LICENSE AGREEMENT
For

RemoteTele Pager

IMPORTANT – PLEASE READ CAREFULLY:

This License Agreement ("Agreement") is a legally binding agreement between you and Decision Computer International Co., Ltd. ("Decision") for the RemoteTele Pager software identified above ("Software"), unless you have a signed license agreement with Decision in other forms. By downloading, installing, copying or otherwise using the Software, you agree to be bound by the terms of this Agreement. If you do not agree to the terms of this Agreement, you shall NOT download, install, copy or otherwise use the Software.

The Software is protected by copyright laws and international copyright treaties, as well as by other intellectual property laws and treaties. The Software is licensed, not sold.

1. Grant of License

Subject to the terms and conditions of this Agreement, Decision grants to you a non-exclusive license to:

a. Install and use the Software on a hard disk or other storage device, such as network server.

b. Make and distribute unlimited copies of the Software, including copies for bundling with the commercial distribution of other software, provided that each copy that you make or distribute shall contain this Agreement and the copyright and other proprietary notices that appear in the Software, and further provided that any commercial distribution of the Software or any other software in bundles shall obtain Decision's written consent in advance.

2. Termination

a. Decision expressly reserves the right to terminate, at its sole discretion and without prior notice, the license hereby granted to you. Upon Decision's termination of the license, all your rights and licenses shall be revoked forthwith.

b. The license granted to you under this Agreement shall automatically terminate upon any failure of you to comply with any of the terms of this Agreement.

3. Limitations and Restrictions

You may not reverse engineer, decompile, disassemble or otherwise attempt to discover the source code of the Software. In addition, you may not use the Software to develop, adapt, or otherwise modify another OCX-type software.

4. Copyright and Trademark Rights

All title and copyrights in and to the software and its documentation are owned by Decision and protected by copyright laws and international treaty provisions. Decision and RemoteTele Pager are registered trademarks of Decision, you may use the trademarks to identify that you use the software, but such use does not give you any rights of ownership in the Software and its trademarks.

5. No Warranty

THE SOFTWARE IS LICENSED TO YOU AS IS. DECISION DISCLAIMS ALL WARRANTIES, EITHER EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE PERFORMANCE OR RESULTS THAT YOU MAY OBTAIN BY USING THE SOFTWARE, NON-INFRINGEMENT OF THIRD PARTY RIGHTS. MERCHANTIBILITY, OR FITNESS FOR ANY PARTICULAR PURPOSE WITH RESPECT TO THE SOFTWARE.

TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW, UNDER NO CIRCUMSTANCES SHALL DECISION BE LIABLE TO YOU FOR ANY CONSEQUENTIAL, INCIDENTAL, INDIRECT, OR SPECIAL DAMAGES WHATSOEVER ARISING OUT OF OR IN CONNECTION WITH THE USE OF THE SOFTWARE, INCLUDING BUT NOT LIMITED TO, LOSS OF BUSINESS PROFITS, BUSINESS INTERRUPTION, LOSS OF BUSINESS INFORMATION, OR ANY CLAIM BY ANY THIRD PARTY.

6. Governing Law

This Agreement is governed by the laws of the Republic of China, Taiwan, without regard to its conflict rules of laws.

YOUR ACCEPTANCE OF THIS AGREEMENT WILL BE INDICATED ONCE YOU PROCEED TO DOWNLOAD OR INSTALL THE SOFTWARE.

BeServer
Sets the DecisionRemoteCom OCX to Server, which does not allow any configuration. It will set us default on the property described.

Syntax
Object.*BeSever*

The **BeServer** property syntax has this part:

Part	**Description**
Object	An expression that evaluates to an object

Remarks
You can set OCX to server directly, during this setting a follow up for **PortLocal** which require the user to input Server Setup Local Port. Terminal Type for server must define as a Server.

Warning
Make sure that the **PortLocal** is not use by other device, Port Local 1024 is a default for the RemoteCOM, you may see local port information for sure.

Sample Program
Call Object.BeServer ‘define object has set to configure as Server

CliGetNumberOfPort
Get number of port, view available port from the server.

Syntax
Object.*CliGetNumberOfPort*

The **CliGetNumberOfPort** property syntax has this part:

Part	Description
Object	An expression that evaluates to an object

Remarks
To show or display the available port, user may use **GetNumberOfPort** to store and ready to display. **GetNumberOfPort,** a method that evaluate available port in the Server, then store to **CliGetNumberOfPort.**

Sample Program

```
Call Object.CliGetNumberOfPort     'define an object to store the available port from the
                                   server
```

CliGetvReturn
Get return of Pager functions

Syntax
objec*t.CliGetvReturn*

The **CliGetvReturn** property syntax has these parts:

Part	Description
Object	An expression that evaluates to an object

Data Type
String

Sample Program

```
Call Object.CliGetvReturn     ‘object stores the value done by the CliGetsuReturn

MyNumPort  =  Object.CliGetvReturn
```

Connect
Sets Port and Host IP function, to identify connection. It is a direct setup to configure the connection between the client and the server.

Syntax
Object.*Connect*

The **Connect** property syntax has this part:

Part	**Description**
Object	An expression that evaluates to an object

Remarks
During operation **PortRemote** and **HostRemote**(also know as IP address) is required, to allow and identify which Port to use and which Host computer to communicate. To view connection status, add status function.

Sample Program
```
Call Object.Connect      'object has request to connect to the server.
```

ConnectStatus
Returns the state of the control, expressed as an enumerated type. Read-only and unavailable at design time. To view the current status of the operation between the client and server.

Syntax
Object.*ConnectStatus*

The **ConnectStatus** property syntax has this part:

Part	Description
Object	An expression that evaluates to an object

Settings
The settings for the **ConnectStatus** property are:

#	Value	Description
0	Closed	Default. Closed is the status
1	Open	Open is the status
2	Listening	Listening is the status
3	Connection pending	Connection pending is the status
4	Resolving host	Resolving host is the status
5	Host resolved	Host resolved is the status
6	Connecting	Connecting is the status
7	Connected	Connected is the status
8	Peer is closing the connection	Peer is closing the connection is the status
9	Error	Error is the status

Return Value
Integer

Sample Program
```
Call Object.ConnectStatus     'object has request to show the status of the current
                              operation.
```

DisConnect
Sets function to disconnect communication between the Server and the Client.

Syntax
object.*DisConnect*

The **DisConnect** property syntax has this part:

Part	Description
Object	An expression that evaluates to an object

Remarks
After setting up connection to client or to server, user can disconnect communication, data or information will no longer be attend.

Sample Program
```
Call Object.DisConnect          'object has set to disconnect communication
```

GetConnectionStatus
Get connection status

Syntax
Object.*GetConnectionStatus*

The **GetConnectionStatus** property syntax has this part:

Part	**Description**
Object	An expression that evaluates to an object

Data Type
String

Sample Program

```
Call Object.GetConnectionStatus     'object stores the value done by
                                    GetConnectionStatus

MyNumPort  =  Object.GetConnectionStatus
```

GetNumberOfPort
A value or group of value which store all available remote communication port.

Syntax
Object*.GetNumberOfPort*

The **GetNumberOfPort** property syntax has this part:

Part	**Description**
Object	An expression that evaluates to an object

Sample Program

```
Call Object.GetNumberOfPort          'object get the available port from the server
```

HostRemote
Identify the IP address for communication with server. An error occurs when the client has not properly connected with the remote server. Returns or sets the remote machine to which a control sends or receives data. You can either provide a host name, for example, "FTP://ftp.microsoft.com," or an IP address string in dotted format, such as "100.0.1.1".

Syntax
Object.*HostRemote*

The **HostRemote** property syntax has this part:

Part	**Description**
Object	An expression that evaluates to an object

Remarks
When this property is specified, the **URL** property is updated to show the new value. Also, if the host portion of the URL is updated, this property is also updated to reflect the new value.

Return Value
String

Sample Program

```
Call Object.HostRemote          'assign to object the IP address to connect with the
                                server.
```

Name
Identify the OCX tag name

Syntax
Object.*Name*

The **Name** property syntax has this part:

Part	Description
Object	An expression that evaluates to an object

Sample Program

```
Call Object.Name            ‘assign the name to an object
```

PagerComPort(sCOMPort As String)
This function informs the OCX on which COM port the modem is connected.

Syntax
object.*PagerCOMPort(“Com Port”)*

The **PagerCOMPort** property syntax has these parts

Part	**Description**
Object	An expression that evaluates to an object.
Method	A method expression representing the communications to Open Port number.
Com Port	Specify what communication port

Sample Program
Call object.PagerCOMPort ("COM3”)

PagerDataToSend(sDataToSend As String, lDelay As Long)
This function is used for specifying the Actual Data to be sent to the Pager.

Syntax
object.*PagerDatatoSend(“Data to Send”,Delay)*

The **PagerDatatoSend** property syntax has these parts

Part	**Description**
Object	An expression that evaluates to an object.
Method	A method expression representing the communications to Open Port number.
Data to Send	Specify what data to send
Delay	Delay between the Telephone Number and the Actual Data. The delay is in seconds.

Sample Program
Call object.PagerDatatoSend ("1111111", 4)

PagerDialCode(sCountryCode As String, sAreaCode As String, lDelay As Long)
This method is used for specifying the Country and Area Code to be used in calling the Pager.

Syntax
object.*PagerDialCode(“Country Code”,” Area Code”,Delay)*

The **PagerDialCode** property syntax has these parts

Part	**Description**
Object	An expression that evaluates to an object.
Method	A method expression representing the communications to Open Port number.
Country Code	Specify country code
Area Code	Area code
Delay	Delay needed between Country and Area Code, the delay is in seconds

Sample Program
Call object.PagerDialCode ("63","2", 3);

PagerDialUsing(sToneOrPulse As String, lDelay As Long)
This function presets the OCX to use either “Tone” or “Pulse” signals during dialing.

Syntax
object*.PageDialUsing (“Tone or Pulse”,Delay)*

The **PagerDialUsing** property syntax has these parts

Part	**Description**
Object	An expression that evaluates to an object.
Method	A method expression representing the communications to Open Port number.
Tone or Pulse	Tone or Pulse dialing
Delay	Specifies the Delay needed for each digit of the Phone Number. The delay is in seconds.

Sample Program
Call object.PagerDialUsing ("Pulse", 3)

PagerSend
This function initiates the actual calling and sending of data to the Pager. Be sure to process first the CallPagerCOMPort, CallPagerPassword, CallPagerTelNo and CallPagerDatatoSend methods, before issuing this method.

Syntax
object.*PageSend*

The **PagerSend** property syntax has these parts

Part	**Description**
Object	An expression that evaluates to an object.
Method	A method expression representing the communications to Open Port number.

Sample Program
Call object.PagerSend

Result Values
0-OK
1-No such COM Port
2-Not a DECISION COM Port
3-Wrong OCX Password
4- No Answer
6- No Dialtone
7- Busy

PagerTelNo(sTelephoneNumber As String, lDelay As Long)
This function is used for specifying the Telephone Number that would be dialed.

Syntax
object*.PagerTelNo(“Telephone Number”,Delay)*

The **PagerTelNo** property syntax has these parts

Part	**Description**
Object	An expression that evaluates to an object.
Method	A method expression representing the communications to Open Port number.
Telephone #	Specify telephone number
Delay	Specifies the Delay needed between Area Code and Telephone Number. The delay is in seconds.

Sample Program
Call object.PagerTelNo ("9221299", 3)

PortLocal
Returns or sets the Port Local number to specify certain communication.

Syntax
object*.PortLocal = port*

The **PortLocal** property syntax has this part:

Part	Description
Object	An expression that evaluates to an object
Port	The port to connect to. The default value of this property is 1024

Remarks
The **PortLocal** property is set automatically to the appropriate default port for each protocol. Default port numbers are shown in the table below:

Port	Description
1024	Commonly used for World Wide Web connections of Decision Card.
21	FTP.

Sample Program
Call Object.PortLocal = “1024” ‘object set the port local to 1024

PortRemote
Returns or sets the Port Remote number for specific port

Syntax
object*.PortRemote = port*

The **PortRemote** property syntax has this part:

Part	**Description**
Object	An expression that evaluates to an object
Port	The port to connect to. The default value of this property is 1024

Remarks
The **PortRemote** property is set automatically to the appropriate default port for each protocol. Default port numbers are shown in the table below. Server and Client must have the same Port.

Port	**Description**
1024	Commonly used for World Wide Web connections of Decision Card.
21	FTP.

Sample Program
```
Call Object.PortRemote =  “1024”       ‘object set the port remote to 1024
```

SetConnect
For client use, sets connect to server who required the IP address and the PortRemote. During the operation server and the client must have same PortRemote and also client must know the IP address of the server.

Syntax
Object.*SetConnect*

The **SetConnect** property syntax has this part:

Part	Description
Object	An expression that evaluates to an object

Sample Program

```
Call Object.SetConnect                ‘set the object for connection
```

SetPropertyHostRemote(sIPAddress As String)
Returns or sets the remote machine to which a control sends or receives data. You can either provide a host name, for example, "FTP://ftp.microsoft.com," or an IP address string in dotted format, such as "100.0.1.1".

Syntax
object*.SetPropertyHostRemote(“IP Address”)*

The **SetPropertyHostRemote** property syntax has these parts

Part	**Description**
Object	An expression that evaluates to an object.
Method	A method expression representing the communications to Open Port number.
IP Address	Specify IP address of the server

Sample Program
Call object.SetPropertyHostRemote("202.65.115.98”)

SetPropertyPassword(sThePassWord As String)
A method that make the OCX use

Syntax
Object.*SetPropertyPassword(sThePassword As String)*

The **SetPropertyPassword** property syntax has this part:

Part	**Description**
Object	An expression that evaluates to an object
SthePassword As String	Declaration of the OCX password

Remarks
Decision Computer International Co., LTD. OCX password, trademark of the company. In order to run the application program, protection for any pirating act.

Default Password
Decision Computer

Data Type
String

Sample Program
Call Object.SetPropertyPassword(“Decision Computer”) ‘object remcom has set OCX password

SetPropertyPortLocal(lPortLocal As Long)
Returns or sets the Port Local number to specify certain communication.

Syntax
object.*SetPropertyPortLocal(lPortLocal As Long)*

The **SetPropertyPortLocal** property syntax has this part:

Part	**Description**
Object	An expression that evaluates to an object
LPortLocal As Long	The port to connect to. The default value is 1024

Remarks
Default port numbers are shown in the table below:

Port	**Description**
1024	Commonly used for World Wide Web connections of Decision Card.
21	FTP.

Sample Program
Call Object.SetPropertyPortLocal(“1024”) ‘object set the port local to 1024

SetPropertyPortRemote(lPortRemote As Long)
A method that returns or set the Port Remote number for specific port

Syntax
object.*SetPropertyPortRemote(lRemotePort As Long)*

The **SetPropertyPortRemote** property syntax has this part:

Part	**Description**
Object	An expression that evaluates to an object
LremotePort As Long	Specify the remote port to use

Remarks
Default port numbers are shown in the table below. Server and Client must have the same Port.

Port	**Description**
1024	Commonly used for World Wide Web connections of Decision Card.
21	FTP.

Sample Program
Call Object.SetPropertyPortRemote(“1024”) ‘object set the port remote to 1024

SetPropertyTerminalType(sTerminalType As String)
Specify the terminal type.

Syntax
Object.*SetPropertyTerminalType(sTerminalType As String)*

The **SetPropertyTerminalType** property syntax has this part:

Part	**Description**
Object	An expression that evaluates to an object
STerminalType As String	The type of terminal

Terminal Type

Part	**Description**
Client	Serve as a client
Server	Serve as a server

Sample Program
Call Object.SetPropertyTerminalType(“Client”) ‘object has set terminal to client

SetServer
It is a form of a dialog which user can sets the DecisionRemoteCom OCX to Server, which does allow configuration like the PortLocal.

Syntax
Object.*SetSever*

The **SetServer** property syntax has this part:

Part	**Description**
Object	An expression that evaluates to an object

Remarks
You can set OCX to server directly, during this setting a follow up for **PortLocal** which require the user to input Server Setup Local Port. Terminal Type for server must define as a Server.

Warning
Make sure that the **PortLocal** is not use by other device, Port Local 1024 is default for the RemoteCOM, and you may see local port information for sure.

Sample Program
Call Object.SetServer ‘define object has set to configure as Server

TerminalType
A property identifies which terminal type.

Syntax
Object*.TerminalType [= value]*

The **TerminalType** property syntax has this part:

Part	**Description**
Object	An expression that evaluates to an object
Value	Identify terminal Type

Terminal Type

Part	**Description**
Client	Serve as a client
Server	Serve as a server

Sample Program
Call Object.TerminalType = “Client” ‘object has set terminal to client