
MDT for Android Instruction

February 19, 2016

The NeuroSky® product families consist of hardware and software components for simple integration of this biosensor technology into consumer and industrial end-applications. All products are designed and manufactured to meet consumer thresholds for quality, pricing, and feature sets. NeuroSky sets itself apart by providing building block component solutions that offer friendly synergies with related and complementary technological solutions.

NO WARRANTIES: THE NEUROSKY PRODUCT FAMILIES AND RELATED DOCUMENTATION IS PROVIDED "AS IS" WITHOUT ANY EXPRESS OR IMPLIED WARRANTY OF ANY KIND INCLUDING WARRANTIES OF MERCHANTABILITY, NONINFRINGEMENT OF INTELLECTUAL PROPERTY, INCLUDING PATENTS, COPYRIGHTS OR OTHERWISE, OR FITNESS FOR ANY PARTICULAR PURPOSE. IN NO EVENT SHALL NEUROSKY OR ITS SUPPLIERS BE LIABLE FOR ANY DAMAGES WHATSOEVER (INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF PROFITS, BUSINESS INTERRUPTION, COST OF REPLACEMENT GOODS OR LOSS OF OR DAMAGE TO INFORMATION) ARISING OUT OF THE USE OF OR INABILITY TO USE THE NEUROSKY PRODUCTS OR DOCUMENTATION PROVIDED, EVEN IF NEUROSKY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. , SOME OF THE ABOVE LIMITATIONS MAY NOT APPLY TO YOU BECAUSE SOME JURISDICTIONS PROHIBIT THE EXCLUSION OR LIMITATION OF LIABILITY FOR CONSEQUENTIAL OR INCIDENTAL DAMAGES.

USAGE OF THE NEUROSKY PRODUCTS IS SUBJECT TO AN END-USER LICENSE AGREEMENT.

“Made for iPod,” “Made for iPhone,” and “Made for iPad” mean that an electronic accessory has been designed to connect specifically to iPod, iPhone, or iPad, respectively, and has been certified by the developer to meet Apple performance standards. Apple is not responsible for the operation of this device or its compliance with safety and regulatory standards. Please note that the use of this accessory with iPod, iPhone, or iPad may affect wireless performance.

Contents

[Introduction to MDT](#)

[Introduction to MDT for Android](#)

[Usage](#)

[References and Bug reports](#)

Introduction to MDT

NeuroSky's **Mind Developer Tools** (hereafter abbreviated **MDT** or **Developer Tools**) are a set of software tools that make it easy to create innovative applications that respond to a user's brainwaves and mental state.

If you already have a NeuroSky headset (such as **MindWave Mobile**), you will be able to take full advantage of it with our Developer Tools. If you are trying out the Developer Tools before purchasing a headset, thank you for reviewing the toolset. However, please note the NeuroSky headset is needed while using the developer tools to develop your own app. Our headset is available on Amazon store. To your convenience, here's the direct link to it

http://www.amazon.com/NeuroSky-MindWave-Mobile-BrainWave-Starter/dp/B00B8BF4EM/ref=cm_cr_pr_product_top?ie=UTF8 .

If you have any questions, let us know at support@neurosky.com.

Introduction to MDT for Android

The **MDT for Android** includes: **Application Standards**, **Stream SDK for Android** and **EEG Algorithm SDK for Android** and **EULA**:

- **Application Standards** Document and Icon Images
- **EULA** End User License Agreement
- **Stream SDK for Android** is used to help connect your Android app to a NeuroSky headset via bluetooth, and receive data from headset. It contains the follows file:
 - Stream SDK for Android Development Guide
 - ChangeLog.txt
 - libStreamSDK.jar
 - libNSUART.so
 - Sample Project
 - demo (.apk file)
- **EEG Algorithm SDK for Android** is used to analyze and further interpret EEG data from NeuroSky's headset or TGAM module. It includes Attention, Meditation and Eye Blink Detection. These three algorithms are free to use within your application. It contains the follows file:
 - EEG Algorithm SDK for Android: Development Guide
 - EEG Algorithm SDK library: libs/
 - <CPUARCH>/libNskAlgo.so (compatibleCPUarchitectures: arm64-v8a, armeabi, armeabi-v7a, mips, mips64, x86, x86_64)
 - EEG Algorithm SDK Java interface: jar/
 - NskAlgoSdk.jar
 - Algo SDK Sample project

If you want more information about other EEG algorithms, please contact us at support@neurosky.com.

For details, please check within each package.

Usage

Each SDK includes a sample project and documents which teach you how to use them.

In order to make the integration progress for SDKs more smooth, please review the documents of SDKs. For example, review “**Stream SDK for Android Development Guide.pdf**” to start with the Stream SDK. For EEG Algorithm SDK, please review “**eeg_algorithm_sdk_for_android_development_guide.pdf**”.

“**ApplicationStandards.pdf**” is very useful. It tells you how to use the icons to mark the status of connection in your project.

Here is a code snippet which shows how to use these SDK together (This code snippet is from **EEG Algorithm SDK**’s sample project):

```
private TgStreamHandler callback = new TgStreamHandler() {

    @Override
    public void onStatesChanged(int connectionStates) {
        // code for handle state changes
    }

    @Override
    public void onRecordFail(int flag) {
        // You can handle the record error message here
        Log.e(TAG,"onRecordFail: " +flag);
    }

    @Override
    public void onChecksumFail(byte[] payload, int length, int checksum) {
        // You can handle the bad packets here.
    }

    @Override
    public void onDataReceived(int datatype, int data, Object obj) {
        // You can handle the received data here
        // You can feed the raw data to algo sdk here if necessary.
        //Log.i(TAG,"onDataReceived");
        switch (datatype) {
            case MindDataType.CODE_ATTENTION:
                short attValue[] = {(short)data};
                nskAlgoSdk.NskAlgoDataStream(    NskAlgoDataType.NSK_ALGO_DATA_TYPE_ATT.value,
attValue, 1);
                break;
            case MindDataType.CODE_MEDITATION:
```

```

        short medValue[] = {(short)data};
        nskAlgoSdk.NskAlgoDataStream( NskAlgoDataType.NSK_ALGO_DATA_TYPE_MED.value,
medValue, 1);
        break;
        case MindDataType.CODE_POOR_SIGNAL:
            short pqValue[] = {(short)data};
            nskAlgoSdk.NskAlgoDataStream( NskAlgoDataType.NSK_ALGO_DATA_TYPE_PQ.value,
pqValue, 1);
            break;
        case MindDataType.CODE_RAW:
            raw_data[raw_data_index++] = (short)data;
            if (raw_data_index == 512) {
                nskAlgoSdk.NskAlgoDataStream( NskAlgoDataType.NSK_ALGO_DATA_TYPE_EEG.value,
raw_data, raw_data_index);
                raw_data_index = 0;
            }
            break;
        default:
            break;
    }
}
};

```

TgStreamHandler is used to get information returned by Stream SDK, and onDataReceived function is used to receive data. You can pass the data to EEG SDK here.

References and Bug reports

You can get the latest developer information from here:

<http://developer.neurosky.com/>

Learn about NeuroSky's EEG Data Types here:

http://developer.neurosky.com/docs/doku.php?id=thinkgear_communications_protocol

You may find some additional useful information in the Knowledge Base:

<http://support.neurosky.com/kb>

If you find any bugs, please contact us at:

support@neurosky.com