UHF API introduction

UHFRManager

1.1 getInstance()

Function	**UHFRManager getInstance()**
Description	Get UHF instance, and open the hardware module
Parameter	Null
Return	UHFRManager instance

1.2 getHardware()

Function	String getHardware()

Description	Get hardware version
Parameters	Null
Return	Hardware version string Null failure

1.3 asyncStartReading()

Function	READER_ERR asyncStartReading()
Description	Start reading of multiple mode
Parameters	Null
Return	READER_ERR Success: Reader.READER_ERR.MT_OK_ERR Failure

1.4 asyncStopReading()

Function	READER_ERR asyncStopReading()
Description	Stop reading of multiple mode, use together with asyncStartReading()
Parameters	Null
Return	READER_ERR Success: Reader.READER_ERR.MT_OK_ERR Failure

1.5 setInventoryFilter()

Function	boolean setInventoryFilter()
Description	Set inventory filter
Parameters	byte[] fdata, data to be filtered int fbank, bank of data to be filtered 1 :EPC, 2: TID , 3: USER int fstartaddr, starting address，unit of word boolean matching true: inventory the matched tag; false: inventory the non-matching tag
Return	True for success False for failure

Note: 2 bytes == 1 word

1.6 setCancleInventoryFilter()

Function	boolean setCancleInventoryFilter()
Description	Cancell inventory filter, use together with setInventoryFilter()
Parameters	Null
Return	True for success; False for failure

1.7 tagInventoryRealTime()

Function	public List<TAGINFO> tagInventoryRealTime()
Description	Inventory in real-time and show the tag list, use after asyncStartReading()
Parameters	Null
Return	List<TAGINFO> Tag list Null

1.8 stopTagInventory()

Function	boolean stopTagInventory()
Description	Stop inventory
Parameters	Null
Return	True for success False for failure

1.9 tagInventoryByTimer()

Function	List<TAGINFO> tagInventoryByTimer(short readtime)
Description	Inventory by timer
Parameters	Short readtime, time of single inventory with unit of ms
Return	List<TAGINFO> Tag list Null: reading failure

1.10 getTagData()

Function	READER_ERR getTagData(int mbank, int startaddr, int len, byte[] rdata, byte[] password, short timeout)
Description	Get tag data
Parameters	int mbank, bank to be read, 0:RESERVED, 1:EPC,2:TID, 3:USER int startaddr, starting address of the tag with unit of word int len, length of the tag byte[] rdata, read data, same length as len byte[] password, access password, 4 bytes short timeout，timeout value with unit of ms
Return	READER_ERR Success: Reader.READER_ERR.MT_OK_ERR Failure

1.11 getTagDataByFilter()

Function	byte[] getTagDataByFilter(int mbank, int startaddr, int len,byte[] password, short timeout, byte[] fdata, int fbank,int fstartaddr, boolean matching)
Description	Read specified tag by filter
Parameters	int mbank, memory bank, 0:RESERVED，1:EPC，2:TID，3:USER int startaddr, starting address with unit of word int len, length of the data to be read byte[] password, access password, 4 bytes short timeout, timeout value with unit of ms byte[] fdata, filtered data

	int fbank, filtered bank, 1 :EPC,2: TID ,3: USER int fstartaddr, starting address with unit of word boolean matching or not, true: read the matching tag; false: read the non-matching tag
Return	Byte[] Success Null Failure

1.12 writeTagData()

Function	READER_ERR writeTagData(char mbank, int startaddress, byte[] data, int datalen, byte[] accesspasswd, short timeout)
Description	Write data
Parameters	int mbank, memory bank to be written，0:RESERVED，1:EPC，2:TID，3:USER int startaddress, starting address with unit of word byte[] data, data to be written int datalen, length of the data with unit of word byte[] password, access password, 4 bytes short timeout，timeout value with unit of ms
Return	READER_ERR Success: Reader.READER_ERR.MT_OK_ERR Failure: error information

1.13 writeTagDataByFilter()

Function	READER_ERR writeTagDataByFilter(char mbank, int startaddress, byte[] data, int datalen, byte[] accesspasswd, short timeout, byte[] fdata, int fbank, int fstartaddr, boolean matching)
Description	Write data to the specified tag by filter
Parameters	int mbank, memory bank of data to be written，0:RESERVED，1:EPC，2:TID，3:USER int startaddress, starting address with unit of word byte[] data, data to be written int datalen, length of the data to be written with unit of word byte[] password, access password, 4 bytes short timeout，timeout value with unit of ms byte[] fdata, filtered data int fbank, filtered bank 1 :EPC,2: TID ,3: USER int fstartaddr, starting address with unit of word boolean matching or not，true: write data to the matching tag; false: write data to the non-matching tag
Return	READER_ERR Success：Reader.READER_ERR.MT_OK_ERR Failure：error information

1.14 writeTagEPC()

Function	READER_ERR writeTagEPC(byte[] data, byte[] accesspwd, short timeout)
Description	Write EPC
Parameters	byte[] data, EPC data to be written byte[] accesspwd, access password, 4 bytes short timeout, timeout value with unit of ms

Return	READER_ERR Success：Reader.READER_ERR.MT_OK_ERR Failure: error list

1.15 writeTagEPCByFilter()

Function	READER_ERR writeTagEPCByFilter(byte[] data, byte[] accesspwd, short timeout, byte[] fdata, int fbank, int fstartaddr, boolean matching)
Description	Write EPC of specified tag by filter
Parameters	byte[] data, EPC data to be written byte[] accesspwd, access password, 4 bytes short timeout, timeout value with unit of ms byte[] fdata, filtered data int fbank, filtered bank 1 :EPC,2: TID ,3: USER int fstartaddr, starting address with unit of word boolean matching or not, true: write data to the matching tag; false: write data to the non-matching tag
Return	READER_ERR Success: Reader.READER_ERR.MT_OK_ERR Failure: error information

1.16 lockTag()

Function	READER_ERR lockTag(Lock_Obj lockobject, Lock_Type locktype, byte[] accesspasswd, short timeout)
Description	Lock tag
Parameters	Lock_Obj lockobject, lock object including access password, kill password, EPC bank and USER bank, please refer to the demo source codes Lock_Type locktype, lock type, including lock/unlock, permanent lock, please refer to the demo source codes byte[] accesspasswd, access password, 4 bytes short timeout, timeout value with unit of ms
Return	READER_ERR Success: Reader.READER_ERR.MT_OK_ERR Failure: error information

1.17 lockTagByFilter()

Function	READER_ERR lockTagByFilter(Lock_Obj lockobject, Lock_Type locktype, byte[] accesspasswd, short timeout, byte[] fdata, int fbank, int fstartaddr, boolean matching)
Description	Lock tag by filter
Parameters	Lock_Obj lockobject, lock object including access password, kill password, EPC bank and USER bank, please refer to the demo source codes Lock_Type locktype, lock type, including lock/unlock, permanent lock, please refer to the demo source codes byte[] accesspasswd, access password, 4 bytes short timeout, timeout value with unit of ms byte[] fdata, filtered data int fbank, filtered bank, 1 :EPC,2: TID ,3: USER int fstartaddr, starting address with unit of word boolean matching or not，true: lock the tag matching the filter; false: lock

	the tag non-matching the filter
Return	READER_ERR Success: Reader.READER_ERR.MT_OK_ERR Failure: error information

1.18 killTag()

Function	READER_ERR killTag(byte[] killpasswd, short timeout)
Description	Kill the tag, please be aware about the operating since the tag would not work after being killed Tag cannot be killed with password of 0
Parameters	byte[] killpassword, kill password short timeout, timeout value with unit of ms
Return	READER_ERR Success: Reader.READER_ERR.MT_OK_ERR Failure;

1.19 killTagByFilter()

Function	READER_ERR killTagByFilter(byte[] killpasswd, short timeout, byte[] fdata, int fbank, int fstartaddr, boolean matching)
Descrition	Kill the specified tag by filter
Parameters	byte[] killpasswd, kill password short timeout, timeout value with unit of ms byte[] fdata, data to be filtered int fbank, bank of data to be filtered 1 :EPC,2: TID ,3: USER int fstartaddr, starting address with unit of word boolean matching or not; true: kill the tag that matches the filter data, ,false: kill the tag that does not match the filter
Return	READER_ERR Success: Reader.READER_ERR.MT_OK_ERR Failure:

1.20 setRegion()

Function	READER_ERR setRegion(Region_Conf region)
Description	Set the frequency region
Parameters	Region_Conf region, including CHN, USA, Korea, EU. Failure indicates the frequency is not supported
Return	READER_ERR Success: Reader.READER_ERR.MT_OK_ERR Failure

1.21 getRegion()

Function	Region_Conf getRegion()
Description	Get the current frequency region
Parameters	Null
Return	Region_Conf, please refer to the demo source codes

1.22 getFrequencyPoints()

Function	int[] getFrequencyPoints()
Description	Get the frequency points
Parameters	Null
Return	Int[] Frequency points with unit of kHz

1.23 setFrequencyPoints()

Function	READER_ERR setFrequencyPoints(int[] frequencyPoints)
Description	Set frequency points, frequency points can be obtained by getFrequencyPoints()
Parameters	int[] frequencyPoints with unit of kHz
Return	READER_ERR Success: Reader.READER_ERR.MT_OK_ERR Failure

1.24 setPower()

Function	READER_ERR setPower(int readPower, int writePower)
Description	Set the power value
Parameters	int readPower, range: 5~33 int writePower, range: 5~33
Return	READER_ERR Success: Reader.READER_ERR.MT_OK_ERR Failure:

1.25 getPower()

Function	int[] getPower()
Description	Get the power
Parameters	Null
Return	Int[] success, int[0] as reading power, int[1] as writing power Null failure

1.26 setFastMode()

Function	READER_ERR setFastMode()
Description	Set fast/multiple reading mode with the max power
Parameters	Null
Return	List<EPCDataModel>: EPC data list Null

1.27 getTemperature()

Function	int getTemperature()
Description	Get the module chip temperature
Parameters	Null
Return	int >0 as success, <0 as failure

1.28 close()

Function	Boolean close()
Description	Close the hardware connection

Parameters	Null
Return	True for success; Fail for failure

For more details, please refer to the demo source codes.