

Driver debugging

In some cases, when investigating an issues with the touch device, it is important to understand what the driver is doing at the time the issue occurs.

To this end, UPDD V6 has a number of debugging logging facilities built into the driver.

Session logging

To create a debug log for a given session you can enable the file debugging facility via the internal driver setting filedebug as follows:

Command : [upddutils filedebug n](#) where n equals:

Value	Meaning
0	Disable logging
1	To follow
2	To follow
3	Log run-time critical events

When enabled, debug data is logged in the [updd log file](#)**updd.log** as created anew or appended to an existing log file if it exists so you may wish to delete a previous log file.

Unless otherwise informed we normally require a log from the command 'upddutils filedebug 3', wait 5 seconds and then touch the screen.

In some operating systems you may need to disable logging to be able to access the content of the log. Logging is disabled once the session closes, e.g if the driver is stopped or a reboot occurs.

Example

This log shows the point at which a USB read (issued by the driver) fails and the driver resets the USB bus and retries the read.

```
2017/01/12-08:38:23: TRC: VID:0x457 PID:0x1056 is supported by this UPDD package
```

```
2017/01/12-08:38:23: TRC: USB location bus:001a-port:0002
```

```
2017/01/12-08:38:23: TRC: Location already connected
```

Driver debugging

```
2017/01/12-08:38:23: TRC: Found VID:0x5ac PID:0x8215
2017/01/12-08:38:23: TRC: Found VID:0x819 PID:0x101
2017/01/12-08:38:23: TRC: Found VID:0x88e PID:0x5036
2017/01/12-08:38:23: TRC: Enumeration ends
2017/01/12-08:38:24: DBG: updd-tuio Registered callbacks 0x0
2017/01/12-08:38:24: DBG: updd-tuio Registered callbacks 0x0
2017/01/12-08:38:24: DBG: updd-gestures Registered callbacks 0x0
2017/01/12-08:38:24: DBG: updd-gestures Registered callbacks 0x0
2017/01/12-08:38:25: DBG: updd-tuio Registered callbacks 0x0
2017/01/12-08:38:25: DBG: updd-tuio Registered callbacks 0x0
2017/01/12-08:38:26: ERR: USB error: (read) Input/Output Error
2017/01/12-08:38:26: DBG: Issuing clear halt and retrying
```

In this case the driver will retry the reset 5 times before notifying that the device is 'disconnected'.

Alternative log file path

In some OS, such as Android, if the default log file location is unavailable set the environment variable UPDDLLOGPATH to the target folder before running enabling logging, e.g.

```
export UPDDLLOGPATH=/data/alternatelogpath/
./upddenv ./upddutils filedebug 3
```

NB do not omit the trailing slash.

Startup logging

For logging to be permanently enabled over all driver sessions, to enable API logging and to log additional operational information you need to enable permanent logging.

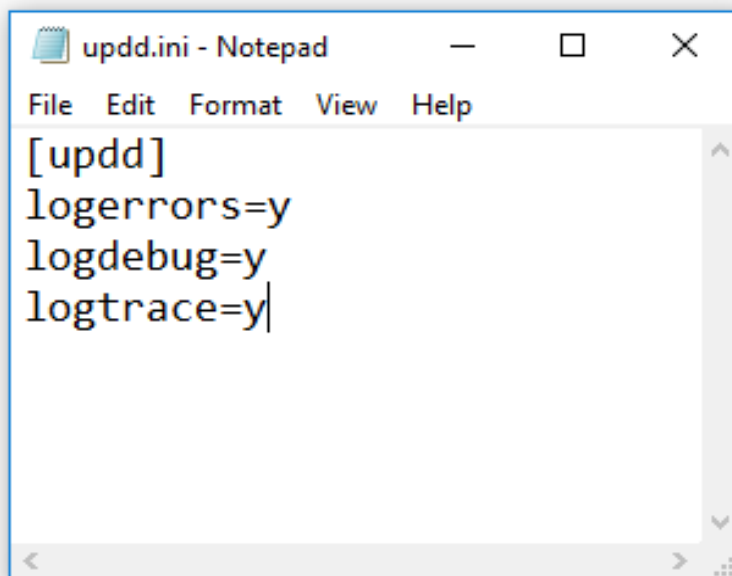
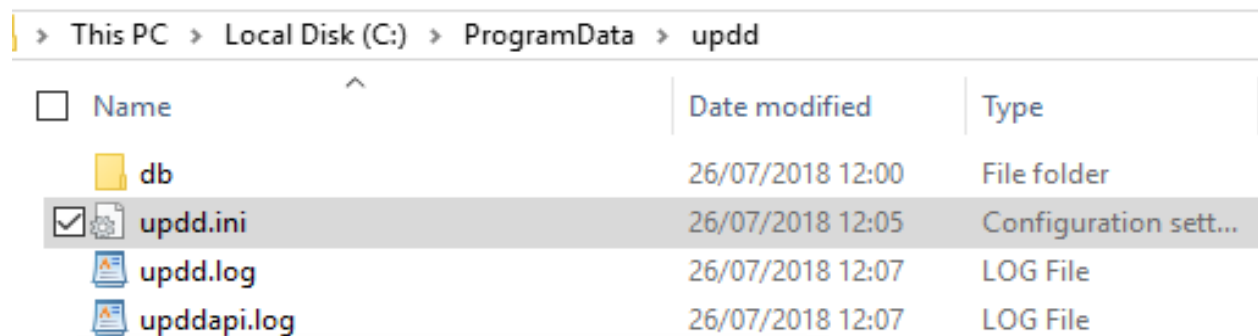
Driver debugging

If using UPDD 6.1.396 or above use the command line utility to [enable / disable boot logging](#).

For earlier versions, you need to create the driver's bootstrap file, updd.ini, and add the content:

```
[updd]
logerrors=y
logdebug=y
logtrace=y
```

As per this Windows example:



Driver debugging

optionally, we may also request that you set these other options

logsql

logdebughigh

logconsole

logtrace

This is useful to log errors that are occurring in a UPDD API based program.

You can download a pre-defined updd.ini file from here - [Windows](#), MacOS, Linux.

The bootstrap file must be placed in the following folder:

Windows	c:\programdata\updd
MacOS	/Library/Preferences/updd/
Linux	/opt/updd/data/updd

The log file is placed in

Windows	c:\programdata\updd\logs
MacOS	/Library/Preferences/updd/logs
Linux	/opt/updd/data/updd/logs

Note: Prior to UPDD 6.0.500 the log file was written to the same folder as the bootstrap file.

Log files

The log files upddapi.log and updd.log will be created in the same folder as the bootstrap file, as seen in the above Windows example.

Remember to delete these entries/file to disable logging otherwise eventually a massive file will be created!

Touch-Base Support

<http://support.touch-base.com/Documentation/50359/Driver-debugging>