	Class Name (some names may be different in CS108 and CS710S)	CS108	CS710S	Description/Notes/Explanation
Library Class				
Package Name		com.csl.cs108library4a	com.csl.cs710library4a	
Library version		2.6.2	3.1.3	

Public Constants				
	OperationTypes	TAG RDOEM,	(Not exposed to application)	Parameters for startOperation in CS108
		TAG INVENTORY COMPACT,		
		TAG INVENTORY,		
		TAG SEARCHING		
	HostCmdResponseTypes in CS108 UplinkPacketTypes in CS710S	NULL,	TYPE UPLINK RESPONDED,	Define different types of uplink data returned
			TYPE UPLINK1 RESPONDED,	
			TYPE BARCODE GOODREAD,	
		TYPE COMMAND BEGIN,	TYPE COMMAND BEGIN,	
		TYPE COMMAND END,	TYPE COMMAND END,	
		TYPE 18K6C INVENTORY,	TYPE 18K6C INVENTORY,	
		TYPE 18K6C INVENTORY COMPACT,	TYPE 18K6C INVENTORY COMPACT,	
		TYPE 18K6C TAG ACCESS,	TYPE 18K6C TAG ACCESS,	
		TYPE ANTENNA CYCLE END	TYPE ANTENNA CYCLE END,	
			TYPE ERRORCODE,	
			TYPE BARCODE INVENTORY	
	HostCommands	NULL,	NULL,	Define different host RFID host commands
		CMD WROEM,	CMD WROEM,	
		CMD RDOEM,	CMD RDOEM,	
		CMD ENGTEST,	CMD ENGTEST,	
		CMD MBPRDREG,	CMD MBPRDREG,	
		CMD MBPWRREG,	CMD MBPWRREG,	
		CMD 18K6CINV,	CMD 18K6CINV,	
		CMD 18K6CREAD,	CMD 18K6CREAD,	
		CMD 18K6CWRITE,	CMD 18K6CWRITE,	
		CMD 18K6CLOCK,	CMD 18K6CLOCK,	
		CMD 18K6CKILL,	CMD 18K6CKILL,	
		CMD SETPWRMGMTCFG,	CMD SETPWRMGMTCFG,	
		CMD UPDATELINKPROFILE,	CMD UPDATELINKPROFILE,	
		CMD 18K6CBLOCKWRITE,	CMD 18K6CBLOCKWRITE,	
		CMD CHANGEEAS,	CMD CHANGEEAS,	
		CMD GETSENSORDATA,	CMD GETSENSORDATA,	
		CMD AUTHENTICATE,	CMD AUTHENTICATE,	
		CMD READBUFFER,	CMD READBUFFER,	
		CMD UNTRACEABLE,	CMD UNTRACEABLE,	
		CMD FDM RDMEM,	CMD FDM RDMEM,	
		CMD FDM WRMEM,	CMD FDM WRMEM,	
		CMD FDM AUTH,	CMD FDM AUTH,	
		CMD FDM GET TEMPERATURE,	CMD FDM GET TEMPERATURE,	
		CMD FDM START LOGGING,	CMD FDM START LOGGING,	
		CMD FDM STOP LOGGING,	CMD FDM STOP LOGGING,	
		CMD FDM WRREG,	CMD FDM WRREG,	
		CMD FDM RDREG,	CMD FDM RDREG,	
		CMD FDM DEEP SLEEP,	CMD FDM DEEP SLEEP,	
		CMD FDM OPMODE CHECK,	CMD FDM OPMODE CHECK,	
		CMD FDM INIT REGFILE,	CMD FDM INIT REGFILE,	
		CMD FDM LED CTRL	CMD FDM LED CTRL,	
			CMD 18K6CINV SELECT,	
			CMD 18K6CINV COMPACT,	
			CMD 18K6CINV COMPACT SELECT,	
			CMD 18K6CINV MB,	
			CMD 18K6CINV MB SELECT	
	CsvColumn	RESERVE BANK,	RESERVE BANK,	Define different items of Csv Column set for Csv file saving
		EPC BANK,	EPC BANK,	
		TID BANK,	TID BANK,	
		USER BANK,	USER BANK,	
		PHASE,	PHASE,	
		CHANNEL,	CHANNEL,	
		TIME,	TIME,	
		TIMEZONE,	TIMEZONE,	
		LOCATION,	LOCATION,	
		DIRECTION,	DIRECTION,	
		OTHERS	OTHERS	
		INVALID STATUS	INVALID STATUS	Define as invalid return value of different variables
		INVALID BACKPORT	INVALID BACKPORT	
		INVALID CODESENSOR	INVALID CODESENSOR	
		INVALID CODERSSI	INVALID CODERSSI	
		INVALID SENSORDATA	INVALID SENSORDATA	
		INVALID CODETEMPC	INVALID CODETEMPC	
		iNO SUCH SETTING	iNO SUCH SETTING	
		(No such separation in CS108)	byNO_SUCH_SETTING, strNO SUCH SETTING	Define as invalid byte and string return values
		dBuV dBm constant	(Not exposed to application)	Define conversion constant
	TaskCancelReason	(the object is in Main module in CS108)	NULL,	Define different case when operation is stopped.
			SAME SETTING,	
			INVALD REQUEST,	
			DESTORY,	
			STOP,	
			BUTTON RELEASE,	
			ERROR,	
			TIMEOUT,	
			RFID RESET	
	TagTypes	(no such TagTypes object in CS108. Use more basic Tid and other parameters to distinguish diferent tags)	TAG NULL,	Define different tag to be handled in the library
			TAG IMPINJ4,	
			TAG UCODE8,	
			TAG UCODEDNA,	
			TAG BAPCARD,	
			TAG COLDCHAIN,	
			TAG AURASENSE,	
			TAG MAGNUS,	
			TAG MAGNUS S1,	
			TAG MAGNUS S2,	
			TAG MAGNUS S3,	
			TAG XERXES,	
			TAG FDMICRO,	
			TAG CTESIUS	

Public classes

	ReaderDevice in CS108 DeviceBase (for barcode data) DeviceCsReader (for scanning data), DeviceTag (for tag data) in CS710	ReaderDevice(String name, String address, boolean selected, String details, String strPc, String strXpc, String strCrc16, String strMdid, String strExtra1, int extra1Bank, int extra1Offset, String strExtra2, int extra2Bank, int extra2Offset, String strTimeOfRead, String strTimeZone, String strLocation, String strCompass, int count, double rssi, int phase, int channel, int port, int status, int backPort1, int backPort2, int codeSensor, int codeRssi, float codeTempC, String brand, int sensorData)	DeviceBase(String name, String identity, boolean selected, int count)	Initalise a tag/device data record
		ReaderDevice(String name, String address, boolean selected, String details, int count, double rssi)	DeviceBase(String name, String identity, boolean selected, int count)	Initalise a tag data/device record
		String getName()	String getName()	Get name of data record
		void setName(String name)	void setName(String name)	Set name of data record
		String getAddress()	String getIdentity()	Get address of data record
		void setAddress(String address)	void setIdentity(String identity)	Set address of data record
		boolean getSelected()	boolean getSelected()	Get selected status of the data record
		void setSelected(boolean selected)	void setSelected(boolean selected))	Set selected status of the data record
		int getCount()	int getCount()	Get count of data record
		void setCount(int count)	void setCount(int count)	Set count of data record
		int compareTo(ReaderDevice other)	int compareTo(ReaderDevice other)	Compare routine for sorting sharedObjects
		(No such separation in CS108)	DeviceCsReader(String name, String address, boolean selected, String details, int count, double rssi, int serviceUUID2p1)	Initialisation
		(No such separation in CS108)	DeviceTag(String name, String address, boolean selected, String details, String strPc, String strXpc, String strCrc16, String strMdid, String strExtra1, int extra1Bank, int extra1Offset, String strExtra2, int extra2Bank, int extra2Offset, String strTimeOfRead, String strTimeZone, String strLocation, String strCompass, int count, double rssi, int phase, int channel, int port, int status, int backPort1, int backPort2, int codeSensor, int codeRssi, float codeTempC, String brand, int sensorData)	Initialisation
		(No such separation in CS108)	DeviceTag(String name, String address, boolean selected, String details, int count, double rssi, int serviceUUID2p1)	Initialisation
		(No such separation in CS108)	DeviceTag(String address, boolean selected, int count)	Initialisation
		String getDetails()	String getDetails()	Get details of data record
		void setDetails(String details)	void setDetails(String details)	Set details of data record
		String getPc()	String getPc()	Get Pc of data record
		String getXpc()	String getXpc()	Get Xpc of data record
		setXpc(String strXpc)	setXpc(String strXpc)	Set Xpc of data record
		String getRes()	String getRes()	Get Res of data record
		String getRes2()	String getRes2()	Get Res2 of data record
		String getEpc()	String getEpc()	Get Epc of data record
		String getTid()	String getTid()	Get Tid of data record
		String getUser()	String getUser()	Get User of data record
		String getMdid()	String getMdid()	Get Mdid of data record
		double getRssi()	double getRssi()	Get rssi of data record
		void setRssi(double rssi)	void setRssi(double rssi)	Set rssi of data record
		int getPhase();	int getPhase();	Get phase of data record
		void setPhase(int phase)	void setPhase(int phase)	Set phase of data record
		int getChannel()	int getChannel()	Get channel of data record
		void setChannel(int channel)	void setChannel(int channel)	Set channel of data record
		int getPort()	int getPort()	Get port of data record
		void setPort(int port)	void setPort(int port)	Set port of data record
		int getStatus()	int getStatus()	Get status of data record
		void setStatus(int status)	void setStatus(int status)	Set status of data record
		int getBackport1()	int getBackport1()	Get backport1 of data record
		void setBackport1(int backport1)	void setBackport1(int backport1)	Set backport1 of data record
		int getBackport2()	int getBackport2()	Get backport2 of data record
		void setBackport2(int backport1)	void setBackport2(int backport1)	Set backport2 of data record
		int getCodeSensor()	int getCodeSensor()	Get codesensor of data record
		setCodeSensor(int codeSensor)	setCodeSensor(int codeSensor)	Set codesensor of data record
		int getCodeSensorMax()	int getCodeSensorMax()	Get codesensorMax of data record
		setCodeSensorMax(int codeSensorMax)	setCodeSensorMax(int codeSensorMax)	Set codesensorMax of data record
		int getCodeRssi()	int getCodeRssi()	Get CodeRssi of data record
		void setCodeRssi(int codeRssi)	void setCodeRssi(int codeRssi)	Set CodeRssi of data record
		float getCodeTempC()	float getCodeTempC()	Get CodeTempC of data record
		void setCodeTempC(float codeTempC)	void setCodeTempC(float codeTempC)	Set CodeTempC of data record
		String getBrand()	String getBrand()	Get Brand of data record
		void setBrand(String brand)	void setBrand(String brand)	Set Brand of data record
		int getSensorData()	int getSensorData()	Get SensorData of data record
		void setSensorData(int sensorData)	void setSensorData(int sensorData)	Set SensorData of data record
		String getstrExtra1()	String getstrExtra1()	Get the string of extra bank 1
		setExtra1(String strExtra1, int extra1Bank, int extra1Offset)	setExtra1(String strExtra1, int extra1Bank, int extra1Offset)	Set the string of extra bank 1
		String getstrExtra2()	String getstrExtra2()	Get the string of extra bank 2
		setExtra2(String strExtra2, int extra2Bank, int extra2Offset)	setExtra2(String strExtra2, int extra2Bank, int extra2Offset)	Set the string of extra bank 2
		boolean isBleConnected()	boolean isConnected()	Check connection status of data record
		void setConnected(boolean isConnected)	void setConnected(boolean isConnected)	Set connection status of data record
		String getTimeOfRead()	String getTimeOfRead()	Get TimeOfRead of data record
		String getTimeZone ()	String getTimeZone ()	Get TimeZone of data record
		String getLocation ()	String getLocation ()	Get Location of data record
		void setLocation(String location)	void setLocation(String location)	Set Location of data record
		String getCompass ()	String getCompass ()	Get Compass of data record
		void setCompass(String compass)	void setCompass(String compass)	Set Compass of data record
	Cs108Connector.Cs108ScanData in CS108 CsReaderScanData in CS710S	BluetoothDevice device	String deviceName, deviceAddress; int deviceBondState	Scanned device object
		int rssi	int rssi	device rssi
		byte[] scanRecord	byte[] scanRecord	advertisement data
		BluetoothDevice getDevice()	(Not exposed to application)	Get device object
		String getName()	String getDeviceName()	Get device name
		String getAddress()	String getDeviceAddress()	Get device address
		(No such feature in CS108)	int getDeviceBondState()	
		byte[] getScanRecord()	byte[] getScanRecord()	Get the device advertisement data
	Cs108Connector.Rx000pkgData in CS108 UplinkPacket in CS710S	HostCmdResponseTypes responseType	UplinkPacketTypes decodeDataType	Rx000 response package type
		int flags	(Not necessary as flags is used part of dataHeader)	Rx000 response flag byte
		(Not expose to application in CS108)	byte[] dataHeader	Response package header data
		byte[] dataValues	byte[] dataValues	Rx000 response package data
		long decodedTime	long decodedTime	decoded time field within response data
		double decodedRssi	double decodedRssi	decoded rssi field within response data
		int decodedPhase	int decodedPhase	decoded phase field within response data
		int decodedChidx	int decodedChidx	decoded channel index field within response data
		int decodedPort	int decodedPort	decoded port field within response data
		byte[] decodedPc	byte[] decodedPc	decoded PC field within response data
		byte[] decodedEpc	byte[] decodedEpc	decoded EPC field within response data
		byte[] decodedCrc	byte[] decodedCrc	decoded CRC field within response data
		byte[] decodedData1	byte[] decodedData1	Decoded data1 field within response data
		byte[] decodedData2	byte[] decodedData2	Decoded data2 field within response data
		String decodedResult	String decodedResult	Decoded string available for user application if Ok
		String decodedError	String decodedError	Debug string to be displayed for user application if eror
	Cs108Connector.Rx000EngSetting	void resetRSSI()	(No such Engineering registers in CS710S)	Reset RSSI of the environment read
			String strCrc16;	

	TagData	(TagData is the part of DeviceTag data in CS710S. No such separation in CS108)	String strPc, strXpc;	This is part of data object of DeviceTag. This is displayed as a list during inventory.
			String strEpc, strAddresss;	
			int portstatus;	
			int backport1, backport2, codeSensor, codeRssi;	
			String brand;	
			float codeTempC;	
			int iSensorData;	
			String strExtra1, strExtra2;	
			Library4A.MultiBankData multiBankData;	
	MultiBankData	(in CS108, multibank setting is in different variables, not a single variable)	int extra2Bank; extra1Count, extra2Count, extra1Offset, extra2Offset	This is multibank register in Atmel controller
	AsyncTasknAsync	(Async routines are in the main module in CS108)	void cancel(boolean bCancel)	This is the new routine to replace Async class
			boolean isCancelled()	
			boolean isRunning()	
			public void execute()	
			void publishProgress()	

Public Variables				
		int invalidate	(Not exposed to application)	The number of invalid data received during RFID inventory
		int validate	(Not exposed to application)	The number of valid data received during RFID inventory
		int invalidUpdata	(Not exposed to application)	The number of inventory package that cannot be decoded.

Public constructors				
		Cs108Library4A(Context context, TextView mLogView)	Library4A(Context context, TextView mLogView)	Initialise the class with parameters:
				Context: context, such as getActivity(), of the calling object
				mLogView: the textView in the calling object for debugging purpose.

Public interfaces				
		Cs108Connector.NotificationListener	NotificationListener	

Public methods				
General:				
		String getlibraryVersion()	String getlibraryVersion()	Get the version of library
		boolean checkHostProcessorVersion(String version, int majorVersion, int minorVersion, int buildVersion)	boolean isVersionGreaterEqual(String version, int majorVersion, int minorVersion, int buildVersion)	check if the version is above specific version values.
		void appendToLogView(String s)	void appendToLogView(String s)	Show the debug string to the EditText of the right hand side drawer.
		void appendToLog(String s)	void appendToLog(String s)	Show the debug string to the android studio LogCat window
		String byteArrayToString(byte[] packet)	String byteArrayToString(byte[] packet)	Convert the byte array to String
		float decodeCtesiusTemperature(String strActData, String strCalData)	(this routine is not exposed to application)	Decode the temperature in degree C from the CTesisu tag action data and calibration data
		float decodeMicronTemperature(int iTag35, String strActData, String strCalData)	float decodeMicronTemperature(int iTag35, String strActData, String strCalData)	Decode the temperature in degree C from different Micron tag action data and calibration data
		String strFloat16toFloat32(String strData)	String strFloat16toFloat32(String strData)	Change the 16bit float data to normal 32bit float data
		String str2float16(String strData)	String str2float16(String strData)	Change the string to 16bit float data
		float temperatureC2F (float fTemp)	(Not exposed to application)	Convert temperature from degree C to F
		String temperatureC2F (String strValue)	String temperatureC2F (String strValue)	Convert temperature from degree C to F
		String temperatureF2C (String strValue)	String temperatureF2C (String strValue)	Convert temperature from degree F to C
		void setSameCheck(boolean sameCheck1)	void setSameCheck(boolean sameCheck1)	set if parameter setting check is same as stored value and skips repeated
		void saveSetting2File()	void saveSetting2File()	save the user configurable parameters to file.
		int getRssiDisplaySetting()	int getRssiDisplaySetting()	Get the rssi display type with output: 0 for dBuV, 1 for dBm
		boolean setRssiDisplaySetting(int rssiDisplaySelect)	boolean setRssiDisplaySetting(int rssiDisplaySelect)	Set the rssi display type with parameter: 0 for dBuV, 1 for dBm
		byte getPopulation2Q(int population)	byte getPopulation2Q(int population)	get the Q value from population
		int getPopulation()	int getPopulation()	get the tag population set
		boolean setPopulation(int population)	boolean setPopulation(int population)	set the tag population
		byte getQValue()	byte getQValue()	get the Q Value to be set
		boolean setQValue(byte byteValue)	boolean setQValue(byte byteValue)	set the Q value
		int getBeepCount()	int getBeepCount()	Get beep delay count for tag received
		boolean setBeepCount(int beepCount)	boolean setBeepCount(int beepCount)	set beep delay count for tag received
		boolean getInventoryBeep()	boolean getInventoryBeep()	Get if beep is needed for RFID inventory
		boolean setInventoryBeep(boolean inventoryBeep)	boolean setInventoryBeep(boolean inventoryBeep)	Set if beep is needed for RFID inventory with parameter: true for enable, false for disable
		boolean getInventoryVibrate()	boolean getInventoryVibrateEnable()	Get if vibrator is needed for RFID inventory
		boolean setInventoryVibrate(boolean inventoryVibrate)	boolean setInventoryVibrateEnable(boolean inventoryVibrate)	Set if vibrator is needed for RFID inventory with parameter: true for enable, false for disable
		int getVibrateModeSetting()	int getVibrate4AllSetting()	Get the vibrator operation mode with output: 0 for vibrate for new tag, 1 for vibrate for all tag
		boolean setVibrateModeSetting(int vibrateModeSelect)	boolean setVibrate4AllSetting(int vibrateModeSelect)	Get the vibrator operation mode with parameter: 0 for vibrate for new tag, 1 for vibrate for all tag
		int getVibrateTime()	int getVibrateTime()	Get vibration time in ms for RFID inventory
		boolean setVibrateTime(int vibrateTime)	boolean setVibrateTime(int vibrateTime)	Set vibration time in ms for RFID inventory
		int getVibrateWindow()	int getVibrateWindow()	Get the vibration window in second for RFID inventory
		boolean setVibrateWindow(int vibrateWindow)	boolean setVibrateWindow(int vibrateWindow)	set the vibration window in second for RFID inventory
		boolean setVibrateOn(boolean on)	boolean setVibrateOn(boolean on)	turn the vibrator on or off
				parameter:
				on: true for on. False for off.
		boolean getSaveFileEnable()	boolean getSaveFileEnable()	Get if inventory data is saved to external file
		boolean setSaveFileEnable(boolean saveFileEnable)	boolean setSaveFileEnable(boolean saveFileEnable)	Set if inventory data is saved to external file with parameter: true for enable, false for disable
		boolean getSaveCloudEnable()	boolean getSaveCloudEnable()	Get if inventory data is saved to cloud server
		boolean setSaveCloudEnable(boolean saveCloudEnable)	boolean setSaveCloudEnable(boolean saveCloudEnable)	Set if inventory data is saved to cloud server with parameter: true for enable, false for disable
		boolean getSaveNewCloudEnable()	boolean getSaveNewCloudEnable()	Get if inventory NEW data is saved to cloud server
		boolean setSaveNewCloudEnable(boolean saveNewCloudEnable)	boolean setSaveNewCloudEnable(boolean saveNewCloudEnable)	Set if inventory NEW data is saved to cloud server with parameter: true for enable, false for disable
		boolean getSaveAllCloudEnable()	boolean getSaveAllCloudEnable()	Get if inventory ALL data is saved to cloud server
		boolean setSaveAllCloudEnable(boolean saveNewCloudEnable)	boolean setSaveAllCloudEnable(boolean saveNewCloudEnable)	Set if inventory ALL data is saved to cloud server with parameter: true for enable, false for disable
		String getServerLocation()	String getServerLocation()	Get the cloud server location
		boolean setServerLocation(String serverLocation)	boolean setServerLocation(String serverLocation)	Set the cloud server location
		int getServerTimeout()	int getServerTimeout()	Get the cloud connection timeout in second
		boolean setServerTimeout(int serverTimeout)	boolean setServerTimeout(int serverTimeout)	Set the cloud connection timeout in second
		int getSavingFormatSetting()	int getSavingFormatSetting()	Get the saving format
		boolean setSavingFormatSetting(int savingFormatSelect)	boolean setSavingFormatSetting(int savingFormatSelect)	Set the file saving format
		int getCsvColumnSelectSetting()	int getCsvColumnSelectSetting()	Get columns selected in csv format
		boolean setCsvColumnSelectSetting(int csvColumnSelect)	boolean setCsvColumnSelectSetting(int csvColumnSelect)	Set the columns selected in csv format
		boolean getTriggerReporting()	boolean getTriggerReporting()	Check if trigger auto reporting is enabled
		boolean setTriggerReporting(boolean triggerReporting)	boolean setTriggerReporting(boolean triggerReporting)	Enable/Disable trigger auto reporting
		short getTriggerReportingCount()	short getTriggerReportingCount()	Get the period of trigger auto reporting
		boolean setTriggerReportingCount(short triggerReportingCount)	boolean setTriggerReportingCount(short triggerReportingCount)	Set the period of trigger auto reporting
		boolean setAutoTriggerReporting(byte timeSecond)	(Use setTriggerReportingCount only in CS710S)	Set the period of trigger auto reporting
		boolean stopAutoTriggerReporting()	(Use setTriggerReportingCount only in CS710S)	Stop trigger auto reporting
		String checkVersion()	String checkVersion()	Check different firmware versions.

		int getTriggerCount()	int getTriggerCount()	Get the number of trigger received
		(Use dBuV_dBm_constant directly)	double dBm_to_dBuV(double value)	Conversion from dBm to dBuV
		(Use dBuV_dBm_constant directly)	double dbuV_to_dBm(double value)	Conversion from dBuV to dBm
		String deformatWriteAccessData(String strIn)	String deformatWriteAccessData(String strIn)	Routine to deformat the format in
		(the function is implemented separately in the CS108 application code)	boolean isWriteExtStoragePermitted(boolean requestPermission)	Check if write external file is permitted
		(the function is implemented separately in the CS108 application code)	String openWriteFile(String strFileNameHeader)	Open a file for writing
		(the function is implemented separately in the CS108 application code)	File getWriteFileOpened()	Get the opened file for writing data
		(No such feature in CS108)	boolean isModifiedEntry()	Check if the text box in Setting page is modified before
		(No such feature in CS108)	void resetModifiedEntry()	Reset the modified status of the textbox
		(No such feature in CS108)	void showConfigFile()	Show the configuration parameters saved in Setting page
		(No such feature in CS108)	void showRxGainChangeLog()	Show the log data when the RxGain is changed before in Setting page
Bluetooth related:				
		String getBluetoothICFirmwareVersion()	String getBluetoothICFirmwareVersion()	get the bluetooth firmware version in string, in sequence of major, minor and build versions separated by dot character.
		String getBluetoothICFirmwareName()	String getLinkedDeviceName()	get the bluetooth name
		boolean setBluetoothICFirmwareName(String name)	boolean setLinkedDeviceName(String name)	set the bluetooth name
		boolean isBleScanning()	boolean isScanning()	check if scanning bluetooth devices. Return: true implies scanning
		boolean scanLeDevice(final boolean enable)	boolean scanLeDevice(final boolean enable)	Start/stop scanning Bluetooth devices with parameters: enable: true to start scanning. False to stop scanning
		boolean isConnected()	boolean isConnected()	check if the Bluetooth is connected return: true implies connected.
		boolean connect(ReaderDevice readerDevice)	boolean connect(DeviceCsReader tagDevice)	start Bluetooth connection with parameter: readerDevice: the selected scanned bluetooth device
			boolean connectAgain()	
		void disconnect(boolean tempDisconnect)	void disconnect(boolean tempDisconnect)	start bluetooth disconnection with parameter: tempDisconnect: true to disconnect temporarily. False to disconnect
		boolean forceBTdisconnect()	(Not used in CS710S)	Start bluetooth disconnection immediately
		String getBluetoothDeviceAddress()	String getLinkedDeviceAddress()	get the mac address of the connected bluetooth device
		String getBluetoothDeviceName()	int getLinkedDevicePortNumber()	get the name of the connected bluetooth device
		int getRssi()	int getRssi()	get the rssi signal strength of the bluetooth connection in unit –dbm.
		long getStreamInRate()	long getStreamInRate()	get the data rate of CS108 coming data in byte per second.
		Cs108ScanData getNewDeviceScanned()	Cs108ScanData getNewDeviceScanned()	Get new scanned device information with output: null for nothing.
		(Not such feature in CS108)	int getMtu()	Get the MTU value of the connection
		(this feature is in main CS108 module)	void turnOnLocationDevice(boolean	turn on location device
		(this feature is in main CS108 module)	void turnOnSensorDevice(boolean onStatus)	turn on Ecompass sensor device
		(this feature is in main CS108 module)	String getTimeStamp()	Get the time stamp
		(this feature is in main CS108 module)	String getLocation()	Get the global location data
		(this feature is in main CS108 module)	String getEcompass()	Get the Ecompass data
Host Processor Firmware related:				
		String getHostProcessorICSerialNumber()	String getHostProcessorICSerialNumber()	get the serial number of the reader
		String getHostProcessorICBoardVersion()	String getHostProcessorICBoardVersion()	get the main board version
		String hostProcessorICGetFirmwareVersion()	String hostProcessorICGetFirmwareVersion()	Get the Firmware version of the host processor
		int getcsModel()	(Not exposed to application)	Get the csModel number: 108 for Cs108, 463 for Cs46
Notification related:				
		boolean batteryLevelRequest()	boolean batteryLevelRequest()	request to update the battery level immediately. Return: true for success request.
		boolean setBatteryAutoReport(boolean on)	(Not exposed to application)	set automatic battery reporting
		boolean setAutoRFIDAbort(boolean enable)	boolean setAutoRFIDAbort(boolean enable)	set automatic RFID inventory Abort
		boolean getAutoRFIDAbort ()	boolean getAutoRfidAbortRequest()	Get status of automatic RFID inventory Abort
		boolean setAutoBarStartSTop (boolean enable)	boolean setAutoBarStartSTop (boolean	set automatic Barcode start/stop
		boolean getAutoBarStartSTop()	getAutoBarStartSTopRequest()	Get the status of automatic Barcode start/stop
		int getBatteryLevel()	(Not exposed to application)	get the latest battery level received from the cs108.
		String getBatteryDisplay(boolean voltageDisplay)	String getBatteryDisplay(Boolean voltageDisplay)	Get the display information about battery status with parameter: true for voltage display. False for percentage display
		String isBatteryLow()	String isBatteryLow()	Check if battery is low. If output string is null, Ok. Otherwise, it shows the battery level percentage.
		int getBatteryCount()	int getBatteryCount()	get the battery reported index which is used to check if battery is updated.
		int getBatteryDisplaySetting()	int getBatteryDisplaySetting()	get the battery display type. 0 for voltage display, 1 for percentage display
		boolean setBatteryDisplaySetting(int batteryDisplaySelect)	boolean setBatteryDisplaySetting(int batteryDisplaySelect)	set the battery display type with parameters: 0 for voltage display, 1 for percentage display
		boolean getTriggerButtonStatus()	boolean getTriggerButtonStatus()	get the latest trigger button status received from the cs108.
		(Not expose to application)	boolean getTriggerStatusRequest()	Get the immediate trigger status
		void setNotificationListener(NotificationListener listener)	void setNotificationListener(NotificationListener	set the listener routine for the trigger button.
		byte[] onNotificationEvent()	Library4A.UplinkPacket readCsReaderUplinkPacketDecoded()	Read the RFID uplink data
Barcode detector related:				
		boolean isBarcodeFailure()	boolean isBarcodeFailure()	check if barcode module is failure
		String getBarcodeSerial()	String getBarcodeSerial()	get the serial number of the barcode module
		boolean getBarcodeOnStatus()	boolean getBarcodeOnStatus()	get the on status of the barcode device
		boolean setBarcodeOn(boolean on)	boolean setBarcodeOn(boolean on)	turn the barcode detector on or off parameter: on: true for on. False for off.
		boolean barcodeSendCommandTrigger()	boolean barcodeSendCommandTrigger()	send command data to set trigger mode in the barcode module
		boolean barcodeSendCommandSetPreSuffix()	boolean	send command data to set a pre-defined prefix and suffix in the barcode
		boolean barcodeSendCommandResetPreSuffix()	boolean barcodeSendCommandResetPreSuffix()	send command data to clear the prefix and suffix in the barcode module
		getBarcodePreSuffix()		Start to get the prefix and suffix string of the barcode module
		void getBarcodeReadingMode()		Start to get the reading mode of the barcode module
		boolean barcodeSendCommandConinuous()	boolean barcodeSendCommandConinuous()	send command data to set continuous mode
		boolean barcodeInventory(boolean start)	boolean barcodeInventory(boolean start)	start/stop barcode inventory
		byte[] onBarcodeEvent()	Library4A.UplinkPacket readCsReaderUplinkPacketDecoded()	Get new barcode inventoried information with output: null for nothing.
		String getBarcodeVersion()	String getBarcodeVersion()	Get barcode firmware version
		String getBarcodeESN()	(Not exposed to application)	Get ESN of barcode
		String getBarcodeDate()	String getBarcodeDate()	Get firmware date of the barcode
RFID detector related:				
		boolean getRfidOnStatus()	boolean getRfidOnStatus()	check if RFID detector is on or off. Return: true if on
		boolean setRfidOn(boolean onStatus)	(Not exposed to application)	turn RFID detector on or off parameter: onStatus: true to turn on. False to turn off.
		boolean isRfidFailure()	boolean isRfidFailure()	check if barcode module is failure
		int mrfidToWriteSize()	int mToWriteSize()	get the length of the data queuing to send to RFID detector.
		void mrfidToWritePrint()	(Removed in CS710S)	For debugging purpose, print out the data queue to be sent to RFID detector.
		String getRadioSerial()	String getRadioSerial()	get the serial number of the radio module
		(No such things in CS108)	String getRadioChipSerial()	Get the serial number of the E710 chip
		String getRadioBoardVersion()	String getRadioBoardVersion()	get the board version of the radio module
		String getMacVer()	String getMacVer()	get the Mac version in string, in sequence of major, minor and build version, separated by dot character.
		String getModelNumber()	String getModelNumber()	get the model number of the radio module
		int getPortNumber()	Use getLinkedDevicePortNumber() only in CS710S	Get the number of Port for the csModel
		void macWrite(int address, int value)	(Not exposed to application)	Write data to mac register of rfid module
		public void setReaderDefault()	(Not exposed to application)	Set default CS108 default data
		void set_fdCmdCfg(int value)	(Not exposed to application)	Set Fudan tag CmdCfg register before access
		void set_fdRegAddr(int addr)	(Not exposed to application)	Set Fudan tag RegAddr register before access
		void set_fdWrite(int addr, long value)	(Not exposed to application)	Set Fudan tag Write registers before access

void set_fdPwd(int value)	(Not exposed to application)	Set Fudan tag Pwd register before access
void set_fdBlockAddr4GetTemperature(int addr)	(Not exposed to application)	Set Fudan tag block Address for getting temperature before access
void set_fdReadMem(int addr, long len)	(Not exposed to application)	Set Fudan tag read memory registers before access
void set_fdWriteMem(int addr, int len, long value)	(Not exposed to application)	Set Fudan tag write memory registers before access
void setImpinJExtension(boolean tagFocus, boolean fastId)	void setImpinJExtension(boolean tagFocus, boolean fastId)	Set ImpinJ tag tagFocus and fastID registers before accesss
int getAntennaSelect()	int getAntennaSelect()	Get the antenna port selected for the csModel
boolean setAntennaSelect(int number)	boolean setAntennaSelect(int number)	Set the antenna port selected for the csModel
boolean getAntennaEnable()	boolean getAntennaEnable()	Get the antenna enable status for the csMode
boolean setAntennaEnable(boolean enable)	boolean setAntennaEnable(boolean enable)	Enable/Disable the antenna for the csModel
int getAntennaCycle()	(No such feature in CS710S)	get the antenna cycle value
boolean setAntennaCycle(int antennaCycle)	(No such feature in CS710S)	set the antenna cycle value parameter: antennaCycle from 1 to 65535
boolean setAntennaInvCount(long antennaInvCount)	(No such feature in CS710S)	set the antenna inventory count value from 0 to 0xffffffff
long getAntennaDwell()	long getAntennaDwell()	get the antenna dwell time
boolean setAntennaDwell(long antennaDwell)	boolean setAntennaDwell(long antennaDwell)	set the antenna dwell time parameter: antennaDwell in unit of 1ms.
long getPwrlevel()	long getAntennaPower()	get the antenna power level
boolean setPowerLevel(long pwrlevel)	boolean setAntennaPower(long pwrlevel)	set the antenna power level parameter: pwrlevel in unit of 0.1dbm, up to 300.
int getQueryTarget()	int getQueryTarget()	get the query target value set for inventory.
int getQuerySession()	int getQuerySession()	get the query session value set for inventory.
int getQuerySelect()	int getQuerySelect()	get the query select value set for inventory.
boolean setTagGroup(int sL, int session, int target1)	boolean setTagGroup(int sL, int session, int target1)	set query group parametes parameters: sL: set the query select value for the inventory. From 0 to 3. Session: set the query session for the inventory. From 0 to 3 Flag: set the query target for the inventory. 0 for A. others for B.
int getTagFocus()	int getTagFocus()	get the tag focus value set for inventory.
(implemented in CS108 main module)	int getFastId()	get the tag fastId value set for inventory.
boolean setTagFocus(boolean tagFocusNew)	boolean setTagFocus(boolean tagFocusNew)	set the tag focus value set for inventory.
boolean setInvBrandId(boolean invBrandId)	boolean setInvBrandId(boolean invBrandId)	Set the brand ID for brand checking of the tag information.
boolean getInvAlgo()	boolean getDynamicAlgo()	get the query algorithm set for the inventory.
boolean setInvAlgo(boolean dynamicAlgo)	boolean setDynamicAlgo(boolean dynamicAlgo)	set the query algorithm for the inventory. Parameter: invAlgo: false implies fixed. True implies dynamic algorithm.
List<String> getProfileList()	List<String> getProfileList()	
int getCurrentProfile()	int getCurrentProfile()	get the query profile set for the inventory.
boolean setCurrentLinkProfile(int profile)	boolean setCurrentLinkProfile(int profile)	set the query profile for the inventory. Profile: 0 to 3. See Appendix A for details of the 4 profiles
String getEnvironmentalRSSI()	String getEnvironmentalRSSI()	Get the environmental RSSI read in the RFID module.
void resetEnvironmentalRSSI()	void resetEnvironmentalRSSI()	Reset the environmental RSSI read in the RFID module.
int getHighCompression()	int getHighCompression()	Get High Compression value for inventory
int getRflnaGain()	int getRflnaGain()	Get RF LNA Gain value for inventory
int getIflnaGain ()	int getIflnaGain ()	Get IF LNA Gain value for inventory
int getAgcGain ()	int getAgcGain ()	Get AGC Gain value for inventory
int getRxGain ()	int getRxGain ()	Get RX Gain value for inventory
boolean setRxGain(int highCompression, int rflnagain, int iflnagain, int agcgain)	boolean setRxGain(int highCompression, int rflnagain, int iflnagain, int agcgain)	Set HighCompression, RF LNA Gain, IF LNA Gain, AGC Gain values for inventory
boolean setRxGain(int rxGain)		Set only Rx Gain values for inventory
byte getTagDelay()	byte getTagDelay()	get the tag delay time (in ms) before next tag coming.
boolean setTagDelay(byte tagDelay)	boolean setTagDelay(byte tagDelay)	set the tag delay before next tag coming in unit of ms.
(No such feature in CS108)	byte getSelectDelay()	Get Select delay register in Atmel controller
(No such feature in CS108)	boolean setSelectDelay(byte tagDelay)	Set Select delay register in Atmel controller
(No such feature in CS108)	List<String> getRxAttenList()	Get the enum list of RxAtten register in E710
(No such feature in CS108)	List<String> getMixerList()	Get the enum list of Mixer register in E710
(No such feature in CS108)	List<String> getPga1List()	Get the enum list of Pga1 register in E710
(No such feature in CS108)	List<String> getPga2List()	Get the enum list of Pga2 register in E710
(No such feature in CS108)	List<String> getPga3List()	Get the enum list of Pga3 register in E710
(No such feature in CS108)	short getRxAttenGain()	Get the value of RxAtten register in E710
(No such feature in CS108)	int getMixerGain()	Get the value of Mixer register in E710
(No such feature in CS108)	int getPga1Gain()	Get the value of Pga1 register in E710
(No such feature in CS108)	int getPga2Gain()	Get the value of Pga2 register in E710
(No such feature in CS108)	int getPga3Gain()	Get the value of Pga3 register in E710
(No such feature in CS108)	boolean setRxGain(int rxAttenGain, int mixerGain, int pga1Gain, int pga2Gain, int pga3Gain)	Set the value of rxAtten, mixerGain, pga1, pga2, pga3 of E710
long getCycleDelay()	long getCycleDelay()	get the tag delay before running next inventory cycle in unit of ms
boolean setCycleDelay(long cycleDelay)	boolean setCycleDelay(long cycleDelay)	set the tag delay before running next inventory cycle in unit of ms
(No such feature in CS108)	int getIntraPacketDelay()	Get intraPacket delay in Atmel controller
(No such feature in CS108)	boolean setIntraPacketDelay(int	Set intraPacket delay in Atmel controller
void getAuthenticateReplyLength()	void getAuthenticateReplyLength()	get the length of the authentication reply data
boolean setTam1Configuration(int keyId, String matchData)	boolean setTam1Configuration(int keyId, String matchData)	Set the Tam1 data for the tam1 authentication check
boolean setTam2Configuration(int keyId, String matchData, int profile, int offset, int blockId, int protMode)	boolean setTam2Configuration(int keyId, String matchData, int profile, int offset, int blockId, int protMode)	Set the Tam2 data for the tam2 authentication check
String getAuthMatchData()		get the authentication match data
boolean setAuthMatchData(String mask)		set the authentication match data
int getUntraceableEpcLength()	int getUntraceableEpcLength()	get the length of the untraceable Epc
boolean setUntraceable(boolean bHideEpc, int ishowEpcSize, int iHideTid, boolean bHideUser, boolean bHideRange)	boolean setUntraceable(boolean bHideEpc, int ishowEpcSize, int iHideTid, boolean bHideUser, boolean bHideRange)	set the untraceable parameters
boolean setUntraceable(int range, boolean user, int tid, int epcLength, boolean epc, boolean uxpc)	boolean setUntraceable(int range, boolean user, int tid, int epcLength, boolean epc, boolean uxpc)	set the untraceable parameters
int getStartQValue()	Int getQValue()	get the start Q value set for dynamic algorithm.
int getMaxQValue()	int getMaxQ()	get the maximum Q value set for dynamic algorithm.
(Set by Q by setDynamicQParms or setFixedQParms)	boolean setMaxQ(int maxQ)	Set max Q in the E710 register
int getMinQValue()	int getMinQ()	get the minimum Q vale set for dynamic algorithm.
(Set by Q by setDynamicQParms or setFixedQParms)	boolean setMinQ(int minQ)	Set min Q in the E710 register
(No such feature in CS108)	byte getNumMinQcycles()	Get number of min Q cycles in E710 register
(No such feature in CS108)	boolean setNumMinQCycles(int numMinQCycles)	Set number of min Q cycles in E710 register
(No such feature in CS108)	byte getQincreaseUseQuery()	Get Q value increase use Query in E710 register
(No such feature in CS108)	boolean setQincreaseUseQuery(boolean qIncreaseUseQuery)	Set Q value increase use Query in E710 register
(No such feature in CS108)	getQdecreaseUseQuery()	Get Q value decrease use Query in E710 register
(No such feature in CS108)	setQdecreaseUseQuery(boolean qDecreaseUseQuery)	Set Q value decrease use Query in E710 register
(No such feature in CS108)	long getMaxQueries()	Get Max queries in E710 register
(No such feature in CS108)	boolean setMaxQueries(long maxQueries)	Set Max queries in E710 register
int getRetryCount()	int getRetryCount()	get the retry count set for dynamic algorithm.
boolean setRetryCount(int retryCount)	boolean setRetryCount(int retryCount)	Set the retry count set for dynamic algorithm.
boolean setDynamicQParms(int startQValue, int minQValue, int maxQValue, int retryCount)	(Not exposed to application)	set the parameters for dynamic algorithm paramters: startQValue: start Q from 0 to 15. minQValue: minimum Q from 0 to 15 maxQValue: maximum Q from 0 to15 retryCount: retry count from 0 to 255
int getFixedQValue()	(Not exposed to application)	get the fixed Q value set for fixed algorithm.
int getFixedRetryCount()	(Not exposed to application)	get the retry count set for the dynamic alogorithm.
boolean getRepeatUnitNoTags()	(Not exposed to application)	check if repeat to run until no tag for fixed algorithm.
boolean setFixedQParms(int qValue, int retryCount, boolean repeatUnitNoTags)	(Not exposed to application)	set the parameters for fixed algorithm. Paramters: qValue: the Q value for fixed algorithm. retryCount: the retry count for the fixed algorithm.

				repeatUnitNoTags: true to repeat until no tag in fixed algorithm.
		public int getInvSelectIndex()	public int getInvSelectIndex()	get the inventory select index
		boolean getSelectEnable()	boolean getSelectEnable()	get the enable status of the inventory select index.
		int getSelectTarget()	int getSelectTarget()	get the target selected for the inventory select index.
		int getSelectAction()	int getSelectAction()	get the action selected for the inventory select index.
		int getSelectMaskBank()	int getSelectMaskBank()	get the mask bank for the inventory select index.
		int getSelectMaskOffset()	int getSelectMaskOffset()	get the mask offset for the inventory select index.
		String getSelectMaskData()	String getSelectMaskData()	get the mask data for the inventory select index.
		boolean setInvSelectIndex(int invSelect)	boolean setInvSelectIndex(int invSelect)	set inventory select index for the inventory select index.
				Parameter: invSelect from 0 to 7.
		boolean setSelectCriteriaDisable()	boolean setSelectCriteriaDisable()	Disable the select of the select index for inventory
		boolean setSelectCriteria(int index, boolean enable, int target, int action, int bank, int offset, String mask, boolean maskbit)	boolean setSelectCriteria(int index, boolean enable, int target, int action, int bank, int offset, String mask, boolean maskbit)	set the select criteria
				parameters:
				enable: the enable status of the inventory select index.
				target: the target selected for the inventory select index.
				action: the action selected for the inventory select index.
				bank: the mask bank for the inventory select index.
				offset: the mask offset for the inventory select index.
				mask: the mask data for the inventory select index.
				maskbit: indicate the mask string is binary or hexadecimal String.
		boolean setSelectCriteria(boolean enable, int target, int action, int bank, int offset, String mask)	boolean setSelectCriteria(boolean enable, int target, int action, int bank, int offset, String mask)	set the select criteria
				parameters:
				enable: the enable status of the inventory select index.
				target: the target selected for the inventory select index.
				action: the action selected for the inventory select index.
				bank: the mask bank for the inventory select index.
				offset: the mask offset for the inventory select index.
				mask: the mask data for the inventory select index.
		boolean setSelectCriteria(int index, boolean enable, int target, int action, int bank, int offset, String mask, int maskblen)	boolean setSelectCriteria(int index, boolean enable, int target, int action, int bank, int offset, String mask, int maskblen)	set the select criteria
				parameters:
				enable: the enable status of the inventory select index.
				target: the target selected for the inventory select index.
				action: the action selected for the inventory select index.
				bank: the mask bank for the inventory select index.
				offset: the mask offset for the inventory select index.
				mask: the mask data for the inventory select index.
				Maskblen: indicate valid bit of the mask string.
		boolean getRssiFilterEnable()	boolean getRssiFilterEnable()	Check if RSSI filtering is enabled or not
		int getRssiFilterType()	int getRssiFilterType()	Check the Filter type
		int getRssiFilterOption()	int getRssiFilterOption()	Check the Filter Option
		boolean setRssiFilterConfig(boolean enable, int rssiFilterType, int rssiFilterOption)	boolean setRssiFilterConfig(boolean enable, int rssiFilterType, int rssiFilterOption)	Set RSSI filtering parameters
		double getRssiFilterThreshold1()	double getRssiFilterThreshold1()	Get RSSI filter threshold 1
		getRssiFilterThreshold2()	getRssiFilterThreshold2()	Get RSSI filter threshold 2
		boolean setRssiFilterThreshold(double rssiFilterThreshold1, double rssiFilterThreshold2)	boolean setRssiFilterThreshold(double rssiFilterThreshold1, double rssiFilterThreshold2)	Set RSSI filter thresholds
		long getRssiFilterCount()	long getRssiFilterCount()	Check RSSI filter count
		setRssiFilterCount(long rssiFilterCount)	setRssiFilterCount(long rssiFilterCount)	Set RSSI filter count
		boolean getInvMatchEnable()	boolean getInvMatchEnable()	get the enable status set for post-filter
		boolean getInvMatchType()	boolean getInvMatchType()	get the match type set for the post-filter.
		int getInvMatchOffset()	int getInvMatchOffset()	get the match offset for the post-filter.
		String getInvMatchData()	String getInvMatchData()	get the mask data for the post-filter.
		boolean setPostMatchCriteria(boolean enable, boolean target, int offset, String mask)	boolean setPostMatchCriteria(boolean enable, boolean target, int offset, String mask)	set the post filter match criteria
				parameters:
				enable: the enable status set for post-filter
				target: the match type set for the post-filter.. True to filter matched ones.
				offset: the match offset for the post-filter.
				mask: the mask data for the post-filter
		int getCountryNumberInList()	int getCountryNumberInList()	get logical channel number used for fixed channel usage.
		String[] getCountryList()	String[] getCountryList()	Get the list of country selectable for the radio device.
		boolean setCountryInList(int countryInList)	boolean setCountryInList(int countryInList)	set the country in the selectable list for the radio device
		boolean getChannelHoppingStatus ()	boolean getChannelHoppingStatus ()	get frequency hopping order. 0 for hopping, 1 for fixed frequency.
		boolean setChannelHoppingStatus(boolean channelOrderHopping)	boolean setChannelHoppingStatus(boolean channelOrderHopping)	Set frequency hopping order. 0 for hopping, 1 for fixed frequency.
		boolean getChannelHoppingDefault()	(Not necessary in CS710S. hopping is fixed as per country)	check if frequency hopping is used
		int getChannel()	int getChannel()	get the channel used
		boolean setChannel(int channelSelect)	boolean setChannel(int channelSelect)	select the channel to be used
		int FreqChnCnt()	int FreqChnCnt()	Get the number of the frequency used
		double getLogicalChannel2PhysicalFreq(int channel)	double getLogicalChannel2PhysicalFreq(int channel)	get actual physical channel number from logical chnnnel number
		(No such feature in CS108)	int getDuplicateEliminationTime()	Get duplication elimination time
		(No such feature in CS108)	boolean setDuplicateEliminationTime(int duplicateEliminationTime)	Set duplication elimination time
		(No such feature in CS108)	boolean setEventPackageUplinkEnable(byte bEnable)	Enable event package uplink
		(No such feature in CS108)	boolean setMultibankReadConfig(int iSet, byte[] byteArrayData)	Set event packet uplink
		boolean setInvModeCompact(boolean invModeCompact)	boolean setInvModeCompact(boolean invModeCompact)	set inventory mode as compact mode if true, normal mode if false.
		boolean setAccessBank(int accessBank)	(Not used. Set multibank Read/Write config in CS710)	set the access bank for access read/write operation.
		boolean setAccessBank(int accessBank, int accessBank2)	(Not used. Set multibank Read/Write config in CS710)	set the access bank for the multi-bank inventory
		boolean setAccessOffset(int accessOffset)	(Not used. Set multibank Read/Write config in CS710)	set the access offset for access read/write operation
		boolean setAccessOffset(int accessOffset, int accessOffset2)	(Not used. Set multibank Read/Write config in CS710)	set the access offset for the multi-bank inventory
		boolean setAccessCount(int accessCount)	(Not used. Set multibank Read/Write config in CS710)	set the access count for access read/write operation
		boolean setAccessCount(int accessCount, int accessCount2)	(Not used. Set multibank Read/Write config in CS710)	set the access count for multi-bank inventory
		boolean setAccessWriteData(String dataInput)	(Not used. Set multibank Read/Write config in CS710)	set the access data for the access write operation
		boolean setTagRead(int tagRead)	(Not used. Set multibank Read/Write config in CS710)	set the number of extra banks read during normal/multi-bank inventory.
		boolean setRx000KillPassword(String password)	boolean setKillPassword(String password)	set the kill password
		boolean setRx000AccessPassword(String password)	boolean setAccessPassword(String	set the access password
		boolean setAccessRetry(boolean accessVerfiy, int accessRetry)	boolean setAccessRetry(boolean accessVerfiy, int accessRetry)	set the access retry parameters:
				accessVerfiy: need to verify or not after access
				accessRetry: number of retry if access failure
		boolean setAccessLockAction(int accessLockAction, int accessLockMask)	boolean setAccessLockAction(int accessLockAction, int accessLockMask)	set lock configuration for lock operation.
		void restoreAfterTagSelect()	void restoreAfterTagSelect()	restore user configured parameter after selected operation
		boolean setSelectedTagByTID (String strTagId, long pwrlevel)	(Not exposed to application)	set the selected tag to be selected or matched for the RFID operation with strTagId as the Tid string part.
		boolean setSelectedTag(String strTagId, int selectBank, long pwrlevel)	boolean setSelectedTag(String strTagId, int selectBank, long pwrlevel)	set the selected tag to be selected or matched for the RFID operation.
				Paramter:
				strTagId: the Epc of the matching tag.
				selectBank: the bank of the matching tag.
				Pwrlevel: power level
		boolean setMatchRep(int matchRep)	(No such feature in CS710S)	Set inventory repeat count for CS108
		boolean setSelectedTag(String selectMask, int selectBank, int selectOffset, long pwrlevel, int qValue, int	boolean setSelectedTag(String selectMask, int selectBank, int selectOffset, long pwrlevel, int	set the selected tag to be selected or matched for the RFID operation.
				Paramter:
				selectMask: the mask of the matching tag.
				selectBank: the bank of the matching tag

		selectBank, int selectOffset, long pwrlevel, int qValue, int matchRep)	selectBank, int selectOffset, long pwrlevel, int qValue, int matchRep)	selectOffset: the mask offset of matching tag. Pwrlevel: power level qValue: the qValue matchRep: the match repeat count
		boolean startOperation(OperationTypes operationTypes)	boolean startRfidInventory()	start RFID detector operation. Parameter: operationTypes: operation type
		(done by giving parameter in startOperation in CS108)	boolean startRfidSearching()	Start RFID searching
		(done by setting multibank related parameters before startOperation in CS108)	Library4A.MultiBankData startRfidInventoryWithMultiBank(Library4A. MultiBankData multiBankData, String mDid, boolean bMultiBank, boolean readTagFiltering)	Start RFID multibank inventory
		boolean abortOperation()	boolean abortRfidOperation()	stop the operation.
		(direct access invalidate in library)	void resetInvalidata()	Reset invalidate
		(direct access invalidate in library)	getInvalidata()	Get invalidate
		(direct access invalidate in library)	int getInvalidUpdata()	Get invalidUpdata
		(direct access validate in library)	int getValidata()	Get validate
		(No such feature in CS108)	long getTagRate()	Get tagRate
		(No such feature in CS108)	long getcrcErrorTagRate()	Get crcErrorTagRate
		boolean sendHostRegRequestHST_CMD(HostCommands hostCommand)	boolean sendHostRegRequestHST_CMD(HostCommands hostCommand)	set operation command to be executed in StartOperation command
		boolean starAuthOperation()	(No such feature now in CS710S)	start authentication operation.
		Rx000pkgData onRFIDEvent()	Library4A.UplinkPacket readCsReaderUplinkPacketDecoded()	Get new RFID inventoried tag information with output: null for nothing. Otherwise it returns the package data inventoried.
		Rx000pkgData onRFIDEvent()	UplinkPacket readUplinkPacket()	This is packet without decoding
		(the procedures are separated in the AsyncTask in CS108 main module)	ConnectorRfidReader.TagData decodeXerxesTagData(UplinkPacket uplinkPacket, String strMdid, Library4A.MultiBankData multiBankData)	Decode the uplink RFID data
		(No such feature in CS108)	void setRfidReaderDefault()	Set the default parameters in Settings
		(This is implemented in the CS108 main module)	void settagTypeExpected(TagTypes tagType)	Set the expected Tag Type for the inventory
		(This is implemented in the CS108 main module)	TagTypes gettagTypeExpected()	Get the expected Tag type set in the library
		(This is implemented in the CS108 main module)	void setselectFor(int selectFor)	Set the expected operation for the expected tag type for the inventory
		(This is implemented in the CS108 main module)	int getselectFor()	Get the expected operation for the expected tag type for the inventory
		(This is implemented in the CS108 main module)	void setmDid(String mDid)	Set no expected tag type for the inventory
		(SettingTask is in main module in CS108)	void startSettingTask(Context context, boolean sameSetting, String invalidRequest)	Start the Setting task
		(SettingTask is in main module in CS108)	boolean isSettingTaskRunning()	Check if the setting task is finished or not
		(SettingTask is in main module in CS108)	void stopSettingTask()	Stop the setting task
		(This is done in CS108 main module)	void resetSelectData()	Reset all tag select registers
		(This is done in setSelectCriteria)	void setselectHold(int selectHold)	Set the select hold time
		(No such operation in CS108)	boolean disableMultibankReadConfig()	Disable multibank registers
		boolean setPwrManagementMode(boolean bLowPowerStandby)	(No such thing in CS710S now)	Set low power mode in CS108
		(Direct mac write is used in CS108)	set em4245 commandConfig()	Set em4245 register value
		(done in AccessTask main module in CS108, no such control register in CS108)	boolean setKillPassword(String password)	Set the killPassword register in E710 Atmel controller
		(done in AccessTask main module in CS108, no such control register in CS108)	boolean setAccessPassword(String password)	Set AccessPassword register in E710 Atmel controller
		(done in AccessTask main module in CS108, no such control register in CS108)	setAccessRead(int accessBank, int accessOffset, int accessCount)	Set AccessRead register in E710 Atmel controller
		(done in AccessTask main module in CS108, no such control register in CS108)	boolean setAccessWrite(int accessBank, int accessOffset, int accessCount, String dataInput)	Set AccessWrite Register in E710 Atmel controller
Class moved from Application to library package in CS710S				
	AesCmac	AesCmac()	AesCmac()	initialisation
		void init(Key key)	void init(Key key)	Intitialize the data with key
		void updateBlock(byte[] data)	void updateBlock(byte[] data)	Update block with new bytes
		byte[] doFinal()	byte[] doFinal()	Finalize the block data with key
	CustomAlertDialog	CustomAlertDialog()	CustomAlertDialog()	initialisation
		boolean Confirm(Activity act, String Title, String ConfirmText, String OkBtn, Runnable okProcedure, String CancelBtn, Runnable cancelProcedure)	boolean Confirm(Activity act, String Title, String ConfirmText, String OkBtn, Runnable okProcedure, String CancelBtn, Runnable cancelProcedure)	Start the dialog