Medtronic	NEUROMODULATION CONFIDENTIAL	DOCUMENT/RECORD

This document/record is electronically controlled, printed copies are considered uncontrolled.

Identifier	Version	Author
NRP1088-36867	4.0	Benjamin Isaacson

Title: Interface Specification, Nexus-D API

Pages: (including this page) 129

APPROVALS

Signed By	Responsibility	Date/Time (GMT)
Johnsen, Tom	Regulatory Affairs Approver	05/22/2014 07:31:59 PM
Benjamin Isaacson	Author / SW Design Lead	05/22/2014 07:33:24 PM
Bourget, Duane	SWVT approver	05/22/2014 07:45:56 PM

Nexus-D API

Interface Specification

CAUTION – Investigational device. Limited by Federal (or United States) law to investigational use.

1. Scope

This document is intended to be a supplement to the User Manual, Nexus-D API. It contains the lower level technical details that an engineer intending to programmatically interface to the Nexus-D API would be interested in. It does not contain any details about the Nexus-D System Interface.

2. References

Document	Title
NRP1088-36356	User Manual, Nexus-D API
NRP1088-35790	Interface Specification, Nexus-D System

Nexus-D API Documentation

Package mdt.neuro.nexus

Interface Summary		Page
IDisposable	The Interface IDisposable.	*19*

Class Summary		Page
ApplicationCommand	The Class ApplicationCommand.	*4*
CommandAccessor	The Class CommandAccessor.	*9*
DataPacketAccessor	The Class DataPacketAccessor.	*17*
NexusInstrument	The Class NexusInstrument.	*20*
ObservableConnection	The Class ObservableConnection.	*30*
ParameterizedRunnable	The Class ParameterizedRunnable.	*34*
SerialConnection	The Class Connection.	*39*
SerialPortErrors	The Class SerialPortErrors.	*42*
ThreadedNexusInstrument	The Class ThreadedNexusInstrument.	*43*
ThreadManager	The Class ThreadManager.	*50*
Util	The Class Util.	*52*

Enum Summary		Page
ConnectionStatus	The Enum ConnectionStatus.	*15*
ResponseCodes	The Enum ResponseCodes.	*36*

Class ApplicationCommand

mdt.neuro.nexus

```
java.lang.Object
  └ mdt.neuro.nexus.ApplicationCommand
```

Direct Known Subclasses:

CycleStim, DecAmplitude, DecFrequency, DecPulseWidth, GetInsInfo, GetNexusStatus, GetRealtimeData, IncAmplitude, IncFrequency, IncPulseWidth, PulseStim, ResetCycle, RestoreClinicianSettings, SendTrigger, SetActiveGroup, SetNexusConfiguration, StartRealTime, StartSensing, StopRealTime, StopSensing, TherapyOff, TherapyOn

---

```
public class ApplicationCommand
extends Object
```

The Class ApplicationCommand. This class represents a command to be sent to the Nexus-2 system.

Field Summary		Page
protected List<Byte>	**commandBytes** The command data (command data plus command code).	6
protected short	**commandCode** The command code.	6
protected boolean	**crcValid** The crc valid.	6
protected List<Byte>	**request** The request.	7
protected List<Byte>	**response** The response.	6
protected short	**responseCode** The response code.	6
protected List<Byte>	**responseData** The response data (not including crc, command code, or response code).	6
protected int	**timeout** The timeout.	6

Constructor Summary		Page
protected	**ApplicationCommand**(List<Byte> bytes, int timeoutMs) Instantiates a new application command.	7
protected	**ApplicationCommand**(short code, int timeoutMs) Instantiates a new application command.	7

Method Summary		Page
protected void	**consumeBytes**(List<Byte> bytes)	7
protected List<Byte>	**getRequestData**() Gets the request data (does not include CRC).	7
protected List<Byte>	**getResponseBytes**() Gets the response bytes that comprise this payload (including CRC).	8
protected List<Byte>	**getResponseData**() Gets the response data (does not include CRC).	7

Field Detail

commandCode

```
protected short commandCode
```

The command code.

---

commandBytes

```
protected List<Byte> commandBytes
```

The command data (command data plus command code).

---

responseCode

```
protected short responseCode
```

The response code.

---

responseData

```
protected List<Byte> responseData
```

The response data (not including crc, command code, or response code).

---

crcValid

```
protected boolean crcValid
```

The crc valid.

---

timeout

```
protected int timeout
```

The timeout.

---

response

```
protected final List<Byte> response
```

The response.

---

request

```
protected final List<Byte> request
```

The request.

Constructor Detail

ApplicationCommand

```
protected ApplicationCommand(short code,
                             int timeoutMs)
```

Instantiates a new application command.

Parameters:

`code` - the command code
`timeoutMs` - the command timeout (ms)

ApplicationCommand

```
protected ApplicationCommand(List<Byte> bytes,
                             int timeoutMs)
```

Instantiates a new application command.

Parameters:

`bytes` - the bytes to be sent
`timeoutMs` - the timeout ms

Method Detail

getRequestData

```
protected List<Byte> getRequestData()
```

Gets the request data (does not include CRC).

Returns:

the request data

---

consumeBytes

```
protected void consumeBytes(List<Byte> bytes)
```

---

getResponseData

```
protected List<Byte> getResponseData()
```

Gets the response data (does not include CRC).

Returns:
the response data

---

getResponseBytes

```
protected List<Byte> getResponseBytes()
```

Gets the response bytes that comprise this payload (including CRC).

Returns:
the response bytes

Class CommandAccessor

mdt.neuro.nexus

java.lang.Object
  └ **mdt.neuro.nexus.CommandAccessor**

---

abstract public class **CommandAccessor**
extends Object

The Class CommandAccessor. This class allows friend access to the creation of commands by internal API classes not in the commands package.

Constructor Summary	Page
CommandAccessor()	10

Method Summary		Page
protected abstract CycleStim	**getCycleStimCommand**() Gets the cycle stim command.	10
protected abstract DecAmplitude	**getDecAmplitudeCommand**(byte progNum, byte numSteps, byte repeats) Gets the dec amplitude command.	10
protected abstract DecFrequency	**getDecFrequencyCommand**(byte groupNum) Gets the dec frequency command.	11
protected abstract DecPulseWidth	**getDecPulseWidthCommand**(byte progNum) Gets the dec pulse width command.	11
protected abstract GetInsInfo	**getGetInsInfoCommand**() Gets the gets the ins info command.	11
protected abstract GetNexusStatus	**getGetNexusStatusCommand**() Gets the gets the nexus status command.	11
protected abstract GetRealtimeData	**getGetRealTimeDataCommand**() Gets the gets the real time data command.	11
protected abstract IncAmplitude	**getIncAmplitudeCommand**(byte progNum, byte numSteps, byte repeats) Gets the inc amplitude command.	12
protected abstract IncFrequency	**getIncFrequencyCommand**(byte groupNum) Gets the inc frequency command.	12
protected abstract IncPulseWidth	**getIncPulseWidthCommand**(byte progNum) Gets the inc pulse width command.	12
protected abstract ResetCycle	**getResetCycleCommand**() Gets the reset cycle command.	12
protected abstract RestoreClinicianSettings	**getRestoreClinicianSettingsCommand**(byte groupNumber) Gets the restore clinician settings command.	13
protected abstract SendTrigger	**getSendTriggerCommand**() Gets the send trigger command.	13
protected abstract SetActiveGroup	**getSetActiveGroupCommand**(byte gpNum) Gets the sets the active group command.	13
protected abstract SetNexusConfiguration	**getSetNexusConfigurationCommand**(byte maintSessTimeoutSec, byte hostSessTimeoutMin) Gets the sets the nexus configuration command.	13
protected abstract StartRealTime	**getStartRealTimeCommand**(boolean channel1) Gets the start real time command.	13

protected abstract StartSensing	**getStartSensingCommand**() Gets the start sensing command.	*14*
protected abstract StopRealTime	**getStopRealTimeCommand**() Gets the stop real time command.	*14*
protected abstract StopSensing	**getStopSensingCommand**() Gets the stop sensing command.	*14*
protected abstract TherapyOff	**getTherapyOffCommand**() Gets the therapy off command.	*14*
protected abstract TherapyOn	**getTherapyOnCommand**() Gets the therapy on command.	*14*
static void	**setProvider**(CommandAccessor provider) Sets the provider.	*10*

Constructor Detail

CommandAccessor

```
public CommandAccessor()
```

Method Detail

setProvider

```
public static void setProvider(CommandAccessor provider)
```

Sets the provider.

Parameters:
`provider` - the new provider

---

getCycleStimCommand

```
protected abstract CycleStim getCycleStimCommand()
```

Gets the cycle stim command.

Returns:
the cycle stim command

---

getDecAmplitudeCommand

```
protected abstract DecAmplitude getDecAmplitudeCommand(byte progNum,
                                                       byte numSteps,
                                                       byte repeats)
```

Gets the dec amplitude command.

Parameters:
`progNum` - the prog num
`numSteps` - the num steps

repeats - the repeats

Returns:

the dec amplitude command

---

getDecFrequencyCommand

```
protected abstract DecFrequency getDecFrequencyCommand(byte groupNum)
```

Gets the dec frequency command.

Parameters:

groupNum - the group num

Returns:

the dec frequency command

---

getDecPulseWidthCommand

```
protected abstract DecPulseWidth getDecPulseWidthCommand(byte progNum)
```

Gets the dec pulse width command.

Parameters:

progNum - the prog num

Returns:

the dec pulse width command

---

getGetInsInfoCommand

```
protected abstract GetInsInfo getGetInsInfoCommand()
```

Gets the gets the ins info command.

Returns:

the gets the ins info command

---

getGetNexusStatusCommand

```
protected abstract GetNexusStatus getGetNexusStatusCommand()
```

Gets the gets the nexus status command.

Returns:

the gets the nexus status command

---

getGetRealTimeDataCommand

```
protected abstract GetRealtimeData getGetRealTimeDataCommand()
```

Gets the gets the real time data command.

Returns:
the gets the real time data command

---

getIncAmplitudeCommand

```
protected abstract IncAmplitude getIncAmplitudeCommand(byte progNum,
                                                       byte numSteps,
                                                       byte repeats)
```

Gets the inc amplitude command.

Parameters:
`progNum` - the prog num
`numSteps` - the num steps
`repeats` - the repeats
Returns:
the inc amplitude command

---

getIncFrequencyCommand

```
protected abstract IncFrequency getIncFrequencyCommand(byte groupNum)
```

Gets the inc frequency command.

Parameters:
`groupNum` - the group num
Returns:
the inc frequency command

---

getIncPulseWidthCommand

```
protected abstract IncPulseWidth getIncPulseWidthCommand(byte progNum)
```

Gets the inc pulse width command.

Parameters:
`progNum` - the prog num
Returns:
the inc pulse width command

---

getResetCycleCommand

```
protected abstract ResetCycle getResetCycleCommand()
```

Gets the reset cycle command.

Returns:
the reset cycle command

---

getRestoreClinicianSettingsCommand

```
protected abstract RestoreClinicianSettings getRestoreClinicianSettingsCommand(byte groupNumbe
r)
```

Gets the restore clinician settings command.

Parameters:
`groupNumber` - the group number
Returns:
the restore clinician settings command

---

getSendTriggerCommand

```
protected abstract SendTrigger getSendTriggerCommand()
```

Gets the send trigger command.

Returns:
the send trigger command

---

getSetActiveGroupCommand

```
protected abstract SetActiveGroup getSetActiveGroupCommand(byte gpNum)
```

Gets the sets the active group command.

Parameters:
`gpNum` - the gp num
Returns:
the sets the active group command

---

getSetNexusConfigurationCommand

```
protected abstract SetNexusConfiguration getSetNexusConfigurationCommand(byte maintSessTimeout
Sec,
                                                                         byte hostSessTimeoutM
in)
```

Gets the sets the nexus configuration command.

Parameters:
`maintSessTimeoutSec` - the maint sess timeout sec
`hostSessTimeoutMin` - the host sess timeout min
Returns:
the sets the nexus configuration command

---

getStartRealTimeCommand

```
protected abstract StartRealTime getStartRealTimeCommand(boolean channel1)
```

Gets the start real time command.

Parameters:
`channel1` - the channel1
Returns:
the start real time command

---

getStartSensingCommand

```
protected abstract StartSensing getStartSensingCommand()
```

Gets the start sensing command.

Returns:
the start sensing command

---

getStopRealTimeCommand

```
protected abstract StopRealTime getStopRealTimeCommand()
```

Gets the stop real time command.

Returns:
the stop real time command

---

getStopSensingCommand

```
protected abstract StopSensing getStopSensingCommand()
```

Gets the stop sensing command.

Returns:
the stop sensing command

---

getTherapyOffCommand

```
protected abstract TherapyOff getTherapyOffCommand()
```

Gets the therapy off command.

Returns:
the therapy off command

---

getTherapyOnCommand

```
protected abstract TherapyOn getTherapyOnCommand()
```

Gets the therapy on command.

Returns:
the therapy on command

Enum ConnectionStatus

mdt.neuro.nexus

```
java.lang.Object
  └ java.lang.Enum<ConnectionStatus>
      └ mdt.neuro.nexus.ConnectionStatus
```

All Implemented Interfaces:

Comparable<ConnectionStatus>, Serializable

```
public enum ConnectionStatus
extends Enum<ConnectionStatus>
```

The Enum ConnectionStatus.

Enum Constant Summary	Page
Connected Communication established.	15
Connecting In the process of establishing communication.	15
Disconnected Communication is not established.	15

Method Summary		Page
static ConnectionStatus	**valueOf**(String name)	16
static ConnectionStatus[]	**values**()	16

Enum Constant Detail

Connected

```
public static final ConnectionStatus Connected
```

Communication established.

Connecting

```
public static final ConnectionStatus Connecting
```

In the process of establishing communication.

Disconnected

```
public static final ConnectionStatus Disconnected
```

Communication is not established.

Method Detail

values

```
public static ConnectionStatus[] values()
```

---

valueOf

```
public static ConnectionStatus valueOf(String name)
```

Class DataPacketAccessor

mdt.neuro.nexus

```
java.lang.Object
  └ mdt.neuro.nexus.DataPacketAccessor
```

```
abstract public class DataPacketAccessor
extends Object
```

The Class DataPacketAccessor. This class defines methods to allow access to DataPacket methods to other internal classes of the API

Constructor Summary	Page
DataPacketAccessor()	17

Method Summary		Page
protected abstract void	**setNumMissedPatterns**(DataPacket pkt, short misses) Sets the num missed packets.	18
protected abstract void	**setPacketNum**(DataPacket pkt, int index, short num) Sets the packet num.	17
static void	**setProvider**(DataPacketAccessor provider) Sets the provider.	17

Constructor Detail

DataPacketAccessor

```
public DataPacketAccessor()
```

Method Detail

setProvider

```
public static void setProvider(DataPacketAccessor provider)
```

Sets the provider.

Parameters:

`provider` - the new provider

---

setPacketNum

```
protected abstract void setPacketNum(DataPacket pkt,
                                     int index,
                                     short num)
```

Sets the packet num.

Parameters:

`pkt` - the pkt
`index` - the index
`num` - the num

---

setNumMissedPatterns

```
protected abstract void setNumMissedPatterns(DataPacket pkt,
                                             short misses)
```

Sets the num missed packets.

Parameters:

`pkt` - the pkt
`misses` - the misses

Interface IDisposable

mdt.neuro.nexus

All Known Implementing Classes:
ObservableConnection, SerialConnection

```
public interface IDisposable
```

The Interface IDisposable.

Method Summary		Page
void	**dispose**() Dispose of this instance.	19

Method Detail

dispose

```
void dispose()
```

Dispose of this instance.

Class NexusInstrument

mdt.neuro.nexus

```
java.lang.Object
  └ java.util.Observable
      └ mdt.neuro.nexus.NexusInstrument
```

All Implemented Interfaces:

Observer

```
public class NexusInstrument
extends Observable
implements Observer
```

The Class NexusInstrument. This class is the main abstraction of the Nexus-2 system interface to the host application.

Method Summary		Page
List<Byte>	**arbitraryCommand**(List<Byte> bytes, int timeoutMs) Sends a command of arbitrary bytes.	23
double[][]	**computeTemplate**(DataPacket[] pkts) Computes a template based on passed in data packets that can then be subtracted from other data packets to reduce artifact induced by telemetry.	29
int	**connect**(ObservableConnection conn) Connect this instance and initializes the internal API classes required for communication.	22
int	**cycleStim**() Cycle stim on the device (off then back on).	28
double	**decStimParameter**(TherapyParameter type, byte progNum) Decrements the given stimulation parameter.	24
double	**decStimParameter**(TherapyParameter type, byte progNum, byte numSteps, byte numRepeats) Decrements the given stimulation parameter.	24
boolean	**disconnect**() Disconnects this instance.	22
void	**dispose**() Disposes of this instance.	22
DataPacket	**getDataPacket**() Gets a data packet from the INS.	25
InsInfo	**getInsInfo**() Gets the INS info, including stimulation settings and sensing status.	25
static NexusInstrument	**getInstance**() Gets the single instance of NexusInstrument.	22
int	**getLastInsResponseCode**() Gets the last ins response code.	22
ResponseCodes	**getLastNexusResponseCode**() Gets the last nexus response code.	22
NexusStatus	**getNexusStatus**() Gets the status of the Nexus-D system.	25

String	**getVersion**() Gets the version of the API.	21
double	**incStimParameter**(TherapyParameter type, byte progNum) Increments the given stimulation parameter.	23
double	**incStimParameter**(TherapyParameter type, byte progNum, byte numSteps, byte numRepeats) Increments the given stimulation parameter.	24
double	**pulseStimParameter**(byte progNum, byte numSteps, byte numDelay) Sends a batched inc/dec to allow for fine control of the number of stimulation pulses that are delivered at a given level.	29
int	**resetCycle**() Resets the stimulation cycle.	28
int	**restoreClinicianSettings**(byte groupNumber) Restores the clinician settings saved in the INS.	28
int	**sendTrigger**() Send a trigger to the INS.	28
int	**setActiveGroup**(byte groupNum) Sets the active group.	27
int	**setNexusConfiguration**(byte maintSessionTimeoutSec, byte hostSessionTimeoutMin) Sets the nexus configuration.	25
int	**startDataSession**() Starts a data session with the INS.	26
int	**startDataSession**(boolean channelOne) Starts a data session with the INS.	26
int	**startSensing**() Enables the sensing feature in the INS.	26
int	**stopDataSession**() Stops a data session.	27
int	**stopSensing**() Disables the sensing feature in the INS.	26
int	**therapyOff**() Turns therapy off.	27
int	**therapyOn**() Turns therapy on.	27
String	**toString**()	23
void	**update**(Observable o, Object arg)	23

Method Detail

getVersion

```
public String getVersion()
```

Gets the version of the API.

Returns:
the version

getLastNexusResponseCode

```
public ResponseCodes getLastNexusResponseCode()
```

Gets the last nexus response code.

Returns:
the nexusResponseCode

---

getLastInsResponseCode

```
public int getLastInsResponseCode()
```

Gets the last ins response code.

Returns:
the insResponseCode

---

getInstance

```
public static synchronized NexusInstrument getInstance()
```

Gets the single instance of NexusInstrument.

Returns:
single instance of NexusInstrument

---

connect

```
public int connect(ObservableConnection conn)
```

Connect this instance and initializes the internal API classes required for communication.

Parameters:
`conn` - the observable connection to connect to (e.g. serial connection)
Returns:
result of the operation (0 for success)

---

disconnect

```
public boolean disconnect()
```

Disconnects this instance.

Returns:
true, if disconnect was successful

---

dispose

```
public void dispose()
```

Disposes of this instance.

---

update

```
public void update(Observable o,
                   Object arg)
```

Specified by:
update in interface Observer

---

toString

```
public String toString()
```

Overrides:
toString in class Object

---

arbitraryCommand

```
public List<Byte> arbitraryCommand(List<Byte> bytes,
                                   int timeoutMs)
                            throws InterruptedException
```

Sends a command of arbitrary bytes.

Parameters:
bytes - the bytes that compose the command. This includes the command code and parameter bytes but does not include the CRC.
timeoutMs - the timeout ms
Returns:
the object returned
Throws:
InterruptedException - the interrupted exception

---

incStimParameter

```
public double incStimParameter(TherapyParameter type,
                               byte progNum)
                        throws InterruptedException
```

Increments the given stimulation parameter.

Parameters:
type - the parameter type
progNum - the program number
Returns:
the resulting TherapyParameter value as a result of the operation. -1 if the operation failed. The resulting amplitude value is calculated based on the last known AmplitudeResolution (V or CC). If this is changed without sending a GetInsInfo command to give the API knowledge of the change, the returned value will be incorrect
Throws:
InterruptedException - the interrupted exception

incStimParameter

```
public double incStimParameter(TherapyParameter type,
                               byte progNum,
                               byte numSteps,
                               byte numRepeats)
                        throws InterruptedException
```

Increments the given stimulation parameter.

Parameters:

`type` - the parameter type
`progNum` - the program number
`numSteps` - the number of steps to increment (only for amplitude)
`numRepeats` - the num repeats (only for amplitude)

Returns:

the resulting `TherapyParameter` value as a result of the operation. -1 if the operation failed. The resulting amplitude value is calculated based on the last known `AmplitudeResolution` (V or CC). If this is changed without sending a `GetInsInfo` command to give the API knowledge of the change, the returned value will be incorrect

Throws:

`InterruptedException` - the interrupted exception

decStimParameter

```
public double decStimParameter(TherapyParameter type,
                               byte progNum)
                        throws InterruptedException
```

Decrements the given stimulation parameter.

Parameters:

`type` - the parameter type
`progNum` - the program number

Returns:

the resulting `TherapyParameter` value as a result of the operation. -1 if the operation failed. The resulting amplitude value is calculated based on the last known `AmplitudeResolution` (V or CC). If this is changed without sending a `GetInsInfo` command to give the API knowledge of the change, the returned value will be incorrect

Throws:

`InterruptedException` - the interrupted exception

decStimParameter

```
public double decStimParameter(TherapyParameter type,
                               byte progNum,
                               byte numSteps,
                               byte numRepeats)
                        throws InterruptedException
```

Decrements the given stimulation parameter.

Parameters:

`type` - the parameter type
`progNum` - the program number
`numSteps` - the number of steps to decrement (only for amplitude)

numRepeats - the num repeats (only for amplitude)

Returns:

the resulting TherapyParameter value as a result of the operation. -1 if the operation failed. The resulting amplitude value is calculated based on the last known AmplitudeResolution (V or CC). If this is changed without sending a GetInsInfo command to give the API knowledge of the change, the returned value will be incorrect.

Throws:

InterruptedException - the interrupted exception

---

getNexusStatus

```
public NexusStatus getNexusStatus()
                           throws InterruptedException
```

Gets the status of the Nexus-D system.

Returns:

the Nexus status object or null in the case of an error

Throws:

InterruptedException - the interrupted exception

---

getInsInfo

```
public InsInfo getInsInfo()
                    throws InterruptedException
```

Gets the INS info, including stimulation settings and sensing status.

Returns:

the INS info object or null in the case of an error

Throws:

InterruptedException - the interrupted exception

---

getDataPacket

```
public DataPacket getDataPacket()
                         throws InterruptedException
```

Gets a data packet from the INS. The contents of the data packet are dependent upon the sensing settings configured in the INS. These sensing settings cannot be configured by the Nexus-D system.

Returns:

the data packet or null in the case of an error

Throws:

InterruptedException - the interrupted exception

---

setNexusConfiguration

```
public int setNexusConfiguration(byte maintSessionTimeoutSec,
                                 byte hostSessionTimeoutMin)
                          throws InterruptedException
```

Sets the nexus configuration.

Parameters:
`maintSessionTimeoutSec` - the maintenance session timeout in seconds
`hostSessionTimeoutMin` - the host session timeout minutes
Returns:
the result of the operation (INS response code) or -1 for a timeout
Throws:
`InterruptedException` - the interrupted exception

---

startSensing

```
public int startSensing()
                 throws InterruptedException
```

Enables the sensing feature in the INS.

Returns:
the result of the operation (INS response code) or -1 for a timeout
Throws:
`InterruptedException` - the interrupted exception

---

stopSensing

```
public int stopSensing()
                throws InterruptedException
```

Disables the sensing feature in the INS.

Returns:
the result of the operation (INS response code) or -1 for a timeout
Throws:
`InterruptedException` - the interrupted exception

---

startDataSession

```
public int startDataSession(boolean channelOne)
                    throws InterruptedException
```

Starts a data session with the INS.

Parameters:
`channelOne` - If 422 Hz is the TD sampling frequency, channel 1 will be the channel used based on this boolean
Returns:
the result of the operation (INS response code) or -1 for a timeout
Throws:
`InterruptedException` - the interrupted exception

---

startDataSession

```
public int startDataSession()
                    throws InterruptedException
```

Starts a data session with the INS. If 422 Hz is the TD sampling frequency, channel 1 will be the default channel used

Returns:
the result of the operation (INS response code) or -1 for a timeout
Throws:
`InterruptedException` - the interrupted exception

---

stopDataSession

```
public int stopDataSession()
                   throws InterruptedException
```

Stops a data session.

Returns:
the result of the operation (INS response code) or -1 for a timeout
Throws:
`InterruptedException` - the interrupted exception

---

therapyOn

```
public int therapyOn()
               throws InterruptedException
```

Turns therapy on.

Returns:
the result of the operation (INS response code) or -1 for a timeout
Throws:
`InterruptedException` - the interrupted exception

---

therapyOff

```
public int therapyOff()
                throws InterruptedException
```

Turns therapy off.

Returns:
the result of the operation (INS response code) or -1 for a timeout
Throws:
`InterruptedException` - the interrupted exception

---

setActiveGroup

```
public int setActiveGroup(byte groupNum)
                   throws InterruptedException
```

Sets the active group.

Parameters:
`groupNum` - the group number to activate
Returns:
the result of the operation (INS response code) or -1 for a timeout
Throws:
`InterruptedException` - the interrupted exception

---

restoreClinicianSettings

```
public int restoreClinicianSettings(byte groupNumber)
                             throws InterruptedException
```

Restores the clinician settings saved in the INS.

Parameters:
`groupNumber` - the group number to restore (00 for all)
Returns:
the result of the operation (INS response code) or -1 for a timeout
Throws:
`InterruptedException` - the interrupted exception

---

sendTrigger

```
public int sendTrigger()
                throws InterruptedException
```

Send a trigger to the INS.

Returns:
the result of the operation (INS response code) or -1 for a timeout
Throws:
`InterruptedException` - the interrupted exception

---

cycleStim

```
public int cycleStim()
                 throws InterruptedException
```

Cycle stim on the device (off then back on).

Returns:
the result of the operation (INS response code) or -1 for a timeout
Throws:
`InterruptedException` - the interrupted exception

---

resetCycle

```
public int resetCycle()
                  throws InterruptedException
```

Resets the stimulation cycle.

Returns:
the result of the operation (INS response code) or -1 for a timeout
Throws:
`InterruptedException` - the interrupted exception

---

pulseStimParameter

```
public double pulseStimParameter(byte progNum,
                                 byte numSteps,
                                 byte numDelay)
                          throws InterruptedException
```

Sends a batched inc/dec to allow for fine control of the number of stimulation pulses that are delivered at a given level.

Parameters:
`progNum` - the prog num
`numSteps` - the num steps
`numDelay` - the num delay
Returns:
the new amplitude
Throws:
`InterruptedException` - the interrupted exception

---

computeTemplate

```
public double[][] computeTemplate(DataPacket[] pkts)
```

Computes a template based on passed in data packets that can then be subtracted from other data packets to reduce artifact induced by telemetry. At least 20 packets must be passed in to compute a valid template.

Parameters:
`pkts` - the pkts
Returns:
the double[][] template

Class ObservableConnection

mdt.neuro.nexus

```
java.lang.Object
  └ java.util.Observable
      └ mdt.neuro.nexus.ObservableConnection
```

All Implemented Interfaces:

IDisposable

Direct Known Subclasses:

SerialConnection

```
abstract public class ObservableConnection
extends Observable
implements IDisposable
```

The Class ObservableConnection.

Field Summary		Page
protected ConnectionStatus	**status** The status.	31

Constructor Summary	Page
ObservableConnection()	31

Method Summary		Page
protected abstract int	**connect**() Connect.	31
protected abstract boolean	**disconnect**() Disconnect.	31
void	**dispose**() Disposes of this instance.	32
protected abstract int	**getInputBufferBytesCount**() Gets the input buffer bytes count.	32
ConnectionStatus	**getStatus**() Gets the status of this connection.	31
protected abstract byte[]	**read**() Read from the connection.	32
protected void	**receive**() Receives data from the connection.	32
protected abstract int	**write**(byte[] bufferedData, int count) Writes data to the connection.	32
protected int	**writeBytes**(byte[] bufferedData, int count) Write to the connection.	31

Field Detail

status

```
protected ConnectionStatus status
```

The status.

Constructor Detail

ObservableConnection

```
public ObservableConnection()
```

Method Detail

getStatus

```
public ConnectionStatus getStatus()
```

Gets the status of this connection.

Returns:
the status

---

connect

```
protected abstract int connect()
```

Connect.

Returns:
0, if successful. Error code otherwise

---

disconnect

```
protected abstract boolean disconnect()
```

Disconnect.

Returns:
true, if successful

---

writeBytes

```
protected int writeBytes(byte[] bufferedData,
                         int count)
```

Write to the connection.

Parameters:
`bufferedData` - the buffered data
`count` - the count
Returns:
the int

---

write

```
protected abstract int write(byte[] bufferedData,
                             int count)
```

Writes data to the connection.

Parameters:
`bufferedData` - the buffered data
`count` - the count
Returns:
the int

---

receive

```
protected void receive()
```

Receives data from the connection. Processes the data according to the Nexus-D protocol for forming header and payload

---

read

```
protected abstract byte[] read()
```

Read from the connection.

Returns:
the byte[] bytes that were read

---

getInputBufferBytesCount

```
protected abstract int getInputBufferBytesCount()
```

Gets the input buffer bytes count.

Returns:
the input buffer bytes count

---

dispose

```
public void dispose()
```

Disposes of this instance.

Specified by:

`dispose` in interface `IDisposable`

Class ParameterizedRunnable

mdt.neuro.nexus

java.lang.Object
  └ **mdt.neuro.nexus.ParameterizedRunnable**

All Implemented Interfaces:
Runnable

```
abstract public class ParameterizedRunnable
extends Object
implements Runnable
```

The Class ParameterizedRunnable.

Constructor Summary	Page
ParameterizedRunnable(Object parameter) Instantiates a new parameterized runnable.	34

Method Summary		Page
void	**run**()	34
abstract void	**runWithParameter**(Object param) Executes with the given parameter.	34

Constructor Detail

ParameterizedRunnable

```
public ParameterizedRunnable(Object parameter)
```

Instantiates a new parameterized runnable.

Parameters:
`parameter` - the parameter

Method Detail

runWithParameter

```
public abstract void runWithParameter(Object param)
```

Executes with the given parameter.

Parameters:
`param` - the parameter

---

run

```
public void run()
```

Specified by:

`run` in interface `Runnable`

Enum ResponseCodes

mdt.neuro.nexus

```
java.lang.Object
  └ java.lang.Enum<ResponseCodes>
      └ mdt.neuro.nexus.ResponseCodes
```

All Implemented Interfaces:

Comparable<ResponseCodes>, Serializable

```
public enum ResponseCodes
extends Enum<ResponseCodes>
```

The Enum ResponseCodes.

Enum Constant Summary	Page
BATT_DEPLETED STS2 battery has been depleted.	38
HEADER_CRC_ERROR Header crc error.	37
INS_POR_INDICATED INS POR indicated.	38
INVALID_DATA	38
INVALID_FRAME_ID Invalid frame id.	37
INVALID_FRAME_TYPE Invalid frame type.	37
INVALID_PAYLOAD_LENGTH Invalid payload length.	37
MESSAGE_INCOMPLETE Message Incomplete.	37
NO_RESPONSE No response received from the STS2.	38
NOT_CONNECTED No connection to the STS2.	38
PAYLOAD_CRC_ERROR Payload crc error.	37
PREV_CMD_BUSY Previous command busy.	38
SUCCESS Success.	37

Method Summary		Page
static ResponseCodes	**valueOf**(String name)	38
static ResponseCodes[]	**values**()	38

Enum Constant Detail

SUCCESS

```
public static final ResponseCodes SUCCESS
```

Success.

PAYLOAD_CRC_ERROR

```
public static final ResponseCodes PAYLOAD_CRC_ERROR
```

Payload crc error.

INVALID_FRAME_TYPE

```
public static final ResponseCodes INVALID_FRAME_TYPE
```

Invalid frame type.

MESSAGE_INCOMPLETE

```
public static final ResponseCodes MESSAGE_INCOMPLETE
```

Message Incomplete.

INVALID_FRAME_ID

```
public static final ResponseCodes INVALID_FRAME_ID
```

Invalid frame id.

INVALID_PAYLOAD_LENGTH

```
public static final ResponseCodes INVALID_PAYLOAD_LENGTH
```

Invalid payload length.

HEADER_CRC_ERROR

```
public static final ResponseCodes HEADER_CRC_ERROR
```

Header crc error.

PREV_CMD_BUSY

```
public static final ResponseCodes PREV_CMD_BUSY
```

Previous command busy.

INS_POR_INDICATED

```
public static final ResponseCodes INS_POR_INDICATED
```

INS POR indicated.

BATT_DEPLETED

```
public static final ResponseCodes BATT_DEPLETED
```

STS2 battery has been depleted.

INVALID_DATA

```
public static final ResponseCodes INVALID_DATA
```

NO_RESPONSE

```
public static final ResponseCodes NO_RESPONSE
```

No response received from the STS2.

NOT_CONNECTED

```
public static final ResponseCodes NOT_CONNECTED
```

No connection to the STS2.

Method Detail

values

```
public static ResponseCodes[] values()
```

---

valueOf

```
public static ResponseCodes valueOf(String name)
```

Class SerialConnection

mdt.neuro.nexus

```
java.lang.Object
  └ java.util.Observable
      └ mdt.neuro.nexus.ObservableConnection
          └ mdt.neuro.nexus.SerialConnection
```

All Implemented Interfaces:

IDisposable

```
public class SerialConnection
extends ObservableConnection
```

The Class Connection.

Fields inherited from class mdt.neuro.nexus.ObservableConnection
status

Constructor Summary	Page
SerialConnection(String portName) Instantiates a new connection.	39

Method Summary		Page
protected int	**connect**() Connects this serial instance to the given port.	40
protected boolean	**disconnect**() Disconnects this instance.	40
protected int	**getInputBufferBytesCount**() Gets the input buffer bytes count.	41
protected byte[]	**read**() Read from the connection.	41
void	**serialEvent**(SerialPortEvent event)	41
boolean	**setPortName**(String portName) Sets the port name and initializes the port if it does not exist.	40
protected int	**write**(byte[] toSend, int count) Writes data to the connection.	40

Methods inherited from class mdt.neuro.nexus.ObservableConnection
dispose, getStatus, receive, writeBytes

Constructor Detail

SerialConnection

```
public SerialConnection(String portName)
```

Instantiates a new connection.

Parameters:
`portName` - the port name

Method Detail

setPortName

```
public boolean setPortName(String portName)
                    throws SerialPortException
```

Sets the port name and initializes the port if it does not exist. If a port is already open, it will be closed and a new one will be initialized using the new name

Parameters:
`portName` - the port name
Returns:
true, if successful
Throws:
`SerialPortException` - the serial port exception

---

connect

```
protected int connect()
```

Connects this serial instance to the given port.

Overrides:
`connect` in class `ObservableConnection`
Returns:
true, if successful

---

disconnect

```
protected boolean disconnect()
```

Disconnects this instance.

Overrides:
`disconnect` in class `ObservableConnection`
Returns:
true, if successful

---

write

```
protected int write(byte[] toSend,
                    int count)
```

Writes data to the connection.

Overrides:
`write` in class `ObservableConnection`
Parameters:
`count` - the count

Returns:
the int

---

serialEvent

```
public void serialEvent(SerialPortEvent event)
```

---

read

```
protected byte[] read()
```

Read from the connection.

Overrides:
read in class ObservableConnection
Returns:
the byte[] bytes that were read

---

getInputBufferBytesCount

```
protected int getInputBufferBytesCount()
```

Gets the input buffer bytes count.

Overrides:
getInputBufferBytesCount in class ObservableConnection
Returns:
the input buffer bytes count

Class SerialPortErrors

mdt.neuro.nexus

```
java.lang.Object
  └ mdt.neuro.nexus.SerialPortErrors
```

```
public class SerialPortErrors
extends Object
```

The Class SerialPortErrors.

Field Summary		Page
static int	**PORT_BUSY** The port was busy.	42
static int	**PORT_NOT_FOUND** The port was not found.	42
static int	**PORT_NULL** The port was not defined.	42
static int	**UNKNOWN** The unknown.	42

Field Detail

PORT_NOT_FOUND

```
public static final int PORT_NOT_FOUND
```

The port was not found.

PORT_BUSY

```
public static final int PORT_BUSY
```

The port was busy.

PORT_NULL

```
public static final int PORT_NULL
```

The port was not defined.

UNKNOWN

```
public static final int UNKNOWN
```

The unknown.

Class ThreadedNexusInstrument

mdt.neuro.nexus

java.lang.Object
  └ **mdt.neuro.nexus.ThreadedNexusInstrument**

---

public class **ThreadedNexusInstrument**
extends Object

The Class ThreadedNexusInstrument.

Constructor Summary	Page
ThreadedNexusInstrument()	44

Method Summary		Page
void	**arbitraryCommand**(List<Byte> bytesToSend, int timeoutMs) Sends an command of arbitrary bytes.	48
void	**cycleStim**() Cycles the stimulation.	48
void	**decStimParameter**(TherapyParameter type, byte progNum) Decrement the given stimulation parameter.	45
void	**decStimParameter**(TherapyParameter type, byte progNum, byte numSteps, byte numRepeats) Decrement the given stimulation parameter.	46
void	**getDataPacket**() Gets a data packet from the INS.	46
void	**getInsInfo**() Gets the INS info, including stimulation settings and sensing status.	46
int	**getLastInsResponseCode**() Gets the last ins response code.	45
ResponseCodes	**getLastNexusResponseCode**() Gets the last nexus response code.	44
void	**getNexusStatus**() Gets the status of the Nexus-D system.	46
Object	**getThreadSafeReturnVal**() Gets the thread safe return val.	44
void	**incStimParameter**(TherapyParameter type, byte progNum) Increment the given stimulation parameter.	45
void	**incStimParameter**(TherapyParameter type, byte progNum, byte numSteps, byte numRepeats) Increment the given stimulation parameter.	45
void	**pulseStimParameter**(byte progNum, byte numSteps, byte numDelay) Sends a batched inc/dec to allow for fine control of the number of stimulation pulses that are delivered at a given level.	49
void	**resetCycle**() Reset the stimulation cycle.	49
void	**restoreClinicianSettings**(byte group) Restores the clinician settings saved in the INS.	48

void	**sendTrigger**() Send a trigger to the INS.	*48*
void	**setActiveGroup**(byte group) Sets the active group.	*48*
void	**setNexusConfiguration**(byte maintSessionTimeoutSec, byte hostSessionTimeoutMin) Sets the nexus configuration.	*46*
void	**startDataSession**() Starts a data session with the INS.	*47*
void	**startDataSession**(boolean channelOne) Starts a data session with the INS.	*47*
void	**startSensing**() Enables the sensing feature in the INS.	*47*
void	**stopDataSession**() Stops a data session.	*47*
void	**stopSensing**() Disables the sensing feature in the INS.	*47*
void	**therapyOff**() Turns therapy off.	*48*
void	**therapyOn**() Turns therapy on.	*47*

Constructor Detail

ThreadedNexusInstrument

```
public ThreadedNexusInstrument()
```

Method Detail

getThreadSafeReturnVal

```
public Object getThreadSafeReturnVal()
```

Gets the thread safe return val.

Returns:
the thread safe return val

---

getLastNexusResponseCode

```
public ResponseCodes getLastNexusResponseCode()
```

Gets the last nexus response code.

Returns:
the nexusResponseCode

---

getLastInsResponseCode

```
public int getLastInsResponseCode()
```

Gets the last ins response code.

Returns:
the insResponseCode

---

incStimParameter

```
public void incStimParameter(TherapyParameter type,
                             byte progNum)
```

Increment the given stimulation parameter. Sets the returned value to the resulting TherapyParameter value as a result of the operation. -1 if the operation failed. The resulting amplitude value is calculated based on the last known AmplitudeResolution (V or CC). If this is changed without sending a GetInsInfo command to give the API knowledge of the change, the returned value will be incorrect

Parameters:
type - the type
progNum - the prog num

---

incStimParameter

```
public void incStimParameter(TherapyParameter type,
                             byte progNum,
                             byte numSteps,
                             byte numRepeats)
```

Increment the given stimulation parameter. Sets the returned value to the resulting TherapyParameter value as a result of the operation. -1 if the operation failed. The resulting amplitude value is calculated based on the last known AmplitudeResolution (V or CC). If this is changed without sending a GetInsInfo command to give the API knowledge of the change, the returned value will be incorrect

Parameters:
type - the type
progNum - the prog num
numSteps - the num steps
numRepeats - the num repeats

---

decStimParameter

```
public void decStimParameter(TherapyParameter type,
                             byte progNum)
```

Decrement the given stimulation parameter. Sets the returned value to the resulting TherapyParameter value as a result of the operation. -1 if the operation failed. The resulting amplitude value is calculated based on the last known AmplitudeResolution (V or CC). If this is changed without sending a GetInsInfo command to give the API knowledge of the change, the returned value will be incorrect

Parameters:
type - the type
progNum - the prog num

decStimParameter

```
public void decStimParameter(TherapyParameter type,
                             byte progNum,
                             byte numSteps,
                             byte numRepeats)
```

Decrement the given stimulation parameter. Sets the returned value to the resulting `TherapyParameter` value as a result of the operation. -1 if the operation failed. The resulting amplitude value is calculated based on the last known `AmplitudeResolution` (V or CC). If this is changed without sending a `GetInsInfo` command to give the API knowledge of the change, the returned value will be incorrect

Parameters:

`type` - the type
`progNum` - the prog num
`numSteps` - the num steps
`numRepeats` - the num repeats

getNexusStatus

```
public void getNexusStatus()
```

Gets the status of the Nexus-D system. Sets the returned value to the nexus status object or null in the case of an error

getInsInfo

```
public void getInsInfo()
```

Gets the INS info, including stimulation settings and sensing status. Sets the returned value to the ins info object or null in the case of an error

getDataPacket

```
public void getDataPacket()
```

Gets a data packet from the INS. The contents of the data packet are dependent upon the sensing settings configured in the INS. These sensing settings cannot be configured by the Nexus-D system. Sets the returned value to the data packet or null in the case of an error

setNexusConfiguration

```
public void setNexusConfiguration(byte maintSessionTimeoutSec,
                                  byte hostSessionTimeoutMin)
```

Sets the nexus configuration. Sets the returned value to the result of the operation (INS response code) or -1 for a timeout

Parameters:

`maintSessionTimeoutSec` - the maint session timeout sec
`hostSessionTimeoutMin` - the host session timeout min

startSensing

```
public void startSensing()
```

Enables the sensing feature in the INS. Sets the returned value to the result of the operation (INS response code) or -1 for a timeout

stopSensing

```
public void stopSensing()
```

Disables the sensing feature in the INS. Sets the returned value to the result of the operation (INS response code) or -1 for a timeout

startDataSession

```
public void startDataSession()
```

Starts a data session with the INS. If 422 Hz is the TD sampling frequency, channel 1 will be the default channel used. Sets the returned value to the result of the operation (INS response code) or -1 for a timeout

startDataSession

```
public void startDataSession(boolean channelOne)
```

Starts a data session with the INS. Sets the returned value to the result of the operation (INS response code) or -1 for a timeout

Parameters:

`channelOne` - If 422 Hz is the TD sampling frequency, channel 1 will be the channel used based on this boolean

stopDataSession

```
public void stopDataSession()
```

Stops a data session. Sets the returned value to the result of the operation (INS response code) or -1 for a timeout

therapyOn

```
public void therapyOn()
```

Turns therapy on. Sets the returned value to the result of the operation (INS response code) or -1 for a timeout

therapyOff

```
public void therapyOff()
```

Turns therapy off. Sets the returned value to the result of the operation (INS response code) or -1 for a timeout

---

setActiveGroup

```
public void setActiveGroup(byte group)
```

Sets the active group. Sets the returned value to the result of the operation (INS response code) or -1 for a timeout

Parameters:

`group` - the new active group

---

restoreClinicianSettings

```
public void restoreClinicianSettings(byte group)
```

Restores the clinician settings saved in the INS. Sets the returned value to the result of the operation (INS response code) or -1 for a timeout

Parameters:

`group` - the group

---

sendTrigger

```
public void sendTrigger()
```

Send a trigger to the INS. Sets the returned value to the result of the operation (INS response code) or -1 for a timeout

---

cycleStim

```
public void cycleStim()
```

Cycles the stimulation. Sets the returned value to the result of the operation (INS response code) or -1 for a timeout

---

arbitraryCommand

```
public void arbitraryCommand(List<Byte> bytesToSend,
                             int timeoutMs)
```

Sends an command of arbitrary bytes. Sets the returned value to the returned bytes from the operation

Parameters:

`bytesToSend` - the bytes to send

`timeoutMs` - the timeout ms

---

resetCycle

```
public void resetCycle()
```

Reset the stimulation cycle. Sets the returned value to the result of the operation (INS response code) or -1 for a timeout

---

pulseStimParameter

```
public void pulseStimParameter(byte progNum,
                               byte numSteps,
                               byte numDelay)
```

Sends a batched inc/dec to allow for fine control of the number of stimulation pulses that are delivered at a given level.

Parameters:

`progNum` - the prog num
`numSteps` - the num steps
`numDelay` - the num delay

Class ThreadManager

mdt.neuro.nexus

java.lang.Object
  └ **mdt.neuro.nexus.ThreadManager**

All Implemented Interfaces:
Observer

---

public class **ThreadManager**
extends Object
implements Observer

The Class ThreadManager. This class is a singleton class that allows a Runnable to be executed on a worker thread. It supports the following functionality: Monitor its execution and abort an operation in progress. This method interrupts the Runnable that is being executed. The Runnable should handle the InterruptedException that will be thrown gracefully to ensure consistent operation. The ThreadManager only allows one operation to be executed at a time, and does not support the queuing of operations.

Method Summary		Page
void	**abortOperation**() Abort operation.	51
boolean	**doOperation**(Runnable toExecute) Does operation specified by the Runnable on the worker thread if it is available.	50
static ThreadManager	**getInstance**() Gets the single instance of ThreadManager.	50
boolean	**isExecuting**()	50
void	**update**(Observable o, Object arg)	51

Method Detail

isExecuting

public boolean **isExecuting**()

---

getInstance

public static ThreadManager **getInstance**()

Gets the single instance of ThreadManager.

Returns:
single instance of ThreadManager

---

doOperation

public boolean **doOperation**(Runnable toExecute)

Does operation specified by the Runnable on the worker thread if it is available.

Parameters:

`toExecute` - the Runnable to execute

Returns:

true, if the operation was started on the worker thread

---

abortOperation

```
public void abortOperation()
```

Abort operation. This will interrupt the worker thread. The implemented runnable must handle a potential `InterruptedException` gracefully

---

update

```
public synchronized void update(Observable o,
                                Object arg)
```

Specified by:

`update` in interface `Observer`

Class Util

mdt.neuro.nexus

```
java.lang.Object
  └ mdt.neuro.nexus.Util
```

```
public class Util
extends Object
```

The Class Util.

Constructor Summary	Page
Util()	52

Method Summary		Page
static String	**toHex**(byte[] array) Converts a byte array to a hex string.	52
static String	**toHex**(Byte[] array) Converts a byte array to a hex string.	53
static Byte[]	**toObject**(byte[] array) Converts a byte array to a Byte array.	52

Constructor Detail

Util

```
public Util()
```

Method Detail

toObject

```
public static Byte[] toObject(byte[] array)
```

Converts a byte array to a Byte array.

Parameters:
`array` - the array
Returns:
the byte[]

---

toHex

```
public static String toHex(byte[] array)
```

Converts a byte array to a hex string.

Parameters:
`array` - the array

Returns:
the string

---

toHex

```
public static String toHex(Byte[] array)
```

Converts a byte array to a hex string.

Parameters:
`array` - the array
Returns:
the string

Package mdt.neuro.nexus.commands

Class Summary		Page
CommandCodes	The Class CommandCodes.	54
CommandResponseCodes	The Class CommandResponseCodes.	60
CycleStim	The Class CycleStim.	69
DataStructureAccessor	The Class DataStructureAccessor.	70
DecAmplitude	The Class DecAmplitude.	72
DecFrequency	The Class DecFrequency.	74
DecPulseWidth	The Class DecPulseWidth.	76
GetInsInfo	The Class GetInsInfo.	78
GetNexusStatus	The Class GetNexusStatus.	80
GetRealtimeData	The Class GetRealtimeData.	82
IncAmplitude	The Class IncAmplitude.	84
IncFrequency	The Class IncFrequency.	86
IncPulseWidth	The Class IncPulseWidth.	88
PulseStim	The Class PulseStim.	90
ResetCycle		92
RestoreClinicianSettings	The Class RestoreClinicianSettings.	93
SendTrigger	The Class SendTrigger.	94
SetActiveGroup	The Class SetActiveGroup.	95
SetNexusConfiguration	The Class SetNexusConfiguration.	96
StartRealTime	The Class StartRealTime.	97
StartSensing	The Class StartSensing.	98
StopRealTime	The Class StopRealTime.	99
StopSensing	The Class StopSensing.	100
TherapyOff	The Class TherapyOff.	101
TherapyOn	The Class TherapyOn.	102

Class CommandCodes

mdt.neuro.nexus.commands

```
java.lang.Object
  └ mdt.neuro.nexus.commands.CommandCodes
```

```
public class CommandCodes
extends Object
```

The Class CommandCodes.

Field Summary		Page
static short	**CYCLE_STIM** The cycle stim.	57
static short	**DEC_AMP** The dec amp.	58
static short	**DEC_FREQ** The dec freq.	58
static short	**DEC_PULSEWIDTH** The dec pulsewidth.	58
static short	**GET_INS_INFO** The get ins info.	56
static short	**GET_REALTIME_DATA** The get realtime data.	57
static short	**GET_STATUS** The get status.	57
static short	**INC_AMP** The inc amp.	58
static short	**INC_FREQ** The inc freq.	58
static short	**INC_PULSEWIDTH** The inc pulsewidth.	58
static short	**PULSE_STIM** The pulse stim command can be used to control delivery of a small number of pulses.	57
static short	**RESET_CYCLE** Reset Cycle command can be used to turn stim on then off	57
static short	**RESTORE_CLINICIAN_SETTINGS** The restore clinician settings.	58
static short	**SEND_TRIGGER** The send trigger.	59
static short	**SET_ACTIVE_GROUP** The set active group.	57
static short	**SET_NEXUS_CONFIGURATION** The set nexus configuration.	56
static short	**START_REALTIME** The start realtime.	56
static short	**START_SENSING** The start sensing.	56

static short	**STOP_REALTIME** The stop realtime.	56
static short	**STOP_SENSING** The stop sensing.	56
static short	**THERAPY_OFF** The therapy off.	57
static short	**THERAPY_ON** The therapy on.	57

Field Detail

GET_INS_INFO

```
public static final short GET_INS_INFO
```

The get ins info.

SET_NEXUS_CONFIGURATION

```
public static final short SET_NEXUS_CONFIGURATION
```

The set nexus configuration.

START_SENSING

```
public static final short START_SENSING
```

The start sensing.

STOP_SENSING

```
public static final short STOP_SENSING
```

The stop sensing.

START_REALTIME

```
public static final short START_REALTIME
```

The start realtime.

STOP_REALTIME

```
public static final short STOP_REALTIME
```

The stop realtime.

GET_STATUS

```
public static final short GET_STATUS
```

The get status.

GET_REALTIME_DATA

```
public static final short GET_REALTIME_DATA
```

The get realtime data.

CYCLE_STIM

```
public static final short CYCLE_STIM
```

The cycle stim.

RESET_CYCLE

```
public static final short RESET_CYCLE
```

Reset Cycle command can be used to turn stim on then off

PULSE_STIM

```
public static final short PULSE_STIM
```

The pulse stim command can be used to control delivery of a small number of pulses.

THERAPY_ON

```
public static final short THERAPY_ON
```

The therapy on.

THERAPY_OFF

```
public static final short THERAPY_OFF
```

The therapy off.

SET_ACTIVE_GROUP

```
public static final short SET_ACTIVE_GROUP
```

The set active group.

INC_AMP

```
public static final short INC_AMP
```

The inc amp.

DEC_AMP

```
public static final short DEC_AMP
```

The dec amp.

INC_PULSEWIDTH

```
public static final short INC_PULSEWIDTH
```

The inc pulsewidth.

DEC_PULSEWIDTH

```
public static final short DEC_PULSEWIDTH
```

The dec pulsewidth.

INC_FREQ

```
public static final short INC_FREQ
```

The inc freq.

DEC_FREQ

```
public static final short DEC_FREQ
```

The dec freq.

RESTORE_CLINICIAN_SETTINGS

```
public static final short RESTORE_CLINICIAN_SETTINGS
```

The restore clinician settings.

SEND_TRIGGER

```
public static final short SEND_TRIGGER
```

The send trigger.

Class CommandResponseCodes

mdt.neuro.nexus.commands

```
java.lang.Object
  └ mdt.neuro.nexus.commands.CommandResponseCodes
```

```
public class CommandResponseCodes
extends Object
```

The Class CommandResponseCodes.

Field Summary		Page
static short	**AMP_AT_LOWER_LIM** The amp at lower lim.	63
static short	**AMP_AT_UPPER_LIM** The amp at upper lim.	63
static short	**CHANGE_NOT_DONE** The change not done.	64
static short	**COMMAND_NOT_VALID_FOR_DEVICE** The command is not valid for the device type (i.e.	66
static short	**DEVICE_RESET** The device reset.	64
static short	**FREQ_AT_LOWER_LIM** The freq at lower lim.	64
static short	**FREQ_AT_UPPER_LIM** The freq at upper lim.	64
static short	**GROUP_INVALID** The group invalid.	62
static short	**GROUP_OUT_OF_RANGE** The group out of range.	62
static short	**INS_FILTERING_ON** INS Filtering On (i.e.	66
static short	**INS_POR** The ins por.	66
static short	**INVALID_AMP** The invalid amp.	64
static short	**INVALID_CMD** The invalid cmd.	65
static short	**INVALID_CMD_DATA** The invalid cmd data.	65
static short	**INVALID_PROG_IN_SET** The invalid program in the active set.	62
static short	**INVALID_REALTIME_RESPONSE** The invalid realtime response.	66
static short	**NO_ACTIVE_SET_DEFINED** The no active set defined.	63
static short	**NO_APP_BACKUP** The no app backup.	63

static short	**NO_TELEM_RESP** The no telem resp.	65
static short	**NOT_ACTIVE_GROUP** The not active group.	65
static short	**NOT_ALL_CMDS_EXECUTED** The not all cmds executed.	66
static short	**NOT_READY** The not ready.	65
static short	**NULL_GROUP_ACTIVE** The null group active.	65
static short	**NULL_PACKETS** The null packets.	66
static short	**PIC_BUSY** The pic busy.	67
static short	**PIC_IF_BUSY** The pic if busy.	67
static short	**PIC_INVALID_ADDR** The pic invalid addr.	68
static short	**PIC_INVALID_CMD** The pic invalid cmd.	67
static short	**PIC_INVALID_CMD_TYPE** The pic invalid cmd type.	67
static short	**PIC_INVALID_LEN** The pic invalid len.	67
static short	**PIC_INVALID_PARAM** The pic invalid param.	67
static short	**PIC_NO_RESP** The pic no resp.	67
static short	**PIC_READ_BACK_FAIL** The pic read back fail.	67
static short	**PIC_TEST_IN_PROG** The pic test in prog.	68
static short	**PROG_NOT_IN_GROUP** The prog not in group.	63
static short	**PROG_OUT_OF_RANGE** The prog out of range.	62
static short	**PROGRAM_INVALID** The program invalid.	62
static short	**PW_AT_LOWER_LIM** The pw at lower lim.	63
static short	**PW_AT_UPPER_LIM** The pw at upper lim.	63
static short	**REALTIME_NOT_ACTIVE** The realtime not active.	65
static short	**SENSE_CONFIG_ERROR** The sense config error.	65
static short	**SET_TRANSITION_IN_PROG** The set transition in prog.	64

static short	**SUCCESS** The success.	62
static short	**TELEM_ERROR** The telem error.	66
static short	**THERAPY_NOT_ON** The therapy not on.	63
static short	**THERAPY_OOR** The therapy oor.	64

Field Detail

SUCCESS

```
public static final short SUCCESS
```

The success. (0)

PROG_OUT_OF_RANGE

```
public static final short PROG_OUT_OF_RANGE
```

The prog out of range. (10)

GROUP_OUT_OF_RANGE

```
public static final short GROUP_OUT_OF_RANGE
```

The group out of range. (11)

INVALID_PROG_IN_SET

```
public static final short INVALID_PROG_IN_SET
```

The invalid program in the active set. (12)

PROGRAM_INVALID

```
public static final short PROGRAM_INVALID
```

The program invalid. (13)

GROUP_INVALID

```
public static final short GROUP_INVALID
```

The group invalid. (14)

THERAPY_NOT_ON

```
public static final short THERAPY_NOT_ON
```

The therapy not on. (15)

NO_APP_BACKUP

```
public static final short NO_APP_BACKUP
```

The no app backup. (18)

PROG_NOT_IN_GROUP

```
public static final short PROG_NOT_IN_GROUP
```

The prog not in group. (20)

NO_ACTIVE_SET_DEFINED

```
public static final short NO_ACTIVE_SET_DEFINED
```

The no active set defined. (21)

AMP_AT_UPPER_LIM

```
public static final short AMP_AT_UPPER_LIM
```

The amp at upper lim. (22)

AMP_AT_LOWER_LIM

```
public static final short AMP_AT_LOWER_LIM
```

The amp at lower lim. (23)

PW_AT_UPPER_LIM

```
public static final short PW_AT_UPPER_LIM
```

The pw at upper lim. (24)

PW_AT_LOWER_LIM

```
public static final short PW_AT_LOWER_LIM
```

The pw at lower lim. (25)

FREQ_AT_UPPER_LIM

```
public static final short FREQ_AT_UPPER_LIM
```

The freq at upper lim. (26)

FREQ_AT_LOWER_LIM

```
public static final short FREQ_AT_LOWER_LIM
```

The freq at lower lim. (27)

THERAPY_OOR

```
public static final short THERAPY_OOR
```

The therapy oor. (29)

CHANGE_NOT_DONE

```
public static final short CHANGE_NOT_DONE
```

The change not done. (30)

DEVICE_RESET

```
public static final short DEVICE_RESET
```

The device reset. (32)

SET_TRANSITION_IN_PROG

```
public static final short SET_TRANSITION_IN_PROG
```

The set transition in prog. (50)

INVALID_AMP

```
public static final short INVALID_AMP
```

The invalid amp. (51)

NOT_ACTIVE_GROUP

```
public static final short NOT_ACTIVE_GROUP
```

The not active group. (54)

NULL_GROUP_ACTIVE

```
public static final short NULL_GROUP_ACTIVE
```

The null group active. (59)

NOT_READY

```
public static final short NOT_READY
```

The not ready. (101)

INVALID_CMD

```
public static final short INVALID_CMD
```

The invalid cmd. (102)

INVALID_CMD_DATA

```
public static final short INVALID_CMD_DATA
```

The invalid cmd data. (103)

REALTIME_NOT_ACTIVE

```
public static final short REALTIME_NOT_ACTIVE
```

The realtime not active. (105)

SENSE_CONFIG_ERROR

```
public static final short SENSE_CONFIG_ERROR
```

The sense config error. (106)

NO_TELEM_RESP

```
public static final short NO_TELEM_RESP
```

The no telem resp. (107)

TELEM_ERROR

```
public static final short TELEM_ERROR
```

The telem error. (108)

INS_FILTERING_ON

```
public static final short INS_FILTERING_ON
```

INS Filtering On (i.e. real-time will not start if filtering is enabled in the Activa PC+S) (109)

INS_POR

```
public static final short INS_POR
```

The ins por. (110)

INVALID_REALTIME_RESPONSE

```
public static final short INVALID_REALTIME_RESPONSE
```

The invalid realtime response. (111)

NULL_PACKETS

```
public static final short NULL_PACKETS
```

The null packets. (112)

NOT_ALL_CMDS_EXECUTED

```
public static final short NOT_ALL_CMDS_EXECUTED
```

The not all cmds executed. (114)

COMMAND_NOT_VALID_FOR_DEVICE

```
public static final short COMMAND_NOT_VALID_FOR_DEVICE
```

The command is not valid for the device type (i.e. sensing command for PC device). (115)

PIC_IF_BUSY

```
public static final short PIC_IF_BUSY
```

The pic if busy. (232)

PIC_INVALID_LEN

```
public static final short PIC_INVALID_LEN
```

The pic invalid len. (233)

PIC_NO_RESP

```
public static final short PIC_NO_RESP
```

The pic no resp. (234)

PIC_READ_BACK_FAIL

```
public static final short PIC_READ_BACK_FAIL
```

The pic read back fail. (235)

PIC_BUSY

```
public static final short PIC_BUSY
```

The pic busy. (241)

PIC_INVALID_CMD_TYPE

```
public static final short PIC_INVALID_CMD_TYPE
```

The pic invalid cmd type. (242)

PIC_INVALID_PARAM

```
public static final short PIC_INVALID_PARAM
```

The pic invalid param. (243)

PIC_INVALID_CMD

```
public static final short PIC_INVALID_CMD
```

The pic invalid cmd. (244)

PIC_INVALID_ADDR

```
public static final short PIC_INVALID_ADDR
```

The pic invalid addr. (245)

PIC_TEST_IN_PROG

```
public static final short PIC_TEST_IN_PROG
```

The pic test in prog. (246)

Class CycleStim

mdt.neuro.nexus.commands

java.lang.Object
  └ mdt.neuro.nexus.ApplicationCommand
      └ **mdt.neuro.nexus.commands.CycleStim**

---

public class **CycleStim**
extends ApplicationCommand

The Class CycleStim.

Field Summary		Page
static int	**TIMEOUT** The timeout value (ms) for this command.	69

Fields inherited from class mdt.neuro.nexus.ApplicationCommand
commandBytes, commandCode, crcValid, request, response, responseCode, responseData, timeout

Methods inherited from class mdt.neuro.nexus.ApplicationCommand
consumeBytes, getRequestData, getResponseBytes, getResponseData

Field Detail

TIMEOUT

public static final int **TIMEOUT**

The timeout value (ms) for this command.

Class DataStructureAccessor

mdt.neuro.nexus.commands

```
java.lang.Object
  └ mdt.neuro.nexus.commands.DataStructureAccessor
```

```
abstract public class DataStructureAccessor
extends Object
```

The Class DataStructureAccessor. This class allows friend access to the creation of data structures by internal API classes not in the data package.

Constructor Summary	Page
DataStructureAccessor()	70

Method Summary		Page
protected abstract DataPacket	**newDataPacket**(List<Byte> bytes) Provides a New data packet.	71
protected abstract InsInfo	**newInsInfo**(List<Byte> bytes) Provides a New ins info.	70
protected abstract NexusStatus	**newNexusStatus**(List<Byte> bytes) Provides a New nexus status.	71
static void	**setProvider**(DataStructureAccessor provider) Sets the provider.	70

Constructor Detail

DataStructureAccessor

```
public DataStructureAccessor()
```

Method Detail

setProvider

```
public static void setProvider(DataStructureAccessor provider)
```

Sets the provider.

Parameters:

`provider` - the new provider

---

newInsInfo

```
protected abstract InsInfo newInsInfo(List<Byte> bytes)
```

Provides a New ins info.

Parameters:
`bytes` - the bytes
Returns:
the ins info

---

newNexusStatus

```
protected abstract NexusStatus newNexusStatus(List<Byte> bytes)
```

Provides a New nexus status.

Parameters:
`bytes` - the bytes
Returns:
the nexus status

---

newDataPacket

```
protected abstract DataPacket newDataPacket(List<Byte> bytes)
```

Provides a New data packet.

Parameters:
`bytes` - the bytes
Returns:
the data packet

Class DecAmplitude

mdt.neuro.nexus.commands

```
java.lang.Object
  └ mdt.neuro.nexus.ApplicationCommand
      └ mdt.neuro.nexus.commands.DecAmplitude
```

```
public class DecAmplitude
extends ApplicationCommand
```

The Class DecAmplitude.

Field Summary		Page
static int	**RESPONSE_SIZE** The response size.	72
static int	**TIMEOUT**	72

Fields inherited from class mdt.neuro.nexus.ApplicationCommand
commandBytes, commandCode, crcValid, request, response, responseCode, responseData, timeout

Method Summary		Page
protected void	**consumeBytes**(List<Byte> bytes)	73
double	**getNewAmplitude**() Gets the new amplitude.	73
byte	**getNumExecuted**() Gets the number of DEC commands executed – the actual number executed may not match the number requested.	73

Methods inherited from class mdt.neuro.nexus.ApplicationCommand
getRequestData, getResponseBytes, getResponseData

Field Detail

RESPONSE_SIZE

```
public static final int RESPONSE_SIZE
```

The response size.

TIMEOUT

```
public static final int TIMEOUT
```

Method Detail

getNewAmplitude

```
public double getNewAmplitude()
```

Gets the new amplitude.

Returns:
the newAmplitude

---

getNumExecuted

```
public byte getNumExecuted()
```

Gets the number of DEC commands executed – the actual number executed may not match the number requested.

Returns:
the number of DEC commands executed

---

consumeBytes

```
protected void consumeBytes(List<Byte> bytes)
```

Overrides:
consumeBytes in class ApplicationCommand

Class DecFrequency

mdt.neuro.nexus.commands

```
java.lang.Object
  └ mdt.neuro.nexus.ApplicationCommand
      └ mdt.neuro.nexus.commands.DecFrequency
```

```
public class DecFrequency
extends ApplicationCommand
```

The Class DecFrequency.

Field Summary		Page
static int	**RESPONSE_SIZE** The response size.	74
static int	**TIMEOUT** The timeout value (ms) for this command.	74

Fields inherited from class mdt.neuro.nexus.ApplicationCommand
commandBytes, commandCode, crcValid, request, response, responseCode, responseData, timeout

Method Summary		Page
protected void	**consumeBytes**(List<Byte> bytes)	75
int	**getNewFrequency**() Gets the new frequency.	75

Methods inherited from class mdt.neuro.nexus.ApplicationCommand
getRequestData, getResponseBytes, getResponseData

Field Detail

RESPONSE_SIZE

```
public static final int RESPONSE_SIZE
```

The response size.

TIMEOUT

```
public static final int TIMEOUT
```

The timeout value (ms) for this command.

Method Detail

getNewFrequency

```
public int getNewFrequency()
```

Gets the new frequency.

Returns:
the newFrequency

---

consumeBytes

```
protected void consumeBytes(List<Byte> bytes)
```

Overrides:
consumeBytes in class ApplicationCommand

Class DecPulseWidth

mdt.neuro.nexus.commands

```
java.lang.Object
  └ mdt.neuro.nexus.ApplicationCommand
      └ mdt.neuro.nexus.commands.DecPulseWidth
```

```
public class DecPulseWidth
extends ApplicationCommand
```

The Class DecPulseWidth.

Field Summary		Page
static int	**RESPONSE_SIZE** The response size.	76
static int	**TIMEOUT** The timeout value (ms) for this command.	76

Fields inherited from class mdt.neuro.nexus.ApplicationCommand
commandBytes, commandCode, crcValid, request, response, responseCode, responseData, timeout

Method Summary		Page
protected void	**consumeBytes**(List<Byte> bytes)	77
short	**getNewPulseWidth**() Gets the new pulse width.	77

Methods inherited from class mdt.neuro.nexus.ApplicationCommand
getRequestData, getResponseBytes, getResponseData

Field Detail

RESPONSE_SIZE

```
public static final int RESPONSE_SIZE
```

The response size.

TIMEOUT

```
public static final int TIMEOUT
```

The timeout value (ms) for this command.

Method Detail

getNewPulseWidth

```
public short getNewPulseWidth()
```

Gets the new pulse width.

Returns:
the newPulseWidth

---

consumeBytes

```
protected void consumeBytes(List<Byte> bytes)
```

Overrides:
consumeBytes in class ApplicationCommand

Class GetInsInfo

mdt.neuro.nexus.commands

```
java.lang.Object
  └ mdt.neuro.nexus.ApplicationCommand
      └ mdt.neuro.nexus.commands.GetInsInfo
```

```
public class GetInsInfo
extends ApplicationCommand
```

The Class GetInsInfo.

Field Summary		Page
static int	**RESPONSE_SIZE** The response size.	78
static int	**TIMEOUT** The timeout value (ms) for this command.	78

Fields inherited from class mdt.neuro.nexus.ApplicationCommand
commandBytes, commandCode, crcValid, request, response, responseCode, responseData, timeout

Method Summary		Page
protected void	**consumeBytes**(List<Byte> bytes)	79
InsInfo	**getInfo**() Gets the info.	79

Methods inherited from class mdt.neuro.nexus.ApplicationCommand
getRequestData, getResponseBytes, getResponseData

Field Detail

RESPONSE_SIZE

```
public static final int RESPONSE_SIZE
```

The response size.

TIMEOUT

```
public static final int TIMEOUT
```

The timeout value (ms) for this command.

Method Detail

getInfo

```
public InsInfo getInfo()
```

Gets the info.

Returns:
the info

---

consumeBytes

```
protected void consumeBytes(List<Byte> bytes)
```

Overrides:
consumeBytes in class ApplicationCommand

Class GetNexusStatus

mdt.neuro.nexus.commands

```
java.lang.Object
  └ mdt.neuro.nexus.ApplicationCommand
      └ mdt.neuro.nexus.commands.GetNexusStatus
```

```
public class GetNexusStatus
extends ApplicationCommand
```

The Class GetNexusStatus.

Field Summary		Page
static int	**TIMEOUT** The timeout value (ms) for this command.	80

Fields inherited from class mdt.neuro.nexus.ApplicationCommand
commandBytes, commandCode, crcValid, request, response, responseCode, responseData, timeout

Method Summary		Page
protected void	**consumeBytes**(List<Byte> bytes)	81
NexusStatus	**getStatus**() Gets the status.	80

Methods inherited from class mdt.neuro.nexus.ApplicationCommand
getRequestData, getResponseBytes, getResponseData

Field Detail

TIMEOUT

```
public static final int TIMEOUT
```

The timeout value (ms) for this command.

Method Detail

getStatus

```
public NexusStatus getStatus()
```

Gets the status.

Returns:
the status

---

consumeBytes

```
protected void consumeBytes(List<Byte> bytes)
```

Overrides:

consumeBytes in class ApplicationCommand

Class GetRealtimeData

mdt.neuro.nexus.commands

```
java.lang.Object
  └ mdt.neuro.nexus.ApplicationCommand
      └ mdt.neuro.nexus.commands.GetRealtimeData
```

```
public class GetRealtimeData
extends ApplicationCommand
```

The Class GetRealtimeData.

Field Summary		Page
static int	**TIMEOUT** The timeout value (ms) for this command.	82

Fields inherited from class mdt.neuro.nexus.ApplicationCommand
commandBytes, commandCode, crcValid, request, response, responseCode, responseData, timeout

Method Summary		Page
protected void	**consumeBytes**(List<Byte> bytes)	83
DataPacket	**getPacket**() Gets the packet.	82

Methods inherited from class mdt.neuro.nexus.ApplicationCommand
getRequestData, getResponseBytes, getResponseData

Field Detail

TIMEOUT

```
public static final int TIMEOUT
```

The timeout value (ms) for this command.

Method Detail

getPacket

```
public DataPacket getPacket()
```

Gets the packet.

Returns:
the packet

---

consumeBytes

```
protected void consumeBytes(List<Byte> bytes)
```

Overrides:

consumeBytes in class ApplicationCommand

Class IncAmplitude

mdt.neuro.nexus.commands

```
java.lang.Object
  └ mdt.neuro.nexus.ApplicationCommand
      └ mdt.neuro.nexus.commands.IncAmplitude
```

```
public class IncAmplitude
extends ApplicationCommand
```

The Class IncAmplitude.

Field Summary		Page
static int	**RESPONSE_SIZE** The response size.	84
static int	**TIMEOUT** The timeout value (ms) for this command.	84

Fields inherited from class mdt.neuro.nexus.ApplicationCommand
commandBytes, commandCode, crcValid, request, response, responseCode, responseData, timeout

Method Summary		Page
protected void	**consumeBytes**(List<Byte> bytes)	85
double	**getNewAmplitude**() Gets the new amplitude.	85
byte	**getNumExecuted**() Gets the number of INC commands executed – the actual number executed may not match the number requested.	85

Methods inherited from class mdt.neuro.nexus.ApplicationCommand
getRequestData, getResponseBytes, getResponseData

Field Detail

RESPONSE_SIZE

```
public static final int RESPONSE_SIZE
```

The response size.

TIMEOUT

```
public static final int TIMEOUT
```

The timeout value (ms) for this command.

Method Detail

getNewAmplitude

```
public double getNewAmplitude()
```

Gets the new amplitude.

Returns:
the newAmplitude

---

getNumExecuted

```
public byte getNumExecuted()
```

Gets the number of INC commands executed – the actual number executed may not match the number requested.

Returns:
the number of INC commands executed

---

consumeBytes

```
protected void consumeBytes(List<Byte> bytes)
```

Overrides:
consumeBytes in class ApplicationCommand

Class IncFrequency

mdt.neuro.nexus.commands

```
java.lang.Object
  └ mdt.neuro.nexus.ApplicationCommand
      └ mdt.neuro.nexus.commands.IncFrequency
```

```
public class IncFrequency
extends ApplicationCommand
```

The Class IncFrequency.

Field Summary		Page
static int	**RESPONSE_SIZE** The response size.	86
static int	**TIMEOUT** The timeout value (ms) for this command.	86

Fields inherited from class mdt.neuro.nexus.ApplicationCommand
commandBytes, commandCode, crcValid, request, response, responseCode, responseData, timeout

Method Summary		Page
protected void	**consumeBytes**(List<Byte> bytes)	87
int	**getNewFrequency**() Gets the new frequency.	87

Methods inherited from class mdt.neuro.nexus.ApplicationCommand
getRequestData, getResponseBytes, getResponseData

Field Detail

RESPONSE_SIZE

```
public static final int RESPONSE_SIZE
```

The response size.

TIMEOUT

```
public static final int TIMEOUT
```

The timeout value (ms) for this command.

Method Detail

getNewFrequency

```
public int getNewFrequency()
```

Gets the new frequency.

Returns:
the newFrequency

---

consumeBytes

```
protected void consumeBytes(List<Byte> bytes)
```

Overrides:
consumeBytes in class ApplicationCommand

Class IncPulseWidth

mdt.neuro.nexus.commands

```
java.lang.Object
  └ mdt.neuro.nexus.ApplicationCommand
      └ mdt.neuro.nexus.commands.IncPulseWidth
```

```
public class IncPulseWidth
extends ApplicationCommand
```

The Class IncPulseWidth.

Field Summary		Page
static int	**RESPONSE_SIZE** The response size.	88
static int	**TIMEOUT** The timeout value (ms) for this command.	88

Fields inherited from class mdt.neuro.nexus.ApplicationCommand
commandBytes, commandCode, crcValid, request, response, responseCode, responseData, timeout

Method Summary		Page
protected void	**consumeBytes**(List<Byte> bytes)	89
short	**getNewPulseWidth**() Gets the new pulse width.	89

Methods inherited from class mdt.neuro.nexus.ApplicationCommand
getRequestData, getResponseBytes, getResponseData

Field Detail

RESPONSE_SIZE

```
public static final int RESPONSE_SIZE
```

The response size.

TIMEOUT

```
public static final int TIMEOUT
```

The timeout value (ms) for this command.

Method Detail

getNewPulseWidth

```
public short getNewPulseWidth()
```

Gets the new pulse width.

Returns:
the newPulseWidth

---

consumeBytes

```
protected void consumeBytes(List<Byte> bytes)
```

Overrides:
consumeBytes in class ApplicationCommand

Class PulseStim

mdt.neuro.nexus.commands

```
java.lang.Object
  └ mdt.neuro.nexus.ApplicationCommand
      └ mdt.neuro.nexus.commands.PulseStim
```

```
public class PulseStim
extends ApplicationCommand
```

The Class PulseStim.

Field Summary		Page
static int	**responseSize** The response size.	90
static int	**timeout**	90

Fields inherited from class mdt.neuro.nexus.ApplicationCommand
commandBytes, commandCode, crcValid, request, response, responseCode, responseData, timeout

Constructor Summary	Page
PulseStim(byte progNumber, byte numSteps, byte numNop) Instantiates a new pulse stim command.	91

Method Summary		Page
void	**consumeBytes**(List<Byte> bytes)	91
double	**getNewAmplitude**() Gets the new amplitude.	91
byte	**getNumExecuted**() Gets the number of DEC commands executed – the actual number executed may not match the number requested.	91

Methods inherited from class mdt.neuro.nexus.ApplicationCommand
getRequestData, getResponseBytes, getResponseData

Field Detail

responseSize

```
public static final int responseSize
```

The response size.

---

timeout

```
public static final int timeout
```

Constructor Detail

PulseStim

```
public PulseStim(byte progNumber,
                 byte numSteps,
                 byte numNop)
```

Instantiates a new pulse stim command.

Parameters:

`progNumber` - the prog number
`numSteps` - the num steps to INC then DEC
`numNop` - the num nop's to execute in between INC and DEC

Method Detail

getNewAmplitude

```
public double getNewAmplitude()
```

Gets the new amplitude.

Returns:

the newAmplitude

---

getNumExecuted

```
public byte getNumExecuted()
```

Gets the number of DEC commands executed – the actual number executed may not match the number requested.

Returns:

the number of DEC commands executed

---

consumeBytes

```
public void consumeBytes(List<Byte> bytes)
```

Overrides:

`consumeBytes` in class `ApplicationCommand`

Class ResetCycle

mdt.neuro.nexus.commands

```
java.lang.Object
  └ mdt.neuro.nexus.ApplicationCommand
      └ mdt.neuro.nexus.commands.ResetCycle
```

```
public class ResetCycle
extends ApplicationCommand
```

Field Summary		Page
static int	**TIMEOUT** The timeout value (ms) for this command.	92

Fields inherited from class mdt.neuro.nexus.ApplicationCommand
commandBytes, commandCode, crcValid, request, response, responseCode, responseData, timeout

Methods inherited from class mdt.neuro.nexus.ApplicationCommand
consumeBytes, getRequestData, getResponseBytes, getResponseData

Field Detail

TIMEOUT

```
public static final int TIMEOUT
```

The timeout value (ms) for this command.

Class RestoreClinicianSettings

mdt.neuro.nexus.commands

```
java.lang.Object
  └ mdt.neuro.nexus.ApplicationCommand
      └ mdt.neuro.nexus.commands.RestoreClinicianSettings
```

```
public class RestoreClinicianSettings
extends ApplicationCommand
```

The Class RestoreClinicianSettings.

Field Summary		Page
static int	**TIMEOUT** The timeout value (ms) for this command.	93

Fields inherited from class mdt.neuro.nexus.ApplicationCommand
commandBytes, commandCode, crcValid, request, response, responseCode, responseData, timeout

Methods inherited from class mdt.neuro.nexus.ApplicationCommand
consumeBytes, getRequestData, getResponseBytes, getResponseData

Field Detail

TIMEOUT

```
public static final int TIMEOUT
```

The timeout value (ms) for this command.

Class SendTrigger

mdt.neuro.nexus.commands

java.lang.Object
  └ mdt.neuro.nexus.ApplicationCommand
      └ **mdt.neuro.nexus.commands.SendTrigger**

---

public class **SendTrigger**
extends ApplicationCommand

The Class SendTrigger.

Field Summary		Page
static int	**TIMEOUT** The timeout value (ms) for this command.	94

Fields inherited from class mdt.neuro.nexus.ApplicationCommand
commandBytes, commandCode, crcValid, request, response, responseCode, responseData, timeout

Methods inherited from class mdt.neuro.nexus.ApplicationCommand
consumeBytes, getRequestData, getResponseBytes, getResponseData

Field Detail

TIMEOUT

public static final int **TIMEOUT**

The timeout value (ms) for this command.

Class SetActiveGroup

mdt.neuro.nexus.commands

```
java.lang.Object
  └ mdt.neuro.nexus.ApplicationCommand
      └ mdt.neuro.nexus.commands.SetActiveGroup
```

```
public class SetActiveGroup
extends ApplicationCommand
```

The Class SetActiveGroup.

Field Summary		Page
static int	**TIMEOUT** The timeout value (ms) for this command.	95

Fields inherited from class mdt.neuro.nexus.ApplicationCommand
commandBytes, commandCode, crcValid, request, response, responseCode, responseData, timeout

Methods inherited from class mdt.neuro.nexus.ApplicationCommand
consumeBytes, getRequestData, getResponseBytes, getResponseData

Field Detail

TIMEOUT

```
public static final int TIMEOUT
```

The timeout value (ms) for this command.

Class SetNexusConfiguration

mdt.neuro.nexus.commands

```
java.lang.Object
  └ mdt.neuro.nexus.ApplicationCommand
      └ mdt.neuro.nexus.commands.SetNexusConfiguration
```

```
public class SetNexusConfiguration
extends ApplicationCommand
```

The Class SetNexusConfiguration.

Field Summary		Page
static int	**TIMEOUT** The timeout value (ms) for this command.	96

Fields inherited from class mdt.neuro.nexus.ApplicationCommand
commandBytes, commandCode, crcValid, request, response, responseCode, responseData, timeout

Methods inherited from class mdt.neuro.nexus.ApplicationCommand
consumeBytes, getRequestData, getResponseBytes, getResponseData

Field Detail

TIMEOUT

```
public static final int TIMEOUT
```

The timeout value (ms) for this command.

Class StartRealTime

mdt.neuro.nexus.commands

```
java.lang.Object
  └ mdt.neuro.nexus.ApplicationCommand
      └ mdt.neuro.nexus.commands.StartRealTime
```

```
public class StartRealTime
extends ApplicationCommand
```

The Class StartRealTime.

Field Summary		Page
static int	**TIMEOUT** The timeout value (ms) for this command.	97

Fields inherited from class mdt.neuro.nexus.ApplicationCommand
commandBytes, commandCode, crcValid, request, response, responseCode, responseData, timeout

Methods inherited from class mdt.neuro.nexus.ApplicationCommand
consumeBytes, getRequestData, getResponseBytes, getResponseData

Field Detail

TIMEOUT

```
public static final int TIMEOUT
```

The timeout value (ms) for this command.

Class StartSensing

mdt.neuro.nexus.commands

java.lang.Object
  └ mdt.neuro.nexus.ApplicationCommand
      └ **mdt.neuro.nexus.commands.StartSensing**

---

public class **StartSensing**
extends ApplicationCommand

The Class StartSensing.

Field Summary		Page
static int	**TIMEOUT** The timeout value (ms) for this command.	98

Fields inherited from class mdt.neuro.nexus.ApplicationCommand
commandBytes, commandCode, crcValid, request, response, responseCode, responseData, timeout

Methods inherited from class mdt.neuro.nexus.ApplicationCommand
consumeBytes, getRequestData, getResponseBytes, getResponseData

Field Detail

TIMEOUT

public static final int **TIMEOUT**

The timeout value (ms) for this command.

Class StopRealTime

mdt.neuro.nexus.commands

java.lang.Object
  └ mdt.neuro.nexus.ApplicationCommand
      └ **mdt.neuro.nexus.commands.StopRealTime**

---

public class **StopRealTime**
extends ApplicationCommand

The Class StopRealTime.

Field Summary		Page
static int	**TIMEOUT** The timeout value (ms) for this command.	99

Fields inherited from class mdt.neuro.nexus.ApplicationCommand
commandBytes, commandCode, crcValid, request, response, responseCode, responseData, timeout

Methods inherited from class mdt.neuro.nexus.ApplicationCommand
consumeBytes, getRequestData, getResponseBytes, getResponseData

Field Detail

TIMEOUT

public static final int **TIMEOUT**

The timeout value (ms) for this command.

Class StopSensing

mdt.neuro.nexus.commands

```
java.lang.Object
  └ mdt.neuro.nexus.ApplicationCommand
      └ mdt.neuro.nexus.commands.StopSensing
```

```
public class StopSensing
extends ApplicationCommand
```

The Class StopSensing.

Field Summary		Page
static int	**TIMEOUT** The timeout value (ms) for this command.	100

Fields inherited from class mdt.neuro.nexus.ApplicationCommand
commandBytes, commandCode, crcValid, request, response, responseCode, responseData, timeout

Methods inherited from class mdt.neuro.nexus.ApplicationCommand
consumeBytes, getRequestData, getResponseBytes, getResponseData

Field Detail

TIMEOUT

```
public static final int TIMEOUT
```

The timeout value (ms) for this command.

Class TherapyOff

mdt.neuro.nexus.commands

```
java.lang.Object
  └ mdt.neuro.nexus.ApplicationCommand
      └ mdt.neuro.nexus.commands.TherapyOff
```

```
public class TherapyOff
extends ApplicationCommand
```

The Class TherapyOff.

Field Summary		Page
static int	**TIMEOUT** The timeout value (ms) for this command.	101

Fields inherited from class mdt.neuro.nexus.ApplicationCommand
commandBytes, commandCode, crcValid, request, response, responseCode, responseData, timeout

Methods inherited from class mdt.neuro.nexus.ApplicationCommand
consumeBytes, getRequestData, getResponseBytes, getResponseData

Field Detail

TIMEOUT

```
public static final int TIMEOUT
```

The timeout value (ms) for this command.

Class TherapyOn

mdt.neuro.nexus.commands

```
java.lang.Object
  └ mdt.neuro.nexus.ApplicationCommand
      └ mdt.neuro.nexus.commands.TherapyOn
```

```
public class TherapyOn
extends ApplicationCommand
```

The Class TherapyOn.

Field Summary		Page
static int	**TIMEOUT** The timeout value (ms) for this command.	102

Fields inherited from class mdt.neuro.nexus.ApplicationCommand
commandBytes, commandCode, crcValid, request, response, responseCode, responseData, timeout

Methods inherited from class mdt.neuro.nexus.ApplicationCommand
consumeBytes, getRequestData, getResponseBytes, getResponseData

Field Detail

TIMEOUT

```
public static final int TIMEOUT
```

The timeout value (ms) for this command.

Package mdt.neuro.nexus.data

Class Summary		Page
DataPacket	The Class DataPacket.	108
InsInfo	The Class InsInfo.	112
NexusStatus	The Class NexusStatus.	117
ProgramInfo	The Class ProgramInfo.	120

Enum Summary		Page
AmplitudeResolution	The Enum AmplitudeResolution.	103
ChannelType	The Enum ChannelType.	106
NexusState	The Enum NexusState.	115
SensingState	The Enum SensingState.	123
TherapyParameter	The Enum TherapyParameter.	125

Enum AmplitudeResolution

mdt.neuro.nexus.data

```
java.lang.Object
  └ java.lang.Enum<AmplitudeResolution>
      └ mdt.neuro.nexus.data.AmplitudeResolution
```

All Implemented Interfaces:

Comparable<AmplitudeResolution>, Serializable

```
public enum AmplitudeResolution
extends Enum<AmplitudeResolution>
```

The Enum AmplitudeResolution.

Enum Constant Summary	Page
CURRENT Constant current.	104
VOLTAGE Constant voltage.	104

Method Summary		Page
static AmplitudeResolution	**valueOf**(String name)	105
static AmplitudeResolution[]	**values**()	104

Enum Constant Detail

VOLTAGE

```
public static final AmplitudeResolution VOLTAGE
```

Constant voltage.

CURRENT

```
public static final AmplitudeResolution CURRENT
```

Constant current.

Method Detail

values

```
public static AmplitudeResolution[] values()
```

---

valueOf

```
public static AmplitudeResolution valueOf(String name)
```

Enum ChannelType

mdt.neuro.nexus.data

```
java.lang.Object
  └ java.lang.Enum<ChannelType>
      └ mdt.neuro.nexus.data.ChannelType
```

All Implemented Interfaces:

Comparable<ChannelType>, Serializable

```
public enum ChannelType
extends Enum<ChannelType>
```

The Enum ChannelType.

Enum Constant Summary	Page
DISABLED The channel is disabled.	106
POWER The channel is power.	106
TIME The channel is time domain.	106

Method Summary		Page
static ChannelType	**valueOf**(String name)	107
static ChannelType[]	**values**()	107

Enum Constant Detail

DISABLED

```
public static final ChannelType DISABLED
```

The channel is disabled.

POWER

```
public static final ChannelType POWER
```

The channel is power.

TIME

```
public static final ChannelType TIME
```

The channel is time domain.

Method Detail

values

```
public static ChannelType[] values()
```

---

valueOf

```
public static ChannelType valueOf(String name)
```

Class DataPacket

mdt.neuro.nexus.data

```
java.lang.Object
  └ mdt.neuro.nexus.data.DataPacket
```

```
public class DataPacket
extends Object
```

The Class DataPacket.

Method Summary		Page
boolean	**convertToMillivolts**(String fileName) Converts the time domain data in this packet to mV, based on the given XML header file from the Sensing Programmer.	111
boolean	**equals**(Object obj)	111
byte	**getActiveGroup**() Gets the active group.	110
int[]	**getChSampleRates**() Gets the ch sample rates.	110
ChannelType[]	**getChTypes**() Gets the ch types.	110
double[][]	**getConvertedData**() Gets the converted data.	109
short[][]	**getData**() Gets the data.	110
short	**getNumMissedPatterns**() Gets the number of missed packets previous to this packet.	109
short	**getPatternNum1**() Gets the first pattern num.	109
short	**getPatternNum2**() Gets the second pattern num.	109
double[][]	**getSubtractedData**() Gets the template subtracted data.	108
int	**hashCode**()	111
byte	**isStimOn**() Checks if is stim on.	109
void	**subtractTemplate**(double[][] template) Subtract the given template from this packet and store the result in subtracted data.	110
String	**toString**()	111

Method Detail

getSubtractedData

```
public double[][] getSubtractedData()
```

Gets the template subtracted data.

Returns:
the template subtracted data

---

getConvertedData

```
public double[][] getConvertedData()
```

Gets the converted data.

Returns:
the converted data

---

getNumMissedPatterns

```
public short getNumMissedPatterns()
```

Gets the number of missed packets previous to this packet.

Returns:
the number of missed packets

---

getPatternNum1

```
public short getPatternNum1()
```

Gets the first pattern num.

Returns:
the first pattern num

---

getPatternNum2

```
public short getPatternNum2()
```

Gets the second pattern num.

Returns:
the second pattern num

---

isStimOn

```
public byte isStimOn()
```

Checks if is stim on.

Returns:
the stimOn

---

getActiveGroup

```
public byte getActiveGroup()
```

Gets the active group.

Returns:
the activeGroup

---

getData

```
public short[][] getData()
```

Gets the data.

Returns:
the data

---

getChSampleRates

```
public int[] getChSampleRates()
```

Gets the ch sample rates.

Returns:
the ch sample rates

---

getChTypes

```
public ChannelType[] getChTypes()
```

Gets the ch types.

Returns:
the ch types

---

subtractTemplate

```
public void subtractTemplate(double[][] template)
```

Subtract the given template from this packet and store the result in subtracted data.

Parameters:
`template` - the template

---

convertToMillivolts

```
public boolean convertToMillivolts(String fileName)
                            throws ParserConfigurationException,
                                   org.xml.sax.SAXException,
                                   IOException
```

Converts the time domain data in this packet to mV, based on the given XML header file from the Sensing Programmer.

Parameters:

`fileName` - the file name of the XML header

Throws:

`ParserConfigurationException` - the parser configuration exception

`org.xml.sax.SAXException` - the sAX exception

`IOException` - Signals that an I/O exception has occurred.

---

hashCode

```
public int hashCode()
```

Overrides:

`hashCode` in class `Object`

---

equals

```
public boolean equals(Object obj)
```

Overrides:

`equals` in class `Object`

---

toString

```
public String toString()
```

Overrides:

`toString` in class `Object`

Class InsInfo

mdt.neuro.nexus.data

```
java.lang.Object
  └ mdt.neuro.nexus.data.InsInfo
```

```
public class InsInfo
extends Object
```

The Class InsInfo. This class represents stimulation information and the status of sensing within the INS.

Field Summary		Page
static int	**EXPECTED_SIZE** The expected size.	112

Method Summary		Page
boolean	**equals**(Object obj)	113
int	**getActiveGroupFrequency**() Gets the active group frequency.	113
byte	**getActiveGroupNumber**() Gets the active group number.	113
ProgramInfo[]	**getPrograms**() Gets the programs.	113
SensingState	**getSensingState**() Gets the sensing state.	113
int	**hashCode**()	113
boolean	**isTherapyOn**() Checks if is therapy on.	112
String	**toString**()	114

Field Detail

EXPECTED_SIZE

```
public static final int EXPECTED_SIZE
```

The expected size.

Method Detail

isTherapyOn

```
public boolean isTherapyOn()
```

Checks if is therapy on.

Returns:
the therapyOn

getSensingState

```
public SensingState getSensingState()
```

Gets the sensing state.

Returns:
the sensing state

getActiveGroupNumber

```
public byte getActiveGroupNumber()
```

Gets the active group number.

Returns:
the activeGroupNumber

getActiveGroupFrequency

```
public int getActiveGroupFrequency()
```

Gets the active group frequency.

Returns:
the activeGroupFrequency

getPrograms

```
public ProgramInfo[] getPrograms()
```

Gets the programs.

Returns:
the programs

hashCode

```
public int hashCode()
```

Overrides:
hashCode in class Object

equals

```
public boolean equals(Object obj)
```

Overrides:
equals in class Object

---

toString

```
public String toString()
```

Overrides:
toString in class Object

Enum NexusState

mdt.neuro.nexus.data

```
java.lang.Object
  └ java.lang.Enum<NexusState>
      └ mdt.neuro.nexus.data.NexusState
```

All Implemented Interfaces:

Comparable<NexusState>, Serializable

```
public enum NexusState
extends Enum<NexusState>
```

The Enum NexusState.

Enum Constant Summary	Page
IDLE The nexus is idle.	115
INS_CONNECTED The ins connected.	116
LINK_FAILED_DEVICE_ERR The link failed - device error.	116
LINK_FAILED_NO_RESPONSE The link failed - no response from INS.	116
LINKING_TO_INS The nexus is linking to INS.	115
MAINTENANCE_ENABLED The maintenance enabled.	116

Method Summary		Page
static NexusState	**valueOf**(String name)	116
static NexusState[]	**values**()	116

Enum Constant Detail

IDLE

```
public static final NexusState IDLE
```

The nexus is idle.

LINKING_TO_INS

```
public static final NexusState LINKING_TO_INS
```

The nexus is linking to INS.

LINK_FAILED_NO_RESPONSE

```
public static final NexusState LINK_FAILED_NO_RESPONSE
```

The link failed - no response from INS.

LINK_FAILED_DEVICE_ERR

```
public static final NexusState LINK_FAILED_DEVICE_ERR
```

The link failed - device error.

INS_CONNECTED

```
public static final NexusState INS_CONNECTED
```

The ins connected.

MAINTENANCE_ENABLED

```
public static final NexusState MAINTENANCE_ENABLED
```

The maintenance enabled.

Method Detail

values

```
public static NexusState[] values()
```

---

valueOf

```
public static NexusState valueOf(String name)
```

Class NexusStatus

mdt.neuro.nexus.data

```
java.lang.Object
  └ mdt.neuro.nexus.data.NexusStatus
```

```
public class NexusStatus
extends Object
```

The Class NexusStatus.

Field Summary		Page
static int	**EXPECTED_SIZE** The Constant EXPECTED_SIZE.	117

Method Summary		Page
boolean	**equals**(Object obj)	119
double	**getBatteryPercent**() Gets the battery percent.	118
byte	**getHostTimeoutMinutes**() Gets the host timeout minutes.	118
byte	**getMaintenanceTimeoutSeconds**() Gets the maintenance timeout seconds.	119
byte	**getMajorVersion**() Gets the major version.	118
byte	**getMinorVersion**() Gets the minor version.	118
NexusState	**getState**() Gets the state.	117
int	**hashCode**()	119
boolean	**isBatteryDepleted**() Checks if is battery depleted.	118
String	**toString**()	119

Field Detail

EXPECTED_SIZE

```
public static final int EXPECTED_SIZE
```

The Constant EXPECTED_SIZE.

Method Detail

getState

```
public NexusState getState()
```

Gets the state.

Returns:
the state

---

getMajorVersion

```
public byte getMajorVersion()
```

Gets the major version.

Returns:
the majorVersion

---

getMinorVersion

```
public byte getMinorVersion()
```

Gets the minor version.

Returns:
the minorVersion

---

getBatteryPercent

```
public double getBatteryPercent()
```

Gets the battery percent.

Returns:
the batteryPercent

---

isBatteryDepleted

```
public boolean isBatteryDepleted()
```

Checks if is battery depleted.

Returns:
the batteryDepleted

---

getHostTimeoutMinutes

```
public byte getHostTimeoutMinutes()
```

Gets the host timeout minutes.

Returns:
the hostTimeoutMinutes

getMaintenanceTimeoutSeconds

```
public byte getMaintenanceTimeoutSeconds()
```

Gets the maintenance timeout seconds.

Returns:
the maintenanceTimeoutSeconds

hashCode

```
public int hashCode()
```

Overrides:
hashCode in class Object

equals

```
public boolean equals(Object obj)
```

Overrides:
equals in class Object

toString

```
public String toString()
```

Overrides:
toString in class Object

Class ProgramInfo

mdt.neuro.nexus.data

```
java.lang.Object
  └ mdt.neuro.nexus.data.ProgramInfo
```

```
public class ProgramInfo
extends Object
```

The Class ProgramInfo. This class represents the state of a stimulation program in the INS.

Field Summary		Page
static double	**AMP_STEP_SIZE_CC** The constant current amp step size in mA.	120
static double	**AMP_STEP_SIZE_CV** The constant voltage amp step size in V.	120

Method Summary		Page
boolean	**equals**(Object obj)	121
double	**getAmplitude**() Gets the amplitude in V.	121
AmplitudeResolution	**getAmplitudeResolution**() Gets the amplitude resolution.	121
byte	**getProgramIndex**() Gets the program index.	121
int	**getPulseWidth**() Gets the pulse width in uSec.	121
int	**hashCode**()	121
String	**toString**()	122

Field Detail

AMP_STEP_SIZE_CV

```
public static final double AMP_STEP_SIZE_CV
```

The constant voltage amp step size in V.

AMP_STEP_SIZE_CC

```
public static final double AMP_STEP_SIZE_CC
```

The constant current amp step size in mA.

Method Detail

getProgramIndex

```
public byte getProgramIndex()
```

Gets the program index.

Returns:
the programIndex

---

getAmplitude

```
public double getAmplitude()
```

Gets the amplitude in V. This amplitude is the actual value.

Returns:
the amplitude

---

getAmplitudeResolution

```
public AmplitudeResolution getAmplitudeResolution()
```

Gets the amplitude resolution. This value is set by the 8840 to be 1, which corresponds to 50 mV steps.

Returns:
the amplitudeResolution

---

getPulseWidth

```
public int getPulseWidth()
```

Gets the pulse width in uSec.

Returns:
the pulseWidth

---

hashCode

```
public int hashCode()
```

Overrides:
hashCode in class Object

---

equals

```
public boolean equals(Object obj)
```

Overrides:

equals in class Object

---

toString

```
public String toString()
```

Overrides:

toString in class Object

Enum SensingState

mdt.neuro.nexus.data

```
java.lang.Object
  └ java.lang.Enum<SensingState>
      └ mdt.neuro.nexus.data.SensingState
```

All Implemented Interfaces:

Comparable<SensingState>, Serializable

```
public enum SensingState
extends Enum<SensingState>
```

The Enum SensingState.

Enum Constant Summary	Page
DISABLED Sensing disabled.	123
ENABLED Sensing enabled.	123
UNAVAILABLE Status unavailable.	123

Method Summary		Page
static SensingState	**valueOf**(String name)	124
static SensingState[]	**values**()	124

Enum Constant Detail

DISABLED

```
public static final SensingState DISABLED
```

Sensing disabled.

ENABLED

```
public static final SensingState ENABLED
```

Sensing enabled.

UNAVAILABLE

```
public static final SensingState UNAVAILABLE
```

Status unavailable.

Method Detail

values

```
public static SensingState[] values()
```

---

valueOf

```
public static SensingState valueOf(String name)
```

Enum TherapyParameter

mdt.neuro.nexus.data

```
java.lang.Object
  └ java.lang.Enum<TherapyParameter>
      └ mdt.neuro.nexus.data.TherapyParameter
```

All Implemented Interfaces:

Comparable<TherapyParameter>, Serializable

```
public enum TherapyParameter
extends Enum<TherapyParameter>
```

The Enum TherapyParameter.

Enum Constant Summary	Page
AMPLITUDE The Amplitude.	125
FREQUENCY The Frequency.	125
PULSEWIDTH The Pulse width.	125

Method Summary		Page
static TherapyParameter	**valueOf**(String name)	126
static TherapyParameter[]	**values**()	126

Enum Constant Detail

AMPLITUDE

```
public static final TherapyParameter AMPLITUDE
```

The Amplitude.

PULSEWIDTH

```
public static final TherapyParameter PULSEWIDTH
```

The Pulse width.

FREQUENCY

```
public static final TherapyParameter FREQUENCY
```

The Frequency.

Method Detail

values

```
public static TherapyParameter[] values()
```

---

valueOf

```
public static TherapyParameter valueOf(String name)
```

Package mdt.neuro.nexus.support

Class Summary		Page
NexusLogger	The Class NexusLogger.	127

Class NexusLogger

mdt.neuro.nexus.support

```
java.lang.Object
  └ mdt.neuro.nexus.support.NexusLogger
```

```
public class NexusLogger
extends Object
```

The Class NexusLogger.

Constructor Summary	Page
NexusLogger()	128

Method Summary		Page
static Logger	**getLogger**() Gets the logger.	128

Constructor Detail

NexusLogger

```
public NexusLogger()
```

Method Detail

getLogger

```
public static Logger getLogger()
```

Gets the logger.

Returns:
the logger
