpstats, you may want to make sure the following lines are in your ntp.conf:

```
statsdir /var/log/ntpstats/
statistics loopstats peerstats clockstats
```

Revision:	Date:	Change:
1	6/23/2016	Initial Draft
2	7/20/2016	Feedback