

CBManager Class Reference

(version 0.5 draft)

Overview

The CBManager class handles connections and data transfer between Braniac (alpha title) accessory and iOS device.

Tasks

Getting the CBManager Accessory Manager

```
CBManager *cbManager = [[CBManager alloc] init];  
cbManager.delegate = self;
```

Setting Up CBManager

- yellowFlagLow (float)
- yellowFlagHigh (float)
- red1Flag (float)
- red2Flag (float)

Starting and Stopping Connection and Data Stream

- start
- stop

Getting Information about CBManager state

- `hasStarted` *property*
- `rawdata` (NSMutableData *)
- `raw values` (NSMutableArray *)
- `fftData` (NSMutableArray *)

Receiving information about examinee

- `processGreenForChannel`
- `processYellowForChannel`
- `processRed1ForChannel`
- `processRed2ForChannel`

Accessing the Delegate

- `delegate` *property*

Properties

hasStarted

A BOOL value indicating if the current CBManager instance object working: connected to hardware Braniac accessory and receiving data.
Read-only

```
@property (nonatomic, assign, readonly) BOOL  
hasStarted;
```

rawdata

A NSMutableData object containing the unprocessed raw data bytes received from hardware Brainiac accessory. Read-only

```
@property (strong, nonatomic, readonly)
NSMutableData *rawdata;
```

rawvalues

A NSMutableArray object containing processed data values received from hardware Brainiac accessory including hardware order number, double values from each 4 channels, time tick order number. Read-only

```
@property (strong, nonatomic, readonly)
NSMutableArray *rawvalues;
```

fftData

A NSMutableArray object containing processed fast-FFT values made from each 256 raw data values for each channel. Includes FFT double values for each channel, serial order number and time tick value. Read-only

```
@property (strong, nonatomic, readonly)
NSMutableArray *fftData;
```

delegate

The object that acts as the delegate (CBManagerDelegate) of the accessory. All data receiving and status reporting goes through it.

```
@property (nonatomic, strong) id
<CBManagerDelegate> delegate;
```

yellowFlagLow

Bool property used to set the low limit of dominant frequency signal changing to detect relaxation of brain activity. The default value is 0.2

```
@property (nonatomic, assign) double  
yellowFlagLow;
```

yellowFlagHigh

Bool property used to set the high limit of dominant frequency signal changing to detect relaxation of brain activity. The default value is 0.3

```
@property (nonatomic, assign) double  
yellowFlagHigh;
```

red1Flag

Bool property used to set the minimum limit of dominant frequency signal changing to detect overexcitement of brain activity. The default value is 0.2

```
@property (nonatomic, assign) double red1Flag;
```

red2Flag

Bool property used to set the minimum limit of dominant frequency signal changing to detect damping of brain activity. The default value is 0.3

```
@property (nonatomic, assign) double red2Flag;
```

Initialisation and Creating CBManager object

init

Returns CBManager object which is ready to start connecting. Should always be used to start working with hardware Braniac accessory.

```
CBManager *cbManager = [[CBManager alloc] init];  
cbManager.delegate = self;
```

start

Starts connecting to hardware Braniac accessory and receiving data and processing data if connect is successful. Starts dispatching data via delegate methods immediately. Set hasStarted property to true.

```
-(void)start;
```

stop

Stops sending data via delegate methods, disconnecting from hardware Braniac accessory. Set hasStarted property to false.

```
-(void)stop;
```

Receiving periodical information about examinee

processGreenForChannel

Returns flag which define the state of examined man detecting if the state of brain activity is full and active. Defined as: When closing the eyes or with simple contemplation of neutral images dominant frequency in the range of alpha (7–13 Hz) has no oscillations more than 20% for 3 minutes during the registration process. Method returns flag for such activity for last 5 sec (so app should call this method each 5 sec to get trend activity)

`-(BOOL)processGreenForChannel:(NSInteger)channel;`

Parameter

- **channel** - Number for processing channel (1–4)

processYellowForChannel

Returns flag which define the state of examined man detecting if the state of brain activity is Relaxation brain activity (EEG spectrum for simply “nice” relaxation, during which the person can not adequately drive or write software). Defined as: the dominant frequency in the range of alpha (7–13 Hz) increases in amplitude (power spectrum) on greater than 20% but less than 30% within 3 minutes. Method returns flag for such activity for last 5 sec (so app should call this method each 5 sec to get trend activity)

`-(BOOL)processYellowForChannel:(NSInteger)channel;`

Parameter

- **channel** - Number for processing channel (1–4)

processRed1ForChannel

Returns flag which define the state of examined man detecting if the state of brain activity is Excessive stimulation of neurons and therefore the beginning of inappropriate, excessive actions. Defined as: the dominant frequency (range) of alpha (7–13 Hz) is reduced in amplitude (power spectrum) on greater than 20% for 3 minutes. Method returns flag for such activity for last 5 sec (so app should call this method each 5 sec to get trend activity)

– (BOOL)processRed1ForChannel:(NSInteger)channel;

Parameter

- **channel** - Number for processing channel (1–4)

processRed2ForChannel

Returns flag which define the state of examined man detecting if the state of brain activity is in super relaxation. Defined as: the dominant frequency (range) of alpha (7–13 Hz) is increasing in amplitude (power spectrum) on greater than 30% for 3 minutes. Method returns flag for such activity for last 5 sec (so app should call this method each 5 sec to get trend activity)

– (BOOL)processRed2ForChannel:(NSInteger)channel;

Parameter

- **channel** - Number for processing channel (1–4)

CBManagerDelegate Protocol

Reference

Overview

The CBManagerDelegate protocol defines the methods for handling accessory data flow and status messages dispatched from CBManager object.

Tasks

CB_dataUpdatedWithDictionary:

Required method which returns brain activity data values for each of 4 channels.

```
-(void)CB_dataUpdatedWithDictionary:(NSDictionary*)data;
```

Parameters

- **data** - NSDictionary objects with returned data values with NSStrings as keys. See below the contents of data structure.

Description

- *hardware_order_number* - hardware order number identifying number of data packet - *Short*

- *timeframe* - number of milliseconds passed from 1970 year for each packet - *NSString*
- *counter* - serial internal order number identifying number of data packet - *NSInteger*
- *ch1* - brain activity measure for channel 1 (T3) - *double*
- *ch2* - brain activity measure for channel 2 (O1) - *double*
- *ch3* - brain activity measure for channel 3 (T4) - *double*
- *ch4* - brain activity measure for channel 4 (O2) - *double*

CB_fftDataUpdatedWithDictionary

Optional method returning FFT data processed each 1 sec (so for 250 data packets). Each FFT processes packet contains dominant frequencies values for each channel and each range (total 3 range for each channel)

```
-(void)CB_fftDataUpdatedWithDictionary:
(NSDictionary *)data;
```

Parameters

- **data** - NSDictionary objects with returned data values with NSStrings as keys. See below the contents of data structure.

Description

- *timeframe* - number of milliseconds passed from 1970 year for each packet - *NSString*

- *counter* - serial internal order number identifying number of data packet - *NSInteger*
- *ch1* - brain activity FFT dictionary processed for channel 1 - *NSDictionary*
- *ch2* - brain activity FFT dictionary processed for channel 2 - *NSDictionary*
- *ch3* - brain activity FFT dictionary processed for channel 3 - *NSDictionary*
- *ch4* - brain activity FFT dictionary processed for channel 4 - *NSDictionary*

FFT dictionary structure

- *data1* - dominant frequency value for current time range and frequencies range 3–7 Hz - *double*
- *data2* - dominant frequency value for current time range and frequencies range 7–13 Hz - *double*
- *data3* - dominant frequency value for current time range and frequencies range 14–24 Hz - *double*

CB_changedStatus

Optional method returning status messages from CBManager object instance

```
-(void)CB_changedStatus:(NSString *)statusMessage;
```

Parameters

- **statusMessage** - NSString object with text message describing current state of CBManager object instance.