EMPRESS™ 2.4GHz Active RFID Reader

.NET Library API Guide

(Model: HKRAR-EM02 Series)

Revision: 2.0.0.1

Before use, please read these instructions completely

Disclaimer

The information and know-how included in this document are the exclusive property of Hong Kong RFID Limited and are intended for the use of the addressee or the user alone. The addressees shall not forward to another their right of using the information, know-how or document forwarded herewith, in whole or in part in all matters relating or stemming from or involved therein, where for consideration or with consideration, and shall not permit any third party to utilize the information, know-how or the documents forwarded herewith or copies or duplicated thereof, unless at the company's consent in advance and in writing.

Enterprise License

Important Notice

All statements, technical information, and recommendations related to Hong Kong RFID's products are based on information believed to be reliable, but the accuracy or completeness is not guaranteed. Before using this product, you must evaluate it and determine if it is suitable for your intended application. You assume all risks and liability associated with such use. Any statements related to the product which are not contained in HKRFID's current publications, or any contrary statements contained on your purchase order shall have no force or effect unless expressly agreed upon, in writing, by an authorized officer of Hong Kong RFID.

Contents

Empress™ .NET Development Overview

Overview

Empress

Architecture Overview

System / Software Requirements

Installation

HKRAREM Class Library

The HKRAREM class library is a library of classes, interfaces, and value types that provide access to Empress Active Reader system functionality. The namespaces and namespace categories in the class library are listed in the following table and documented in detail in this reference.

Namespaces

HKRFID.Active

Constructor

Name	Description
HKRAREM(String host, ConnectionType connectiontype)	Initialize a new instance of the HKRAREM class using the specified port/hostname and the connection type.

Properties

Name	Description
CommandTimeOutMs	Timeout value for commands in millisecond. Default value is 3000ms.
isConnected	Shows that is the connection is established.

Methods

(Tag Inventory)

Name	Description
int StartTagInventory(void)	Start the reader to capture tags around.
int StopTagInventory(void)	Stop the reader from capturing tags around.

int GetTagBuffer(void)	Get all the Tag data inside the reader buffer. This command valid in “Passive mode$_{1}$”.
int GetTagBufferAndClear(void)	Get and clear all the Tag data inside the reader buffer. This command valid in “Passive mode$_{1}$”.
int ClearTagBuffer(void)	Clear all the Tag data inside the reader buffer. This command valid in “Passive mode$_{1}$”.

(Tag Reader Setting)

Name	Description
int SetReaderChannel(ReaderChannel channel)	Set the reader channel to read different model of tags.
int SetReaderFrequency(ReaderFrequency frequency)	Set the reader frequency to read different model of tags.
int GetGain(void)	Get the gain value of the HKRAR-EM reader.
int SetGain(int gain)	Set the gain value of the HKRAR-EM reader.
ReadInterval GetReadInterval(void)	Get the sample rate of the reader.
int SetReadInterval(ReadInterval interval)	Set the sample rate of the reader.
int SetReaderMode(ReaderMode mode)	Set the reader mode
int SetRequestMode(RequestMode mode)	Set the reader request mode

(Reader Hardware Special Output/Input)

Name	Description
int SetReaderBuzzerAndLEDOn(void)	Set the reader LED flash and buzzer beep once. Only EMSP-S supported
int SetReaderBuzzerOn(void)	Set the reader beep sound. Only EMSP-S supported
int SetReaderLEDOn(void)	Set green LED flash once of the reader. Only EMSP-S supported
int SetGPIODirection(int mask, int direction)	EM02 have 3 GPIO for advance usage. This command can set the GPIO as input or output by user-define.
int Set GPIOValue(int mask, int value)	EM02 have 3 GPIO for advance usage. This command can set the GPIO value by user-define. Only output GPIO can be set value.

(Reader Advanced Setting)

Name	Description
string GetReaderVersion(void)	Get the reader version of the HKRAR-EM.
int GetSerialNumber(ref byte[] output)	Get the reader serial number that you are connected
ReaderStatus GetReaderStatus(void)	Get the reader status that you are connected
int SetReaderTime(DateTimelocalTime)	Set the time of the reader. The time should in Unix time format.
int SetReaderTime(uint time)	Set the time of the reader. The time should in Unix time format.
int SetHeartbeat(boolonoff)	Set the reader enable or disable heard beat function.

int ResetReader(void)	Reset Reader setting.

(Active Tag Control)

Name	Description
int SetTagLEDOn(byte[] tagid, int interval)	Trigger the target tag to turn on the LED for a specified interval.
int SetTagLEDFlash(byte[] tagid, int interval)	Triggers the target tag to flash the LED for a specified interval.
int BroadcastLEDFlash(void)	A broadcast command that triggers all the tags with LED to flash.
int ResetButtonLog(byte[] tagid)	Some of the Active tag(RT02) have button detect function. While the button was triggered (from Press to release / form release to Press), the button log will set to 1. If user needs to reset the button log to 0, user can call this function.

(Class and foundational)

Name	Description
void AsynConnect(void)	Connect the HKRAREM reader using Serial Port or TCP connection as the instance constructed.(Asyn)
void AsynDisconnect(void)	Disconnect the underlying connection of the HKRAREM reader instance.(Asyn)
double CalcHumidity_HT(byte[] data)	Calculate the Humidity of HT tag
double CalcPower_HT_PT(byte[] data)	Calculate the Humidity of HT tag
double CalcTemperature(byte[] data)	Calculate the Humidity of HT tag
double CalcTemperature_PT(byte[] data)	Calculate the Humidity of HT tag
double CalcTemperature_HT(byte[] data)	Calculate the Humidity of HT tag

void Connect(void)	Connect the HKRAREM reader using Serial Port or TCP connection as the instance constructed.
void Disconnect(void)	Disconnect the underlying connection of the HKRAREM reader instance.
string GetLibVersion(void)	Get the version of the API library.

Events

Name	Description
TagReturn	Represents the method that will handle the tag return event of a HKRAREM object.
NumOfTagReturn	Represents the method that will handle the NumOfTagReturn event of a HKRAREM object.

Details Information

Constructor

HKRAREM(string host, ConnectionType connectiontype)		
Initialize a new instance of the HKRAREM class using the specified port/hostname and the connection type.		
Parameters		
Name	Type	*Details*
host	System.String	(Serial port connection)The port to connect the reader (e.g., COM1) (TCP connection)The hostname to connect the reader (e.g., 192.168.1.254)
connectiontype	HKRFID.Active.HKRAREM.ConnectionType	Connection type of the HKRAREM. (e.g., SerialPort or TCP)

Method

void Connect()	
Connect the HKRAREM reader using SerialPort or TCP connection as the instance constructed.	
Exceptions	
Connection Type	Condition
SerialPort Connection	The port is in invalid state/an attempt to set the port state

	failed (e.g. a wrong comport is selected)
TCP Connection	The destination cannot be connected. (e.g. a wrong hostname is set / the port is unreachable)

void Disconnect()

Disconnect the underlying connection of the HKRAREM reader instance.

Exceptions

Exceptions will be thrown if the disconnect failures. Check exception message for more details.

string GetLibVersion()

Get the version of the API library.

Return Value

Type	Details
System.string	Reader Version, an empty string indicates a failure.

int SetReaderChannel(ReaderChannel channel)

Set the reader channel to read different model of tags.

Parameters

Name	Type	Details
channel	HKRFID.Active.HKRAREM.ReaderChannel	The reader channel of the reader.Either LX or EM. For NT02, the reader channel is EM.

Return Value

Type	Details
System.Int32	0 means successful. Return value smaller than 0 indicates an error; please refer the error code for more details.

int SetReaderFrequency(ReaderFrequency frequency)

Set the reader frequency to read different model of tags.

Parameters

Name	Type	Details
frequency	HKRFID.Active.HKRAREM.ReaderFrequency	The reader frequency of the reader. Either US or EU. For NT02, the reader channel is US.

Return Value

Type	Details

System.Int32	0 means successful. Return value smaller than 0 indicates an error; please refer the error code for more details.

int GetGain(void)

Get the gain value of the HKRAR-EM reader.

Return Value

Type	Details
System.Int32	The gain value ranged from 0 – 31, value smaller than 0 indicates an error, please refer the error code for more details.

int SetGain(int gain)

Set the gain value of the HKRAR-EM reader.

Parameters

Name	Type	Details
gain	System.Int32	The gain value to set for the HKRAR-EM reader, ranged from 0 to 31. Smaller the value, more sensitive is the reader. (i.e. 0 has the longest

		reading range, 31 has the shortest)

Return Value

Type	Details
System.Int32	0 means successful. Return value smaller than 0 indicates an error; please refer the error code for more details.

int StartTagInventory(void)

Start the reader to capture tags around. Tag read will be returned by a **Error! Reference source not found.**

Return Value

Type	Details
System.Int32	0 means successful. Return value smaller than 0 indicates an error; please refer the error code for more details.

int StopTagInventory(void)

Stop the reader from capturing tags around.

Return Value

Type	Details

System.Int32	0 means successful. Return value smaller than 0 indicates an error; please refer the error code for more details.

int GetTagBuffer(void)

Get all the Tag data inside the reader buffer. This command valid in “Passive mode Reader will return tag data according to its reader mode.

Return Value

Type	Details
System.Int32	0 means successful. Return value smaller than 0 indicates an error; please refer the error code for more details.

int GetTagBufferAndClear(void)

Get and clear all the Tag data inside the reader buffer. This command valid in “Passive mode$_{1}$”. Reader will return tag data according to its reader mode and clear the buffer.

Return Value

Type	Details
System.Int32	0 means successful. Return value smaller than 0 indicates an error, please refer the error code for more

	details.

int ClearTagBuffer(void)

Clear all the Tag data inside the reader buffer. This command valid in “Passive mode”.

Return Value

Type	Details
System.Int32	0 means successful. Return value smaller than 0 indicates an error; please refer the error code for more details.

int SetTagLEDOn(byte[] tagid, int interval)

Trigger the target tag to turn on the LED for a specified interval.

Parameters

Name	Type	Details
tagid	System.Byte[]	The byte array containing 8 bytes of the target tag ID.
interval	System.Int32	The interval in seconds of the light up time of the tag’s LED. The value should be ranged from 0 to 255.

Return Value

Type	Details
System.Int32	0 means successful. Return value smaller than 0 indicates an error; please refer the error code for more details.

int SetTagLEDFlash(byte[]tagid, int interval)

Triggers the target tag to flash the LED for a specified interval.

Parameters

Name	Type	Details
tagid	System.Byte[]	The byte array containing 8 bytes of the target tag ID.
interval	System.Int32	The interval in seconds of the total flash duration of the tag's LED. The value should be ranged from 0 to 255.

Return Value

Type	Details
System.Int32	0 means successful. Return value smaller than 0 indicates an error; please refer the error code for more details.

int BroadcastLEDFlash(void)

A broadcast command that triggers all the tags with LED to flash.

Return Value

Type	Details
System.Int32	0 means successful. Return value smaller than 0 indicates an error; please refer the error code for more details.

int ResetButtonLog(byte[]tagid)

Reset the “button statue changed” flag of the target tag.

Parameters

Name	Type	Details
tagid	System.Byte[]	The byte array containing 8 bytes of the target tag ID.

Return Value

Type	Details
System.Int32	0 means successful. Return value smaller than 0 indicates an error, please refer the error code for more details.

int SetReaderLEDOn()

Light up the reader's LED for a short interval.

Return Value

Type	Details
System.Int32	0 means successful. Return value smaller than 0 indicates an error, please refer the error code for more details.

int SetReaderBuzzerOn()

Beep the reader's buzzer for a short interval.

Return Value

Type	Details
System.Int32	0 means successful. Return value smaller than 0 indicates an error, please refer the error code for more details.

int SetReaderBuzzerAndLEDOn()

Light up the reader's LED and beep the buzzer for a short interval.

Return Value

Type	Details
System.Int32	0 means successful. Return value smaller than 0 indicates an error, please refer the error code for more details.

int SetGPIODirection(int mask, intdirectionValue)

EM02 have 3 GPIO for advance usage. This command can set the GPIO as input or output by user-define.

Parameters

Name	Type	Details
mask	System.int32	-- in bitmap format, 1: valid setting; 0 : ignore setting
directionValue	System.int32	-- in bitmap format, 0: input; 1: output Example: set pin1 as output, pin 2 as input, pin 3 unchanged

		Mask: 0x03 (aim to set Pin 1 and Pin 2) Direction: 0x01 (Pin 1: output, Pin 2: input)

Return Value

Type	Details
System.Int32	0 means successful. Return value smaller than 0 indicates an error, please refer the error code for more details.

intSetGPIOValue(int mask, int value)

EM02 have 3 GPIO for advance usage. This command can set the GPIO value by user-define. Only output GPIO can be set value.

Parameters

Name	Type	Details
mask	System.int32	-- in bitmap format, 1: valid setting; 0 : ignore setting
directionValue	System.int32	-- in bitmap format, 0: low; 1: high Example: set pin 1 to high, set pin 2 to low , pin 3 unchanged

		Mask: 0x03 (aim to set Pin 1 and Pin 2) Direction: 0x01 (Pin 1: high, Pin 2 : low)

Return Value

Type	Details
System.int32	0 means successful. Return value smaller than 0 indicates an error, please refer the error code for more details.

string GetReaderVersion()

Get the reader version of the HKRAR-EM.

Return Value

Type	Details
System.string	Reader Version, an empty string indicates a failure.

int GetSerialNumber(ref byte[] outputSerialNumber)

get the reader serial number that you are connected

Parameters

Name	Type	Details
outputSerialNumber	System.byte[]	the received serial number will be stone in outputSerialNumber

Return Value

Type	Details
System.int32	0 means successful. Return value smaller than 0 indicates an error, please refer the error code for more details.

ReaderStatus GetReaderStatus()

Get the reader status of the HKRAR-EM.

Return Value

Type	Details
HKRFID.Active.HKRAREM.ReaderStatus	Command Timeout" Exception indicates a failure.

int SetReaderTime(DateTimelocalTime)

Set the time of the reader. The time should in Datetime format.

Parameters

Name	Type	Details
localtime	System.DateTime	user input DateTime object

Return Value

Type	Details
System.int32	0 means successful. Return value smaller than 0 indicates an error, please refer the error code for more details.

int SetHeartBeat(bool onoff)

Set the reader enable or disable heard beat function. When enable heart beat, Reader will return its serial number as heart beat in each interval. Heart beat interval is 10 times of sample time.

Parameters

Name	Type	Details
onoff	System.bool	heart beat : enable : 01 , disable : 00

Return Value

Type	Details
System.int32	0 means successful. Return value smaller than 0 indicates an error; please refer the error code for more details.

int ResetReader()

Reset the Reader to the manufacture setting.

Return Value

Type	Details
System.int32	0 means successful. Return value smaller than 0 indicates an error, please refer the error code for more details.

int SetReadInterval(int interval)

Set the reading interval of the reader, the smaller the interval, the faster the reader will return tag(s) inside the buffer.

Parameters

Name	Type	Details
interval	System.Int32	The reading interval of the reader.

Return Value

Type	Details
System.Int32	0 means successful. Return value smaller than 0 indicates an error, please refer the error code for more details.

int GetReadInterval()

Set the reading interval of the reader, the smaller the interval, the faster the reader will return tag(s) inside the buffer.

Return Value

Type	Details
System.Int32	0 means successful. Return value smaller than 0 indicates an error, please refer the error code for more details.

int SetReaderMode(ReaderMode mode)

Set the reader mode

Parameters

Name	Type	Details
mode	HKRFID.Active.HKRAREM.ReaderMode	The mode type Classic/EM02Mode.

Return Value

Type	Details
System.Int32	0 means successful. Return value smaller than 0 indicates an error, please refer the error code for more details.

int SetRequestMode(RequestMode mode)

Set the reader request mode

Parameters

Name	Type	Details
mode	HKRFID.Active.HKRAREM.RequestMode	The mode type Active/Passive.

Return Value

Type	Details
System.Int32	0 means successful. Return value smaller than 0 indicates an error, please refer the error code for more details.

Events

TagReturn Event

Represents the method that will handle the tagreturn event of a HKRAREM object. A tag return event is raised for every tag read of the reader.

TagReturnHandler(object sender, HKRFID.Active.HKRAREM.TagReturnEventArgs e)

Parameters

Name	Type	Details
sender	System.Object	The sender of the event, which is the HKRAREM object.
e	HKRFID.Active.HKRAREM.TagReturnEventArgs	A TagReturnEventArgs that contains the event data which is the tag information that it returns.

Related Classes

TagReturnEventArgs

Properties

Name	Description
return_tag	The tag return by the event

NumOfTagReturnEventArgs

Properties

Name	Description
num_return_tag	The number of tag return by the event

Tag

Properties

Name	Description
tag_id	The byte array containing the tag_id
status	The tag status

TagStatus

Properties

Name	Description

signal_strength	1 for signal strength high 0 for signal strength
battery_status	1 for normal 0 for low battery
tag_id_length	number of bytes of the tag id
tag_type	the tag type of the tag
sensor_data	a byte array containing the sensor data (if available)
button_on_off	Instant button status (Specific tag only)
button_status_changed	1 if button status is changed after reset otherwise 0 (Specific tag only)
rssi	A integer type store RSSI value (EM02 only)
read_time	Datetime type which store the last seen time of tag (EM02 only)

ReaderStatus (EM02 ONLY)

Properties

Name	Description
Gain	The current gain value of the reader in Integer type.
Channel	The current channel value of the reader in ReaderChannel type.
Frequency	The current frequency of the reader in ReaderFrequency type.
ReadInterval	The current read interval of the reader in ReaderInterval type.
IsReading	Boolean type represent reading tag or not.
ReaderMode	The current Read Mode of the reader in ReaderMode type.
RequestMode	The current Request Mode of the reader in RequestMode type.
GPIODirection	An Integer represent GPIO direction
GPIOValue	An Integer represent GPIO value
ReaderTime	A DateTime store reader time.
HeartBeatEnabled	A Boolean which represent Heartbeat enable or not

string ToString(void)	A method which can output a string with all current reader status in the following form: “Gain: X Channel:X Frequency:X Interval:X Reading:X Reader Mode: X Request Mode:X GPIO: ... Reader Time: X Heartbeat: X

Enumerations

ConnectionType

Member name	Description
SerialPort	RS232 connection type
TCP	Ethernet connection type

ReaderChannel

Member name	Description
LX	
EM	Default channel for NT-02 series.

ReaderFrequency

Member name	Description
US	Default frequency for NT-02 series.
EU	

ReadInterval

Member name	Description
Instant	As fast as possible
Short	A faster interval (~1s)

Long	A standard interval (2s)
Undefined	An error read/timeout/wrong setting

TagType

Member name	Description
HKRAT_NT02	
HKRAT_NT02_Plus	
HKRAT_RT0x	
HKRAT_TT01	
HKRAT_TT01S	
HKRAT_RT02	
HKRAT_TT02	
HKRAT_TT02X	
HKRAT_ZT02	
HKRAT_CT02	
HKRAT_AT01	
HKRAT_PT02	
HKRAT_HT02	
HKRAT_ZT01	

HKRAT_ET02	
RF_Recorder	
Others	
Unknown	

ReaderMode (EM02 ONLY)

Member name	Description
EM02Mode	Default
Classic	

RequestMode (EM02 ONLY)

Member name	Description
Active	Default
Passive	

GPIOPort (EM02 ONLY)

Member name	Description
PIN1	
PIN2	
PIN3	

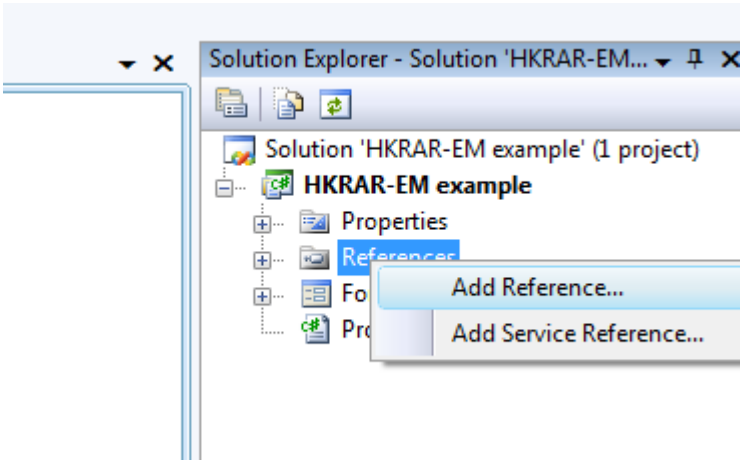
Error Codes

For some of the commands there will be a return value that contains an error code. This table describes the error returned by the reader.

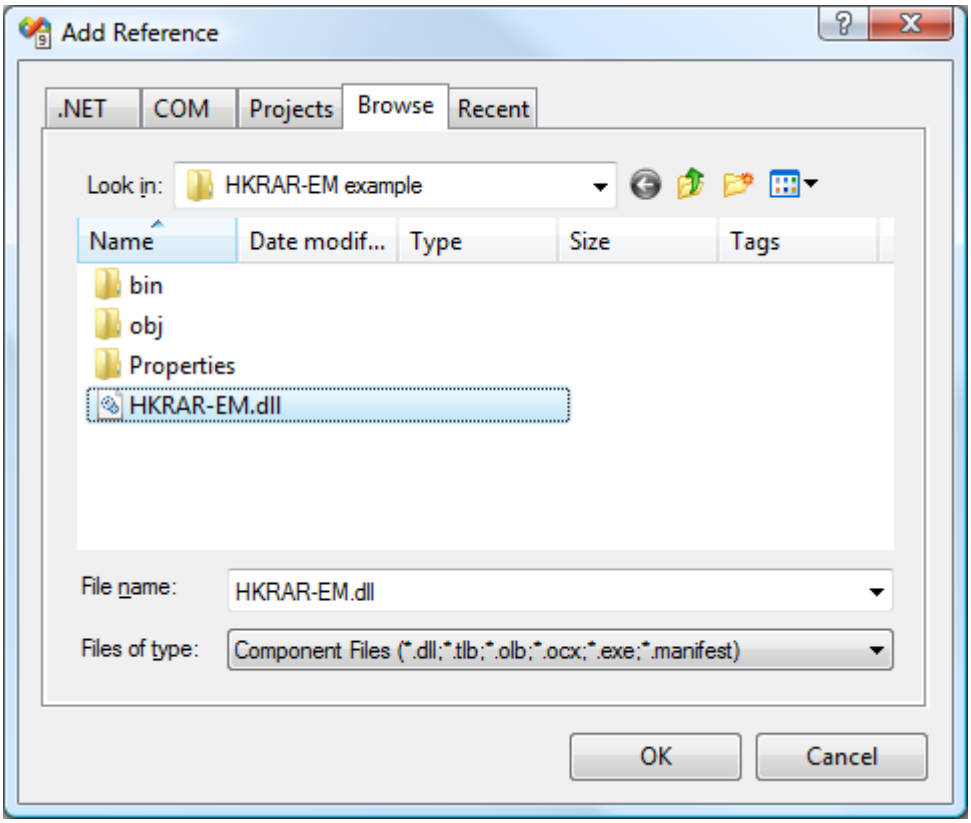
Value	Description
-1	Command error
-2	Command timeout
-3	Wrong parameters

Guideline to use the HKRAREM API in Visual Studio

1. In the visual studio, select Add Reference to add reference to the project.



2. Select the HKRAR-EM.dll which is the reader library for HKRAR-EM reader.



3. Include the namespace HKRFID.Active and you can start using the class HKRAREM. For methods in the API, please refer to the API document.

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;
using HKRFID.Active;

namespace HKRAR_EM_example
{
    public partial class Form1 : Form
    {
        HKRAREM theReader;
        public Form1()
        {
            InitializeComponent();

            theReader = new HKRAREM("192.168.12.2", HKRAREM.ConnectionType.TCP);
            theReader.Connect();
            theReader.StartTagInventory();
        }
    }
}
```

include the namespace

Start using

4. To handle the tag return event when a tag is received.

   Set up the event handler as follow, visual studio will helps to generate the event handler function after you press TAB.

```
theReader.TagReturn +=
                    new EventHandler<HKRAREM.TagReturnEventArgs>(theReader_TagReturn);  (Press TAB to insert)
```

5. You can access the tag details in the generated event handler.

```
void theReader_TagReturn(object sender, HKRAREM.TagReturnEventArgs e)
{
    e.return_tag.
}
```

Equals
GetHashCode
GetType
status
tag_id
ToString

byte[] Tag.tag_id

(End of Document)