

Smart Label Programming Guide

Understand ISO/IEC15693
RWM600x ActiveX Control
GNetPlus Protocol Command Set

REV.G
October 7, 2008

Table of Contents

Install ISO15693 Demo Software	3
Understand ISO15693	4
How to configure RWM600A?	7
How to connect Multi Reader?	9
Programming Guide	10
RWM600 function block.....	10
Commands Table.....	11
Commands Description.....	13
ANNEX A. Error Code.....	42
ANNEX B. AFI Coding	43
ANNEX C. GNetPlus Protocol (ASCII Format).....	44
ANNEX D. Supported Tag	45
ANNEX E. History.....	48

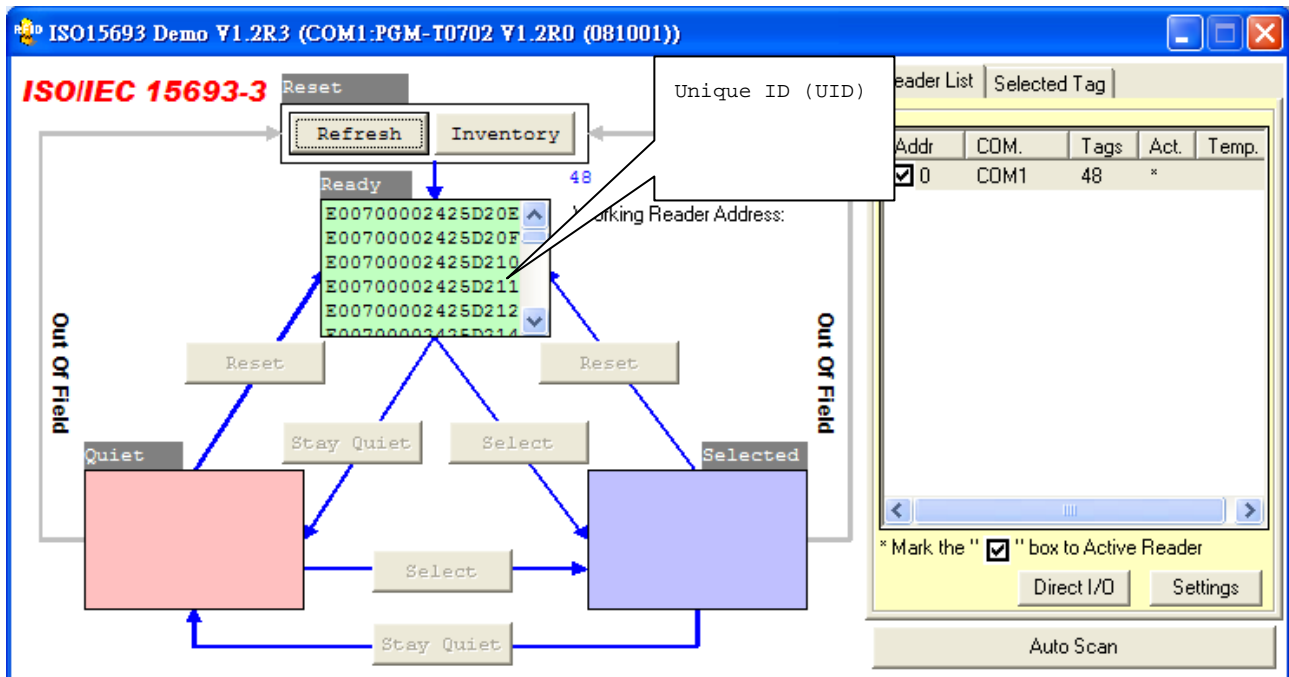
Install ISO15693 Demo Software

Setup:

1. Install the ISO15693 Demo Software (Included in DISK5217)
2. Connect your Smart Label Reader (RWM600SK...etc) to your PC Port.
3. Run ISO15693 Demo Software, and click the [Auto Scan] to auto connect the reader.
4. Mark the Addr check box to activize reader.

Example:

Put one or more Smart Labels (Maximum 5 pcs for RWM600SK) on reader antenna field. Example as below:



Note:

January 12, 2006: New "Settings" button in Version V1.1R0 or later for RWM600A.

March 28, 2007: New "EAS" button in Version V1.1R2 or later for RWM600A.

Understand ISO15693

Introduction

ISO/IEC 15693 is one of a series of International Standards describing the parameters for Smart Label (identification cards), and the use of such Smart Label for international interchange.

A Smart Label can be in one of the 4 following states:

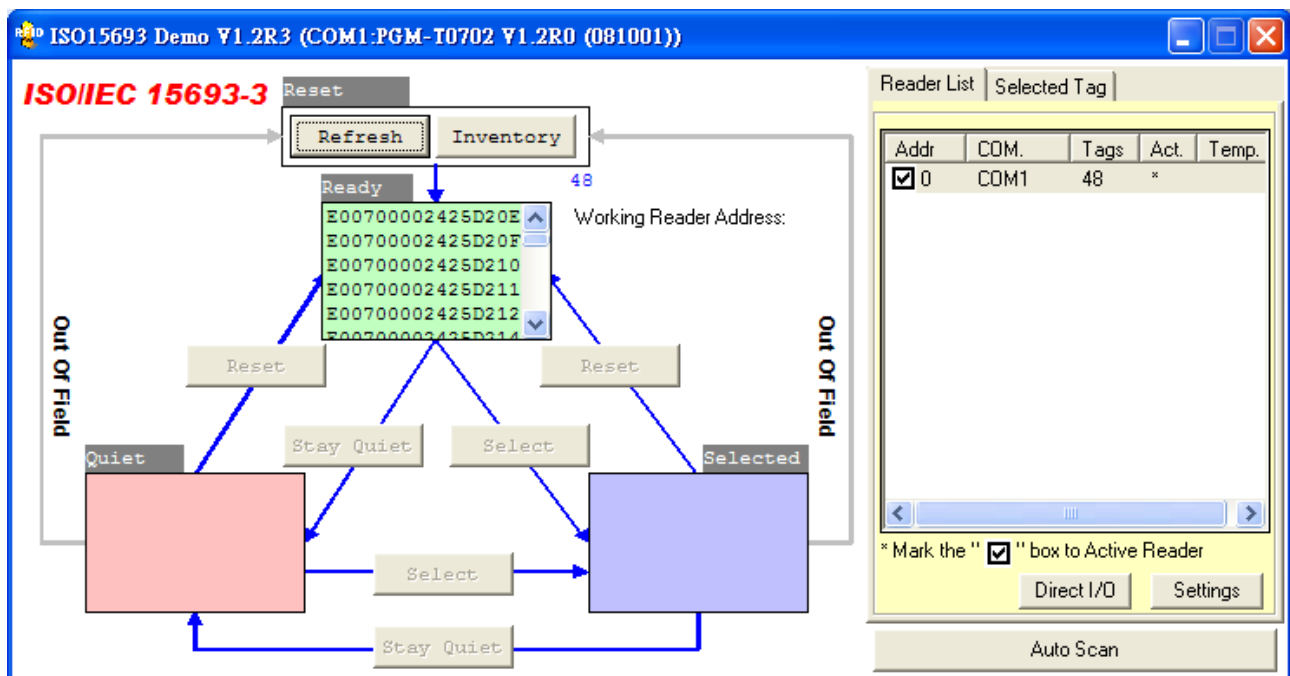
1. Power-off
2. Ready
3. Quiet
4. Selected

Power-off

A Smart Label would lose power supply with it was removed from the reader antenna field, the Smart Label without power supply is in the Power-Off state and it can not accept any Request Commands.

Ready State

While the Smart label gets into the Reader Antenna field it would get power supply. At this point, the Smart Label with power supply is in the Ready state, and it can accept any Request Commands from the reader. As the following figure, the three pieces of Smart Labels are in the Reader Antenna field and in the Ready state, the reader would read all these 3 pieces of Smart Labels with anti-collision technology and collect their Unique ID (UID) in these Smart Labels.

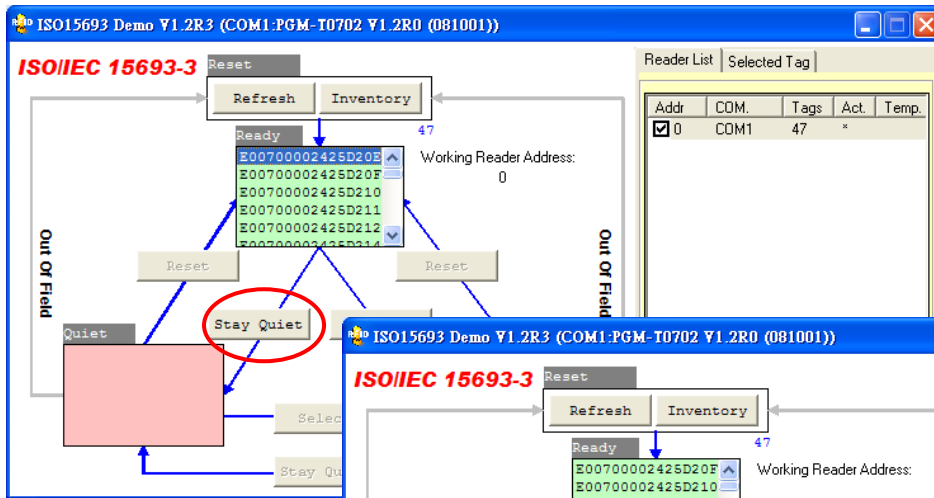


Remark:

Due to anti-collision technology, Reader may read 16 pieces of Smart Labels simultaneously, depending on the size of the antenna. For example, RWM600SK only can read up to 5 pieces of Smart Labels simultaneously because of the small size of the antenna.

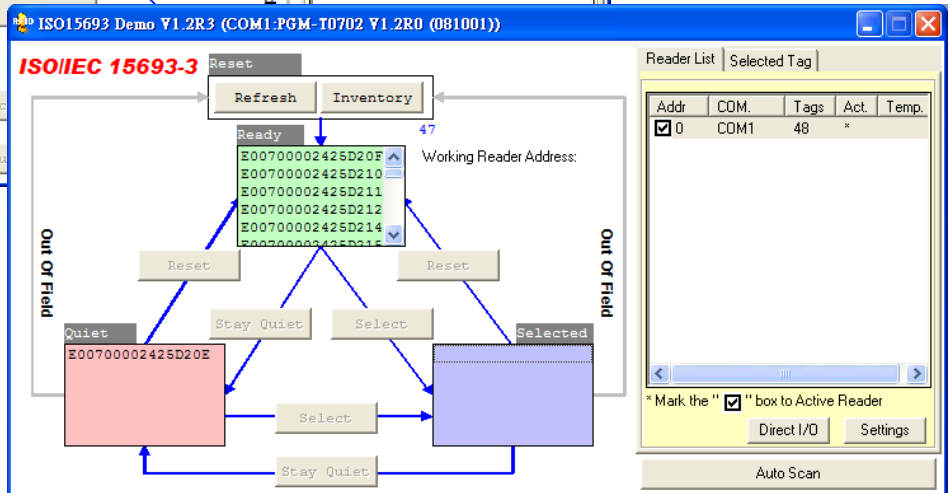
Quiet State

Reader can have the Smart Label become Quiet state by “StayQuiet” command while the Smart Label is within the reader antenna field. When the Smart Label is in the Quiet state, it can not respond the Reader until it is waken up. The Smart Label can be waken up by “ResetToReady” command or “Select” command.



Example: Select UID E00700002425CFA7 and click [Stay Quiet] to have it become Quiet state.

Note: By using Quiet state control, the reader can process some more other Smart Labels without



removing some Smart Labels from its antenna field when the quantity of the Labels in its antenna field is more than what the reader can process simultaneously. For example, RWM600SK only can process up to 5 labels simultaneously. But now there are 7 Smart Labels to be processed, the reader may process 4 Smart Labels first and then have the first 4 labels become Quiet state and then read the other 3 Smart Labels without removing the Smart Labels from its antenna field. After the second 3 Smart Labels is processed, you also may have the second 3 labels become Quiet state and go back to wake up the first 4 Smart Labels to process then again.

Selected State

In order to save the communication time of the Long command packet, ISO/IEC 15693 also have a Selected Mode to communication with the selected Smart Labels would be processed, you may use the “Select” command to select a certain UID label to be processed (like Read, Write, Lock ...etc), then all the Request Command packages communicated with the UID label are not necessary to include this UID. So the command packet is shorter and the communication time is shorter too, also save some communication error rate.

Example: Select UID E00700002425CFA9, and click [Select] to have this UID become Selected state.

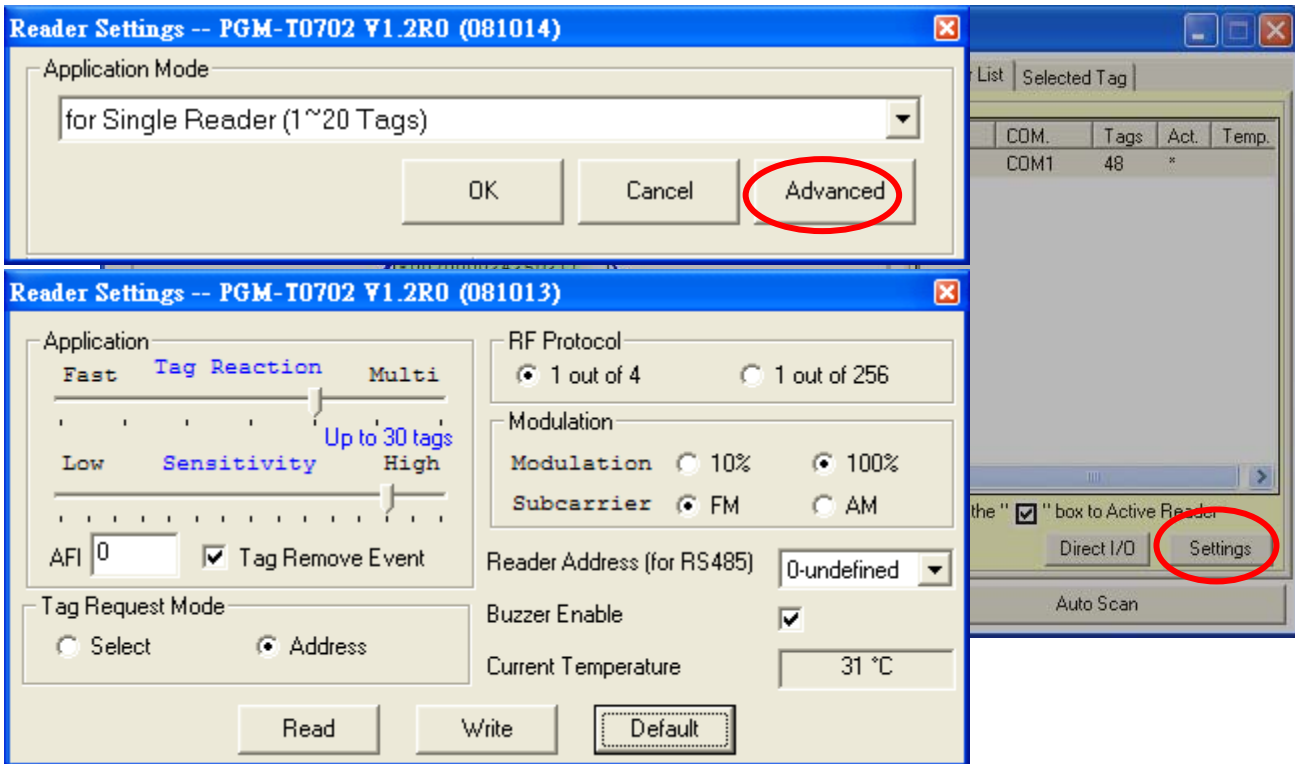
Remark: When select a new UID, the previously Selected UID will go back to Ready state automatically.

Symbols

- Unique identifier (UID):**
 The Smart Label are uniquely identified by a 64 bits unique identifier (UID). This is used for addressing each Smart Label uniquely and individually, during the anti-collision loop and for one to one exchange between a reader and a Smart Label.
- Application family identifier (AFI)**
 AFI represents the type of application targeted by the reader and is used to extract from all the Smart Labels present only the Smart Labels meeting the required application criteria. It may be programmed and locked by the respective commands. AFI is coded on one byte, which constitutes 2 nibbles of 4 bits each. The most significant nibble of AFI is used to code one specific or all application families.
- Data storage format identifier (DSFID)**
 The DSFID indicates how the data is structured in the Smart Label memory. It may be programmed and locked by the respective commands. It is coded on one byte. It allows for instant knowledge on the logical organization of the data. If its programming is not support by the Smart Label, the Smart Label shall respond with the value zero (00h).

How to configure RWM600A?

The RWM600A is configurable for reader parameters for your different applications, and all settings will save into EEPROM of RWM600A. Click "Settings" button on ISO15693 demo software:



Application Mode:

1. Mode
 - a. for **Security/for Multi Readers (1~10 Tags)**
Application for less than 10 tags and get tag quickly. Multi Readers application is suggested this mode.
 - b. for **Single Reader (1~20 Tags)**
Application for about 20 tags and normal using.
 - c. for **Inventory, without remove event (1~32 Tags)**
Application for about 32 Tags and don't cause remove event.
- For other application settings, please use "Advanced"

2. Change All Readers

More than one reader active in readers list; it will change settings of all reader at the same time.

3. OK/Cancel

Save settings or close this window.

Advanced:

1. **Fast and Multi: (for reading speed; Default = middle)**
 - a. If your application focus on security or access control, you may select the "Fast".
 - b. if your application focus on manage goods, you may select the "Multi".
 - c. If your application in both a and b, you may select speed in middle.
2. **Sensitivity: (for tag remove sensitivity; Default = 13)**
If set high sensitivity, the tag remove will immediate cause event. Otherwise, tag reomve event will delay

in low sensitivity status.

3. **AFI: (Application family identifier; Default = Zero; Value range=0~255)**

Reader AFI (not zero, enable) meet Tag AFI, the Tag will be read into RWM600A.

if Reader AFI is zero, the filter function will be disable and RWM600A read tags without AFI checking.

More AFI information, please see the "ANNEX B AFI Coding".

4. **Tag Remove Event**

When tag move out the antenna, the reader enabled/diabled send event to host.

If disabled "Tag Remove Event", reader only sent "Clear Event" when whole tags leave for antenna

5. **Tag Request Mode: (for ISO15693 protocol; Default = Select Mode)**

Adresse mode

In adresse mode, the request shall contain the unique ID (UID) of the addressed Tag.

Any Tag receiving a request with the Address-Mode shall compare the received unique ID (address) to its own ID.

If it matches, it shall execute it (if possible) and return a response to the Reader as specified by the command description.

If it does not match, it shall remain silent.

Select mode

In select mode, the request shall not contain a Tag unique ID.

The Tag in the selected state receiving a request with the Select-Mode shall execute it (if possible) and shall return a response to the Reader as specified by the command description.

Only the Tag in the selected state shall answer to a request.

6. **RF Protocol / Modulation: (Default = 1 out of 4,100%,FM)**

Please refer to ISO15693-3 Register Definition .

7. **Reader Address (for RS485 network; Default = Zero; Value range=0~32)**

You can connect many readers with RS485 network in your system after you update the Reader Address.

8. **Buzzer Enable: (Default=Enable)**

To disable the buzzer of reader, For example, your application for a quiet environment.

9. **Save and cancel the settings:**

If you want to cancel current settings, you can click the "Read" button to back to previous settings of RWM600A. If you ready and want to save the settings, you can click the "Write" button to save the settings into EEPROM of RWM600A.

10. **Default:**

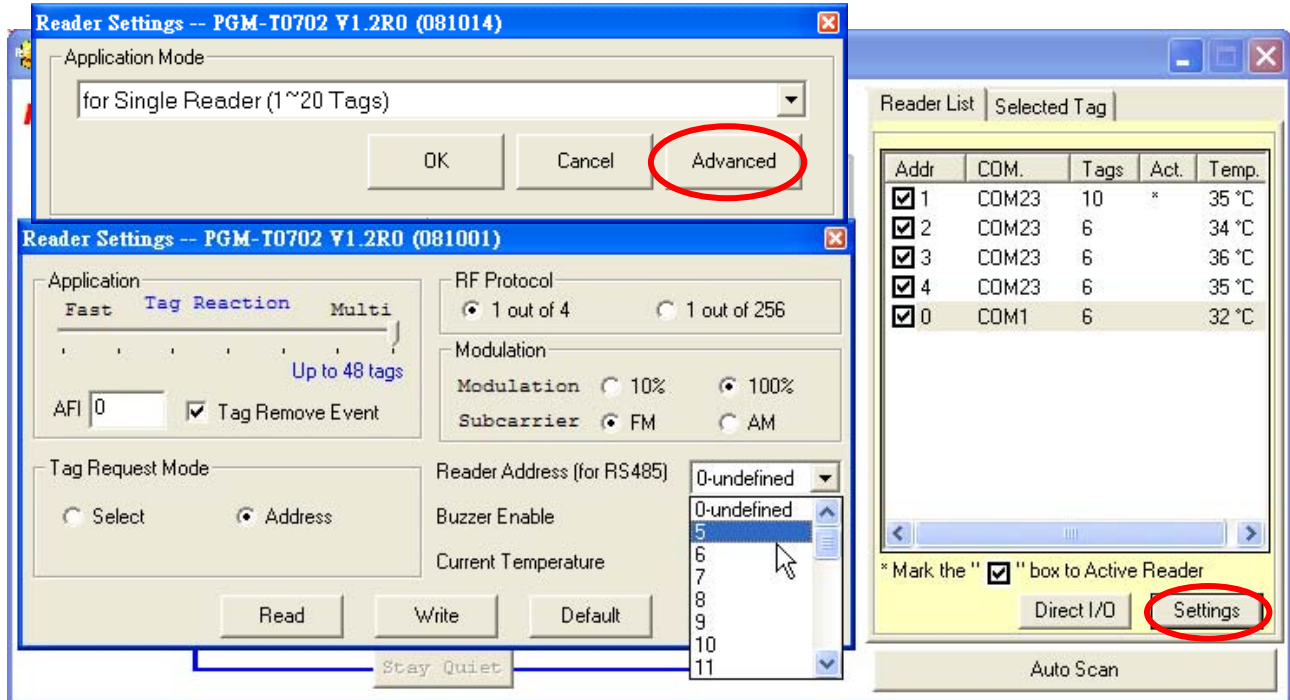
You can make all settings to default value. And click "Write" to save it.

How to connect Multi Reader?

At first time, connect one of all readers. And setup the reader address by click "Settings"->"Advanced".

And then connect second reader and click "AutoScan" to find it. To setup the different reader address number.

As the same way to setup the follow readers.



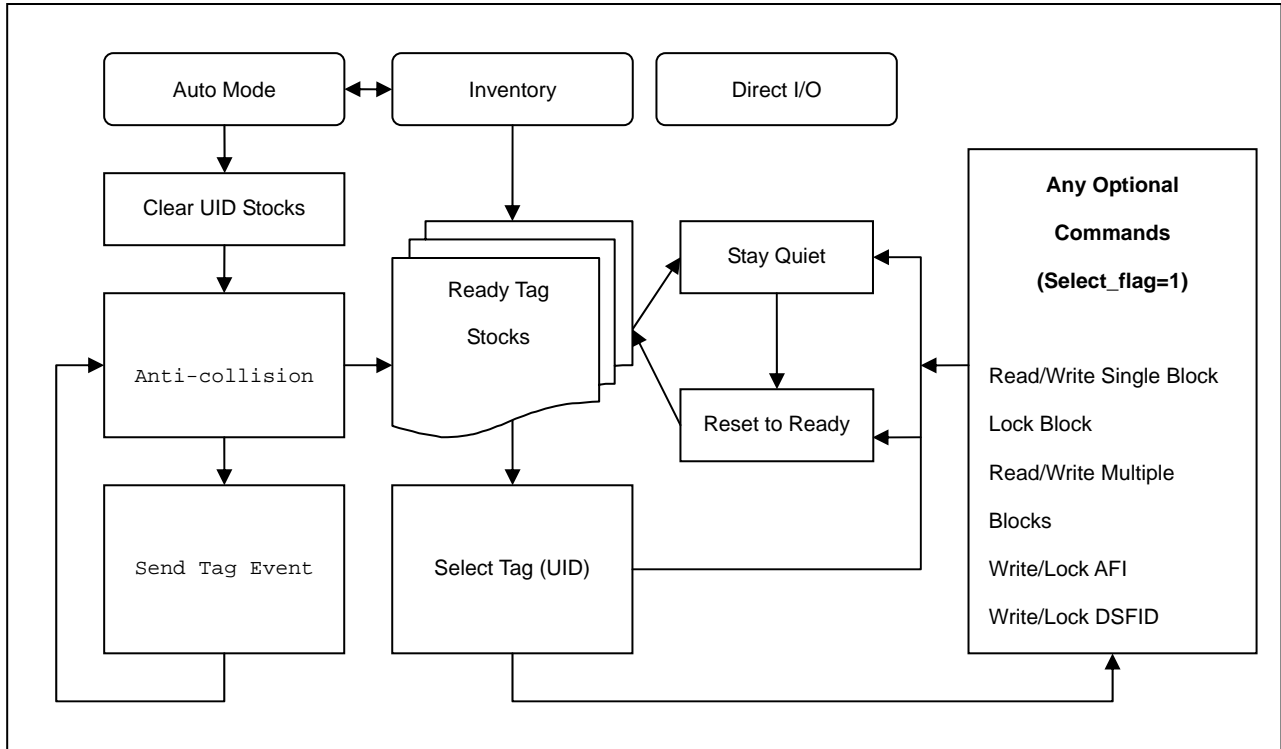
If you want to pause one reader, you just cancel the mark of addr column.

Remark:

1. Connect the reader with PC via USB485A.
2. Keep at least 10~15 cm between each antenna.
3. Keep other 13.56MHz products away.

Programming Guide

RWM600 function block



Auto Mode: (Anti-Collision)

When RWM600 is at "Auto Mode" (RWM600 in Auto Mode after power-on automatically), it would read the Smart Labels, which are within the reader antenna field, and collect those UID and save into the UID stocks, and then create Tag Event to inform the Host. When any Smart labels out of antenna field, it would also create Tag Event to inform the Host.

Command Mode: (Inventory , Direct I/O)

When RWM600 is executing "Inventory" or "Direct I/O", it would automatically stop Auto Mode. Then you need to use "Enable Auto Mode" command to go back to Auto Mode.

Commands Table

GNetPlus Query#	ActiveX Control (RWM600x.OCX)	Parameters	Response	Description
00h	gnetPolling	nAddr	Event	Polling all Reader
01	gnetVersion		Firmware Version	Get Firmware Version
3Fh	EnableAutoMode	Boolean	Boolean	Enable/Disable Auto Mode
40h	isoInventory	MaskLength MaskValue	Boolean DSFID UID	Inventory without AFI
40h	isoInventoryEx	AFI MaskLength MaskValue	Bolean DSFID UID	Inventory with AFI
41h	isoResetToReady	UID	Boolean	Reset to Ready
42h	isoStayQuiet	UID	Boolean	Stay Quiet
43h	isoSelect	UID	Boolean	Select UID
44h	isoGetSystemInformation		Boolean INFO	Get System Information
45h	isoReadSingleBlock	Block Number	Boolean Block Data	Read Single Block
46h	isoWriteSignalBlock	Block Number Block Data	Boolean	Write Single Block
47h	isoLockBlock	Block Number	Boolean	Lock Block
48h	isoReadMultipleBlock	Block Number Count	Count Blocks Data	Read Multiple Blocks
49h	isoWriteMultipleBlock	Block Number Count Blocks Data	Count	Write Multiple Blocks
4Ah	isoWriteAFI	AFI	Boolean	Write AFI
4Bh	isoLockAFI		Boolean	Lock AFI
4Ch	isoWriteDSFID	DSFID	Boolean	Write DSFID
4Dh	isoLockDSFID		Boolean	Lock DSFID
4Eh	isoIsBlockLock	Block Number	Boolean Lock Status	Get Block Lock Status (Single Block)
4Fh	iso15693	Flags Command Parameters	Response Size Response Data	Direct I/O for ISO15693
50h	isoStockRefresh			Refresh the tag stock
51h	Power	Number		RF Power On/Off/AutoOff
52h	GetStockUID	Index	UID	Get UID from RWM600A stock memory, index=0 to 47
	GetStockCount		Count	

GNetPlus Query#	ActiveX Control (RWM600x.OCX)	Parameters	Response	Description
53h	isoetMaskBit	Bit-Length	Bit-Length	Set anti-collision mask_bit length, Value range: 1~7
54h	isoAFI	AFI	AFI	Set AFI of reader Value range: 0~255, 0=Disable
56h	isoRequestMode	mode	mode	Set request mode of RF protocol. 0=Select Mode, 1=Address Mode
5Bh	BuzzerEnable	Boolean		Set Buzzer Enabled/Disabled
A2h	icodeSetEAS		Boolean	Set EAS
A3h	icodeResetEAS		Boolean	Reset EAS
A4h	icodeLockEAS		Boolean	Lock EAS
A5h	icodeEAS		EAS Sequence	EAS Alarm

Note:

January 12, 2006, New command 52h, 53h, 54h, 56h for RWM600A.

June 5, 2006, New command 50h and 3Fh for RWM600A

March 14, 2007, New I.CODE EAS command A2h~A5h for RWM600A

Commands Description

Polling

ActiveX Control Syntax

```
short = Object.gnetPolling(Reader_Address)
```

Return Tag Event (short type, and define in bit)

```
-1 = fai
```

```
bit0 = NOTHING
```

```
bit1 = TAG ARRIVED
```

```
bit2 = TAG REMOVED
```

```
bit3 = CLEAR
```

GNetPlus Protocol

Header	ID	Query#	Len	Parameters	Check	Tail
		00h	00			

Response

Header	ID	Response	Len	Parameters	Check	Tail
		ACK	02	Event + Protocol Buffer Size		

Note:

If Reader address is zero, the polling is broadcast

If reader address is FFh, the polling is broadcast without response "ACK" and the "AlohaEvent" instead.

Inventory (Anti-Collision)

ActiveX Control Syntax

Boolean = Object.**isoInventory**(Short MaskLength, Short MaskValue, Short *pDSFID, BYTE *UID)

Boolean = Object.**IsoInventoryEx**(Short AFI, Short MaskLength, Short MaskValue, Short *pDSFID, BYTE *UID)

GNetPlus Protocol

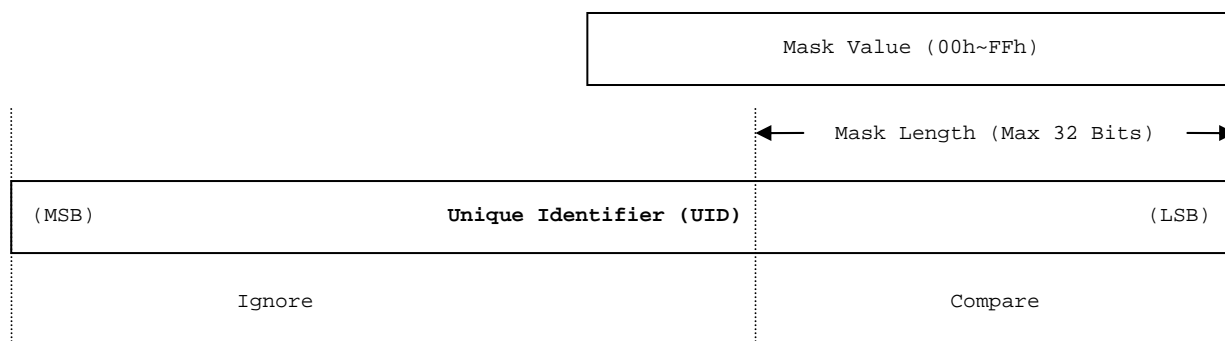
Header	ID	Query#	Len	Parameters	Check	Tail
		40h	02	MaskLength + MaskValue		

Header	ID	Query#	Len	Parameters	Check	Tail
		40h	03	AFI + MaskLength + MaskValue		

Response

Header	ID	Response	Len	Parameters	Check	Tail
		ACK	09	DSFID + (LSB) UID (MSB)		

Description



Use Inventory, The Smart Labels shall respond UID when Mask Value is matching.

Use InventoryEx, The Smart Labels shall respond UID when AFI and Mask Value are matching.

VB Example:

```
Private Sub cmdInventory_Click()
    Static MaskValue As Integer
    Dim nDSFID As Integer
    Dim bUID(7) As Byte
    Dim MaskLength As Integer

    cmdInventory.Enabled = False
    MaskLength = 7

    If RWM600x1.isoInventory(MaskLength, MaskValue, nDSFID, VarPtr(bUID(0))) Then
        If InList(lstReady, UID_CODE(bUID)) = False Then
            lstReady.AddItem UID_CODE(bUID)
        End If
    End If

    labMaskValue.Caption = CHex(MaskValue)
    MaskValue = (MaskValue + 1) And &H7F
    cmdInventory.Enabled = True
End Sub
```

Remark:

1. Use the VB "VarPtr" function to get the variable memory address.
2. The Inventory Command shall stop "Auto Mode" (Disable Anti-Collision).

Reset To Ready

ActiveX Control Syntax

Boolean = Object.**isoResetToReady**(BYTE *UID)

GNetPlus Protocol

Header	ID	Query#	Len	Parameters	Check	Tail
		41h	08	(LSB) UID (MSB)		

Response:

Header	ID	Response	Len	Parameters	Check	Tail
		ACK	00			

VB Example:

```
If RWM600x1.isoResetToReady(VarPtr(bUID(0))) Then
    If InList(lstReady, UID_CODE(bUID)) = False Then
        lstReady.AddItem UID_CODE(bUID)
    End If
    .RemoveItem .ListIndex
End If
```

Stay Quiet

ActiveX Control Syntax:

Boolean = Object.**isoStayQuiet**(BYTE *UID)

GNetPlus Protocol

Header	ID	Query#	Len	Parameters	Check	Tail
		42h	08	(LSB) UID (MSB)		

Response:

Header	ID	Response	Len	Parameters	Check	Tail
		ACK	00			

VB Example:

```
If RWM600x1.isoStayQuiet(VarPtr(bUID(0))) Then
    ' Check Duplicate
    If InList(lstQuiet, UID_CODE(bUID)) = False Then
        lstQuiet.AddItem UID_CODE(bUID)
    End If
End If
```


Select

ActiveX Control Syntax

Boolean = Object.**isoSelect**(BYTE *UID)

GNetPlus Protocol

Header	ID	Query#	Len	Parameters	Check	Tail
		43h	08	(LSB) UID (MSB)		

Response:

Header	ID	Response	Len	Parameters	Check	Tail
		ACK	00			

VB Example:

With lstReady

```
UID_BIN .List(.ListIndex), bUID
```

```
If RWM600x1.isoSelect(VarPtr(bUID(0))) Then
```

```
lstSelected.AddItem UID_CODE(bUID)
```

```
End If
```

```
.RemoveItem .ListIndex
```

End With

Get System Information

ActiveX Control Syntax

Boolean = Object.**isoGetSystemInformation**(INFO_T *Info)

GNetPlus Protocol

Header	ID	Query#	Len	Parameters	Check	Tail
		44h	00			

Response:

Header	ID	Response	Len	Parameters	Check	Tail
		ACK	0C	Info Structure		

Structure INFO

```
BYTE UID[8];  
BYTE DSFID;  
BYTE AFI;  
BYTE NumOfBlock;  
BYTE BlockSize;
```

VB Example:

```
Private Sub cmdSystemInfo_Click()  
    Dim I As Integer  
    Dim selInfo As RWM600XLib.INFO_T  
  
    If RWM600x1.isoGetSystemInformation(VarPtr(selInfo)) Then  
        ' Update Information Label  
        labInfo(0).Caption = UID_CODE(selInfo.UID)  
        txtDSFID.Text = CHex(selInfo.DSFID)  
        txtAFI.Text = CHex(selInfo.AFI)  
        labInfo(1).Caption = selInfo.NumOfBlock  
        labInfo(2).Caption = selInfo.BlockSize  
        txtDSFID.Enabled = True  
        txtAFI.Enabled = True  
  
    End If  
End Sub
```

Read Single Block

ActiveX Control Syntax

Boolean = Object.**isoReadSingleBlock**(Short nBlock, BOOL *bLock, BYTE *Block)

GNetPlus Protocol

Header	ID	Query#	Len	Parameters	Check	Tail
		45h	01	Block Number		

Response:

Header	ID	Response	Len	Parameters	Check	Tail
		ACK	05	Lock Status + Block Data		

VB Example

```
Private Sub cmdRead_Click()  
    Dim blkbuf(3) As Byte  
    Dim bLock as Boolean  
    heBlock.Text = ""  
    If RWM600x1.isoReadSingleBlock(cmbBlock.ListIndex, bLock, VarPtr(blkbuf(0))) Then  
        heBlock.Code = blkbuf  
    End If  
End Sub
```

Write Single Block

ActiveX Control Syntax

Boolean = Object.**isoWriteSingleBlock**(Short nBlock, BYTE *Block)

GNetPlus Protocol

Header	ID	Query#	Len	Parameters	Check	Tail
		46h	05	Block Number + Block Data		

Response:

Header	ID	Response	Len	Parameters	Check	Tail
		ACK	00			

VB Example

```
Private Sub cmdWrite_Click()  
    Dim blkbuf() As Byte  
    blkbuf = heBlock.Code  
    If RWM600x1.isoWriteSingleBlock(cmbBlock.ListIndex, VarPtr(blkbuf(0))) Then  
  
        End If  
End Sub
```

Lock Block

ActiveX Control Syntax

Boolean = Object.**isoLockBlock**(Short nBlock)

GNetPlus Protocol

Header	ID	Query#	Len	Parameters	Check	Tail
		47h	01	Block Number		

Response:

Header	ID	Response	Len	Parameters	Check	Tail
		ACK	00			

VB Example:

```
Private Sub cmdLock_Click()  
    If RWM600x1.isoLockBlock(cmbBlock.ListIndex) Then  
        cmdLock.Enabled = False  
        cmdWrite.Enabled = False  
        Label1(9).Caption = "Locked"  
    End If  
End Sub
```

Read Multiple Block

ActiveX Control Syntax

nCount = Object.**isoReadMultipleBlock**(Short nBlock, Short nCount, BYTE *Buf)

GNetPlus Protocol

Header	ID	Query#	Len	Parameters	Check	Tail
		48h	02	Start Block# + Count (1~8)		

Response:

Header	ID	Response	Len	Parameters	Check	Tail
		ACK	N	Multiple Blocks data		

N = Multiple Blocks Size (unit BYTE)

Block Count = N / 4

Remark: Max Count = 8;

VB Example

```
Private Sub cmdRead_Click()  
    Dim buf() As Byte  
    Dim nStartBlock As Integer  
    Dim nNumOfBlock  
    Dim nCount  
  
    nStartBlock = cmbBlock.ListIndex  
    nNumOfBlock = cmbBlockCount.ListIndex + 1  
  
    ReDim buf(63)  
    nCount = frmMain.RWM600x1.isoReadMultipleBlock(nStartBlock, nNumOfBlock, VarPtr(buf(0)))  
    If nCount > 0 Then  
        ReDim Preserve buf(nCount * 4 - 1)  
        heBlocks.Code = buf  
        heBlocks.MaxLength = nCount * 4  
    Else  
        heBlocks.Text = ""  
    End If  
  
    ReDim buf(0)  
End Sub
```

Write Multiple Block

ActiveX Control Syntax

nCount = Object.**isoWriteMultipleBlock**(Short nBlock, Short nCount, BYTE *Buf)

GNetPlus Protocol

Header	ID	Query#	Len	Parameters	Check	Tail
		49h	N	Start Block# + Count (1~8) + Buffer		

Response:

Header	ID	Response	Len	Parameters	Check	Tail
		ACK	00			

$N = 2 + (\text{Count} \times 4)$ (Unit Byte)

Remark:

The "Write Multiple Block" command is optional , May be it is not support by Smart Label.

Write AFI

ActiveX Control Syntax

Boolean = Object.**isoWriteAFI**(Short nAFI)

GNetPlus Protocol

Header	ID	Query#	Len	Parameters	Check	Tail
		4Ah	01	AFI		

Response:

Header	ID	Response	Len	Parameters	Check	Tail
		ACK	00			

VB Example

```
Private Sub cmdWriteAFI_Click()  
    RWM600x1.isoWriteAFI nAFI  
End Sub
```


Lock AFI

ActiveX Control Syntax

```
Boolean = Object.isoLockAFI()
```

GNetPlus Protocol

Header	ID	Query#	Len	Parameters	Check	Tail
		4Bh	00			

Response:

Header	ID	Response	Len	Parameters	Check	Tail
		ACK	00			

VB Example:

```
Private Sub cmdLockAFI_Click()  
    RWM600x1.isoLockAFI  
End Sub
```

Write DSFID

ActiveX Control Syntax

Boolean = Object.**isoWriteDSFID**(Short nDSFID)

GNetPlus Protocol

Header	ID	Query#	Len	Parameters	Check	Tail
		4Ch	01	DSFID		

Response:

Header	ID	Response	Len	Parameters	Check	Tail
		ACK	00			

VB Example:

```
Private Sub cmdWriteDSFID_Click()  
    RWM600x1.isoWriteDSFID nDSFID  
End Sub
```

Lock DSFID

ActiveX Control Syntax

```
Boolean = Object.isoLockDSFID()
```

GNetPlus Protocol

Header	ID	Query#	Len	Parameters	Check	Tail
		4Dh	00			

Response:

Header	ID	Response	Len	Parameters	Check	Tail
		ACK	00			

VB Example:

```
Private Sub cmdLockDSFID_Click()  
    RWM600x1.isoLockDSFID  
End Sub
```

Get Block Lock Status

ActiveX Control Syntax

Boolean = Object.**IsBlockLock**(Short nBlock, BOOL *bLock)

GNetPlus Protocol

Header	ID	Query#	Len	Parameters	Check	Tail
		4Eh	01	Block Number		

Response:

Header	ID	Response	Len	Parameters	Check	Tail
		ACK	01	Boolean		

Response Boolean : 01 = TRUE, 00 = FALSE.

VB Example:

```
Private Sub cmbBlock_Click()  
    Dim blkbuf(3) As Byte, bLock As Boolean  
  
    cmdRead.Enabled = False  
    cmdWrite.Enabled = False  
    cmdLock.Enabled = False  
  
    ' Updte Block Data  
    heBlock.Text = ""  
    If RWM600x1.isoReadSingleBlock(cmbBlock.ListIndex, VarPtr(blkbuf(0))) Then  
        heBlock.Code = blkbuf  
  
        ' Check Block Lock Status  
        If RWM600x1.isoIsBlockLock(cmbBlock.ListIndex, bLock) Then  
            If bLock = False Then  
                cmdWrite.Enabled = True  
                cmdLock.Enabled = True  
                Labell(9).Caption = "Free"  
            Else  
                Labell(9).Caption = "Locked"  
            End If  
        End If  
        cmdRead.Enabled = True  
        heBlock.Enabled = True  
    End If  
  
End Sub
```

Direct I/O for ISO15693

ActiveX Control Syntax

nRSize = Object.**iso15693**(Short Flags, Short Command, BYTE *Param, Short PSize, BYTE *Resp, Short RSize)

Param = Parameters buffer address

PSize = Parameters buffer size

Resp = Response buffer address

RSize = Response buffer size

Return nRSize = Response size.

GNetPlus Protocol

Header	ID	Query#	Len	Parameters	Check	Tail
		4Fh	N	Flags + Command + Parameters		

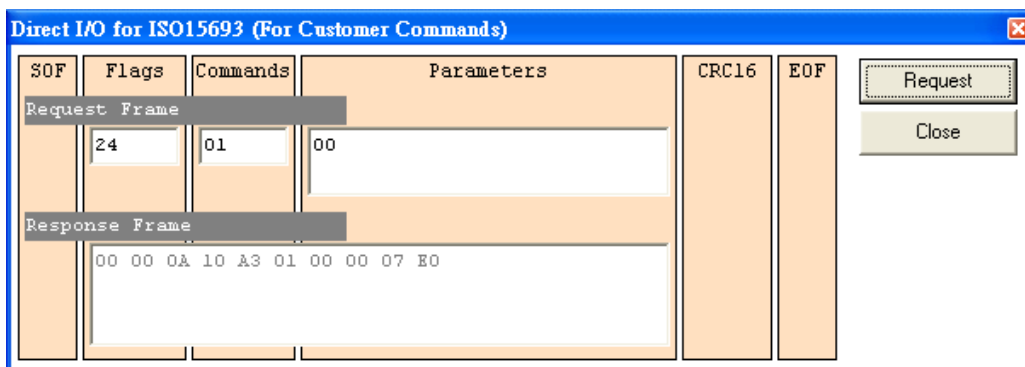
Response:

Header	ID	Response	Len	Parameters	Check	Tail
		ACK	L	Response		

N = 2 + Parameters Size.

L = Response Size.

VB Example:



```
Private Sub cmdRequest_Click()
    Dim Flags() As Byte
    Dim Commands() As Byte
    Dim Parameters() As Byte
    Dim Response() As Byte
    Dim nResponse As Integer

    Flags = heDirect(0).Code
    Commands = heDirect(1).Code
    Parameters = heDirect(2).Code

    ReDim Response(127)
    If heDirect(2).Length > 0 Then
        nResponse = frmMain.RWM600x1.iso15693(Flags(0), Commands(0), VarPtr(Parameters(0)),
            heDirect(2).Length, VarPtr(Response(0)), UBound(Response) + 1)
    Else
        nResponse = frmMain.RWM600x1.iso15693(Flags(0), Commands(0), vbNull, 0,
            VarPtr(Response(0)), UBound(Response) + 1)
    End If
End Sub
```

```
If nResponse > 0 Then
    ReDim Preserve Response(nResponse - 1)
    heResponse.Code = Response
Else
    heResponse.Text = ""
End If

ReDim Response(0)
End Sub
```

Remark:

The Direct Command shall stop "Auto Mode" (Disable Anti-Collision).

Enable Auto Mode

ActiveX Control Syntax

Object.**EnableAutoMode** = Boolean

GNetPlus Protocol

Header	ID	Query#	Len	Parameters	Check	Tail
		3Fh	1	Boolean		

Response:

Header	ID	Response	Len	Parameters	Check	Tail
		ACK	1	Boolean (Current Auto Mode Status)		

Query Boolean : 01 = TRUE, 00=FALSE.

VB Example:

```
Private Sub cmdAuto_Click()  
    lstReady.Clear  
    cmdQuiet(0).Enabled = False  
    RWM600x1.EnableAutoMode = True  
End Sub
```

Stock Refresh

ActiveX Control Syntax

Object.[isoStockRefresh](#)

GNetPlus Protocol

Header	ID	Query#	Len	Parameters	Check	Tail
		50h	0			

Response:

Header	ID	Response	Len	Parameters	Check	Tail
		ACK	0			

Remark:

The host will receive a CLEAN event and tag events from RWM600A after send the "isoStockRefresh".

RF Power Control

ActiveX Control Syntax

Boolean = Object.**Power**(long bON)

GNetPlus Protocol

Header	ID	Query#	Len	Parameters	Check	Tail
		51h	1	1 (ON) / 0 (OFF)		

Header	ID	Query#	Len	Parameters	Check	Tail
		51h	3	2 (AutoOff) + DelayTime(2 Bytes)		

Response:

Header	ID	Response	Len	Parameters	Check	Tail
		ACK	0			

Delay Time = millisecond

Set Anti-Collision Parameters (New RWM600x ActiveX, don't support the method.)

~~ActiveX Control Syntax~~
~~nMask = Object.isoSetAnticollisionParam(Short nMaskLength, Short AFI)~~

~~GNetPlus Protocol~~

Header	ID	Query#	Len	Parameters	Check	Tail
		53h	2	Mask-Length + AFI		

~~Response:~~

Header	ID	Response	Len	Parameters	Check	Tail
		ACK	1	Max Mask Value		

~~Mask-Length = 0-5. (Default=4)~~
~~AFI = 0-255, 0=disable. (Default=0)~~

Set Mask Bit Length

ActiveX Control Syntax

Object.isoMaskBit = Short

GNetPlus Protocol (when Len=0, Response Current Mask Bit Length only)

Header	ID	Query#	Len	Parameters	Check	Tail
		53h	1	Mask Bit Length		

Response:

Header	ID	Response	Len	Parameters	Check	Tail
		ACK	1	Mask Bit Length		

Note:

Value range = 1-7, 1=Fast, 7=Multi.

Set AFI

ActiveX Control Syntax

Object.isoAFI = Short

GNetPlus Protocol (when Len=0, Response Current AFI)

Header	ID	Query#	Len	Parameters	Check	Tail
		54h	1	AFI (0~255)		

Response:

Header	ID	Response	Len	Parameters	Check	Tail
		ACK	1	AFI		

Request Mode (for ISO15693 RF Protocol)

ActiveX Control Syntax

Object.**isoRequestMode** = Short

GNetPlus Protocol (when Len=0, Response Current Request Mode)

Header	ID	Query#	Len	Parameters	Check	Tail
		56h	1	Mode (0=Select, 1=Address)		

Response:

Header	ID	Response	Len	Parameters	Check	Tail
		ACK	1	Mode		

BuzzerEnable

ActiveX Control Syntax

Object.**BuzzerEnable** = Boolean

GNetPlus Protocol (when Len=0, Response Current Request Mode)

Header	ID	Query#	Len	Parameters	Check	Tail
		5Bh	1	Boolean		

Response:

Header	ID	Response	Len	Parameters	Check	Tail
		ACK	1	Boolean		

Set EAS (for I.CODE SL2)

ActiveX Control Syntax

Object.[icodeSetEAS](#)

GNetPlus Protocol (when Len=0, Response Current Request Mode)

Header	ID	Query#	Len	Parameters	Check	Tail
		A2h	0			

Response:

Header	ID	Response	Len	Parameters	Check	Tail
		ACK	1	0		

Reset EAS (for I.CODE SL2)

ActiveX Control Syntax

Object.[icodeResetEAS](#)

GNetPlus Protocol (when Len=0, Response Current Request Mode)

Header	ID	Query#	Len	Parameters	Check	Tail
		A3h	0			

Response:

Header	ID	Response	Len	Parameters	Check	Tail
		ACK	1	0		

Lock EAS (for I.CODE SL2)

ActiveX Control Syntax

Object.[icodeLockEAS](#)

GNetPlus Protocol (when Len=0, Response Current Request Mode)

Header	ID	Query#	Len	Parameters	Check	Tail
		A4h	0			

Response:

Header	ID	Response	Len	Parameters	Check	Tail
		ACK	1	0		

EAS Alarm (for I.CODE SL2)

ActiveX Control Syntax

Object.**icodeEAS**(**BYTE *Sequence**)

GNetPlus Protocol (when Len=0, Response Current Request Mode)

Header	ID	Query#	Len	Parameters	Check	Tail
		A5h	0			

Response:

Header	ID	Response	Len	Parameters	Check	Tail
		ACK	32	EAS Sequence 32 bytes		

If EAS bit is set to 0 the I-CODE remains silent (Timeout).

VB Example

```
Private Const GNET_ACK As Integer = &H6
Private Const GNET_NAK As Integer = &H15

'EAS Query Function Code
Private Const ICODE_EAS_SET_FUNC As Integer = &HA2
Private Const ICODE_EAS_RESET_FUNC As Integer = &HA3
Private Const ICODE_EAS_LOCK_FUNC As Integer = &HA4
Private Const ICODE_EAS_READ_FUNC As Integer = &HA5

'null parameter for Query function
Private QUERY_NULL_PARAM As Long
Private QUERY_NULL_SIZE As IntegerPrivate

Function icodeSetEAS() As Boolean
    Dim iResponse As Integer
    'send command "0xA2" to reader without parameter
    If frmMain.RWM600x1.gnetQuery(ICODE_EAS_SET_FUNC, QUERY_NULL_PARAM, QUERY_NULL_SIZE, iResponse,
QUERY_NULL_PARAM, QUERY_NULL_SIZE) Then
        If iResponse = GNET_ACK Then
            icodeSetEAS = True
        End If
    End If
End Function

Private Function icodeResetEAS() As Boolean
    Dim iResponse As Integer
    'send command "0xA3" to reader without parameter
    If frmMain.RWM600x1.gnetQuery(ICODE_EAS_RESET_FUNC, QUERY_NULL_PARAM, QUERY_NULL_SIZE, iResponse,
QUERY_NULL_PARAM, QUERY_NULL_SIZE) Then
        If iResponse = GNET_ACK Then
```

```

        icodeResetEAS = True
    End If
End If
End Function

Private Function icodeLockEAS() As Boolean
    Dim iResponse As Integer
    'send command "0xA4" to reader without parameter
    If frmMain.RWM600x1.gnetQuery(ICODE_EAS_LOCK_FUNC, QUERY_NULL_PARAM, QUERY_NULL_SIZE, iResponse,
QUERY_NULL_PARAM, QUERY_NULL_SIZE) Then
        If iResponse = GNET_ACK Then
            icodeLockEAS = True
        End If
    End If
End Function

Private Function icodeEAS() As Boolean
    Dim iResponse As Integer
    Dim buf(31) As Byte, iBufSize As Integer
    labEAS.Caption = ""
    'buffer size
    iBufSize = UBound(buf) - LBound(buf) + 1
    'send command "0xA5" to reader with 32bytes buffer for reply
    If frmMain.RWM600x1.gnetQuery(ICODE_EAS_READ_FUNC, QUERY_NULL_PARAM, QUERY_NULL_SIZE, iResponse,
VarPtr(buf(0)), iBufSize) Then
        If iResponse = GNET_ACK Then
            labEAS.Caption = bufToString(buf)
            icodeEAS = True
        Else
            labEAS.Caption = "No EAS Bit !"
            icodeEAS = False
        End If
    End If
End Function

```

Tag Events

ActiveX Control Syntax

```
void Object.OnTagEvent(Short nEvent, Short nDSFID)
```

```
nEvent:
```

```
TAG_CLEAN = 0 // No any Tag in Antenna Field
```

```
TAG_PRESENT = 1 // New Tag in Antenna Field
```

```
TAG_REMOVED = 2 // Tag out of Antenna Field
```

```
TAG_TIMEUP = 3 // RF Auto Off
```

GNetPlus Protocol (Auto Response from reader when reader in auto mode)

Response (Tag Present)

Header	ID	Response	Len	Parameters	Check	Tail
		12h	0A	'I' + DSFID + UID		

Response (Tag Remove)

Header	ID	Response	Len	Parameters	Check	Tail
		12h	9	'R' + UID		

Response (No any Tag in antenna field)

Header	ID	Response	Len	Parameters	Check	Tail
		12h	1	'C'		

Response (RF Auto Off)

Header	ID	Response	Len	Parameters	Check	Tail
		12h	1	'T'		

VB Example

```
Private Sub RWM600x1_OnTagEvent(ByVal nEvent As Integer, ByVal nDSFID As Integer)
    Dim bUID(7) As Byte ' Byte Array for UID, Size 8 Bytes (0~7)
    Dim i As Integer

    RWM600x1.GetCurrentUID VarPtr(bUID(0))

    Select Case nEvent
        Case TAG_CLEAN
            lstReady.Clear
            lstSelected.Clear
            ClearSelected

        Case TAG_PRESENT
            ' Check Duplicate
            If InList(lstReady, UID_CODE(bUID)) < 0 Then
                lstReady.AddItem UID_CODE(bUID)
            End If

        Case TAG_REMOVED
            ' If in ready list
            i = InList(lstReady, UID_CODE(bUID))
            If i >= 0 Then
                lstReady.RemoveItem i
            End If

            ' if in selected list
            i = InList(lstSelected, UID_CODE(bUID))
            If i >= 0 Then
                lstSelected.RemoveItem i
                ClearSelected
            End If
    End Select
End Sub
```

```
End If
```

```
End Select
```

```
End Sub
```

Remark:

Use `Object.GetCurrentUID` to get the current event UID.

Aloha Events

ActiveX Control Syntax

void Object.**OnAlohaEvent**(Short nAddr)

nAddr: Reader Address Number

GNetPlus Protocol (Auto Response from reader when reader in auto mode)

Response (Tag Present)

Header	ID	Response	Len	Parameters	Check	Tail
	nAddr	4Ah	00			

```
Private Sub RWM600x1_OnAlohaEvent(ByVal nAddr As Integer)
    Dim oItem As ListItem
    If nAddr > 0 Then
        m_bReaderEvent = True
        m_bNewReader = True
        addAddress (nAddr)
        Set oItem = lstAddress.findItem(CStr(nAddr))
        If Not oItem Is Nothing Then
            With oItem
                .SubItems(1) = "COM" & RWM600x1.CommPort
            End With
        End If
        cmdDelAddr.Enabled = True
    End If
End Sub
```

ANNEX A. Error Code

ActiveX Control Syntax

nErrorCode = Object.**GetLastError**

GNetPlus Protocol

Response:

Header	ID	Response	Len	Parameters	Check	Tail
		NAK	01	Error Code		

Error Codes Table:

	Error Code	Description
ISO15693	01	The command is not supported.
	02	The command is not recognized.
	03	The command option is not supported.
	0F	Error with no information given or a specific error code is not supported.
	10	The specified block is not available (doesn't exist).
	11	The specified block is already locked and thus cannot be locked again.
	12	The specified block is locked and its content cannot be changed.
	13	The specified block was not successfully programmed.
	14	The specified block was not successfully locked.
	A0~DF	Custom command error codes.
RWM600	00	No Error
	F0	Request Timeout
	F1	CRC Error (ISO15693 CRC16)
GNetPlus	E0	The command is deny
	E4	The command is ILLEGAL
	E6	Over run
	E7	CRC Error (GNetPlus CRC16)
	ED	Out of frame
	EE	The query number out of range
	EF	Unknown error

ANNEX B. AFI Coding

AFI most Significant Nibble	AFI least Significant Nibble	Meaning VICC respond from	Examples / Note
0	0	All families and sub-families	No application preselection
X	0	All sub-families of family X	Wide applicative preselection
X	Y	Only the Yth sub-family of family X	
0	Y	Proprietary sub-family Y only	
1	0, Y	Transport	Mass transit, Bus, Airline
2	0, Y	Financial	IEP, Banking, Retail
3	0, Y	Identification	Access control
4	0, Y	Telecommunication	Public telephony, GSM
5	0, Y	Medical	
6	0, Y	Multimedia	Internet services
7	0, Y	Gaming	
8	0, Y	Data storage	Portable files
9	0, Y	Item management	
A	0, Y	Express parcels	
B	0, Y	Postal Services	
C	0, Y	Airline bags	
D	0, Y	RFU	
E	0, Y	RFU	
F	0, Y	RFU	

NOTE : X=1 to F, Y=1 to F

The support of AFI by the Smart Label (VICC) is optional.

If AFI is not supported by the Smart Label and if use InventoryEx, the Smart Label shall not answer whatever the AFI value is in the request.

If AFI is support by the Smart Label, it shall answer according to the matching rules.

ANNEX C. GNetPlus Protocol (ASCII Format)

The RWM600 support two GNetPlus communication format. The ASCII format is easy to send command from HyperTerminal (or other Communication Terminal software) to reader, and easy to learn the ISO/IEC 15693.

Communication Package

	Header	ID ³	Query/Response	Len	Parameters/Data	Check	Tail
ASCII	Colon ¹	2 Bytes	2 Bytes	2 Bytes		None	CR ⁴
Binary	SOH ²	1 Byte	1 Byte	1 Byte		CRC16	None

Note:

1. Colon = 3Ah = ':'
2. SOH = 01h
3. ID = Reader Address (Communication Address, Default=0)
4. CR = 13h
5. About GNetPlus Binary format , Please see the GNetPlus Protocol instruction.

Examples:

```

:000000<CR>                HOST : Send Polling query command (00h)
:00060101<CR>              READER : Response ACK(06h) with one Tag in antenna field.
:0040020000<CR>           HOST : Send Inventory Query Command (40h)
:000609000D10A301000007E0<CR> READER : Response ACK(06h) and Tag UID=E007000001A3100D
:0043080D10A301000007E0<CR> HOST : Send Select Query Command (43h) with UID.
:00060100<CR>              READER : Response ACK(06h) with Error Code=0
:00450103<CR>              HOST : Read block 3
:00060400000000<CR>       READER : Response ACK(06h) with block 3 data = 00000000

```

ANNEX D. Supported Tag

The ISO/IEC 15693 standard was developed for “Contactless Vicinity Cards”. Adopted in 1998, ISO15693 has significantly enabled global acceptance of 13.56MHz RFID technology. Based on contributions by Texas Instruments and Philips, ISO/IEC 15693 is largely a superset of the features and specifications of the Tag-it HF and I-Code1 products, respectively.

ISO15693-1 defines the physical characteristics of a credit card transponder.

ISO15693-2 specifies the 13.56MHz air interface and modulation methods that accommodate regulatory bodies worldwide.

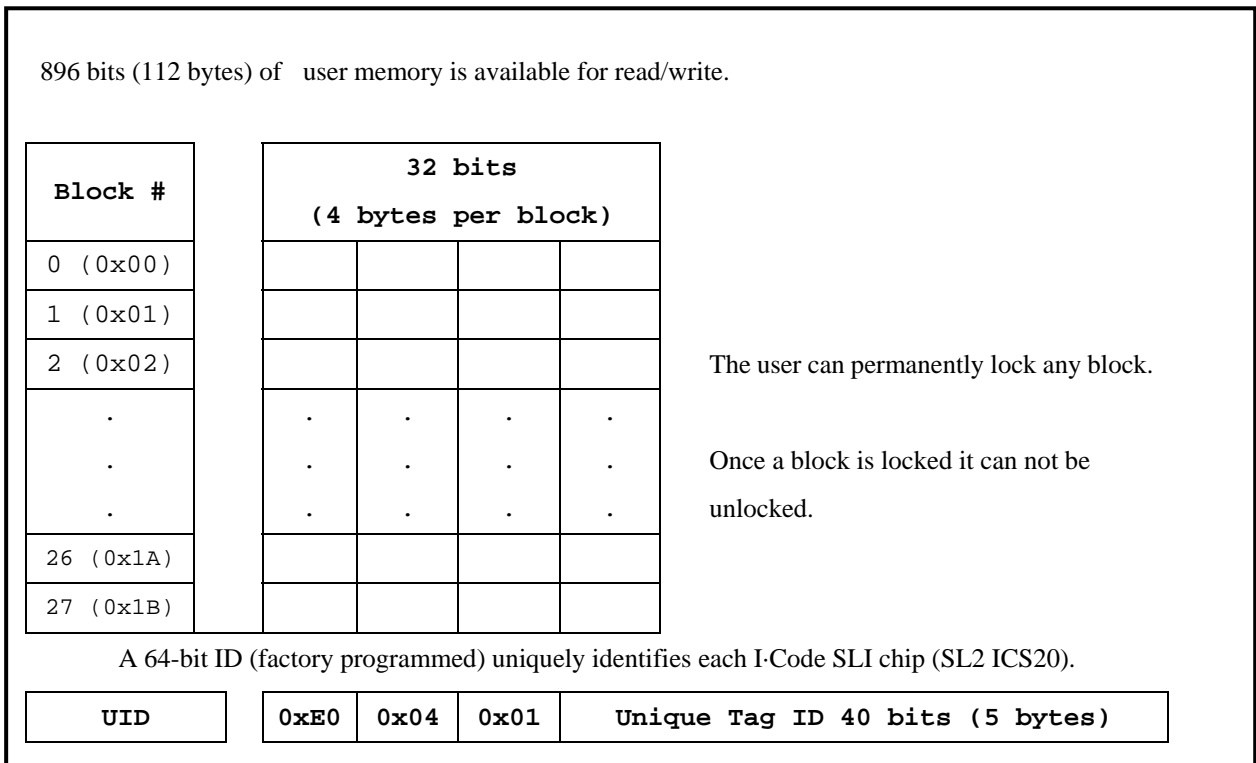
ISO15693-3 specifies the command protocol and anti-collision method for data exchange between tags and readers.

The ISO15693 “standard” permits tags to be manufactured that support optional and custom commands, and that have custom memory structures, sizes and architectures. The SkyeRead family of RFID readers fully supports all four (4) IC manufacturers that offer ISO/IEC 15693 compatible tags.

I-Code SLI

The complete I-Code SLI specification can be found in the Philips publication titled “I-Code SLI Smart Label IC SL2 ICS20 Functional Specification”.

Structure of the I-Code SLI (version SL2 ICS20)



Tag-It HF-I

The complete Tag-It HF-I specification can be found in the Texas Instruments publication titled “Tag-It HF-I Transponder Inlays Reference Guide”.

Memory Structure of the Tag-It HF-I

2K bits (256 bytes) of user memory is available for read/write.

Block #	32 bits (4 bytes per block)			
0 (0x00)				
1 (0x01)				
2 (0x02)				
.
.
.
62 (0x3E)				
63 (0x3F)				

The user can permanently lock any block.

Once a block is locked it can not be unlocked.

A 64-bit ID (factory programmed) uniquely identifies each Tag-It HF-I chip.

UID	0xE0	0x07	Unique Tag ID - 48 bits (6 bytes)
------------	-------------	-------------	--

LRI512

The complete LRI512 specification can be found in ST Microelectronics' publication titled “LRI512 Memory TAG IC 512 bit High Endurance EEPROM 13.56MHz, ISO 15693 Standard Compliant with E.A.S.”.

Memory Structure of the STM LRI512

512 bits (64 bytes) of user memory is available for read/write.

Block #	32 bits (4 bytes per block)			
0 (0x00)				
1 (0x01)				
2 (0x02)				
.
.
.
14 (0x0E)				
15 (0x0F)				

The user can permanently lock any block.

Once a block is locked it can not be unlocked.

A 64-bit ID (factory programmed) uniquely identifies each STM LRI512 chip.

UID	0xE0	0x02	Unique Tag ID 48 bits (6 bytes)
------------	-------------	-------------	--

My-d SRF55VxxP

The complete my-d SRF55VxxP specification can be obtained from Infineon.

My-d SRF55V02P

Memory Structure of the my-d SRF55V02P

29 blocks of 8 bytes = 232 bytes (1856 bits) of user memory is available for read/write.

Block #	64 bits (8 bytes per block)								
3 (0x03)									The user can permanently lock any block
4 (0x04)									
5 (0x05)									
.	Once a block is locked it can not be unlocked.
.	
.	
30 (0x1E)									
31 (0x1F)									

A 64-bit ID (factory programmed) uniquely identifies each my-d SRF55V02P chip.

UID	0x60	0x05	0x02	Unique Tag ID - 40 bits (5 bytes)
------------	-------------	-------------	-------------	--

My-d SRF55V10P

Memory Structure of the my-d SRF55V10P

125 blocks of 8 bytes = 1000 bytes (8000 bits) of user memory is available for read/write.

Block #	64 bits (8 bytes per block)								
3 (0x03)									The user can permanently lock any block
4 (0x04)									
.	Once a block is locked it can not be unlocked.
.	
.	
127 (0x7F)									

A 64-bit ID (factory programmed) uniquely identifies each my-d SRF55V10P chip.

UID	0x60	0x05	0x00	Unique Tag ID - 40 bits (5 bytes)
------------	-------------	-------------	-------------	--

ANNEX E. History

August 03, 2004 (REV.A)

Initial RWM600 Programming Guide.

January 12, 2006 (REV.B)

Add RWM600A new function. (Page 6)

June 7, 2006 (REV.C)

Add Stock Refresh command for RWM600A. (Page 8, Query=50h)

March 28, 2007 (REV.D)

Add EAS function

December 27, 2007 (REV.E)

Add PCR360 product

August 20, 2008 (REV.F)

Add "Tag Remove Event"

October 7, 2008 (REV.G)

Add "Aloha Events"

Add "Multi Reader Connection" (page 8)

PROMAG[®]

GIGA-TMS INC.

<http://www.gigatms.com.tw>

<mailto:promag@gigatms.com.tw>

TEL : +886-2-26954214

FAX : +886-2-26954213

Office: 8F, No. 31, Lane 169, Kang-Ning St., Hsi-Chih, Taipei, Taiwan