



Wisom Technology Corporation

11F-2, No. 289, Sec. 2, Guang-Fu Rd., Hsin-Chu 300, Taiwan, R.O.C.

Tel: 886-3-5163339

Fax: 886-3-5163679

WM Series
Software Development Kit
Programmer's Guide
For .Net framework development
Version Release 1.01



Wilson Technology Corporation

11F-2, No. 289, Sec. 2, Guang-Fu Rd., Hsin-Chu 300, Taiwan, R.O.C.

Tel: 886-3-5163339

Fax: 886-3-5163679

1. Class Config

Configuration for RS232

1.1 public int contrast

1.2 public BAUDRATE_PARA baudrate

0: 9600 1: 19200 2: 38400 3: 57600 4: 115200

1.3 public int autoAdjustImage

Adjust image automatically

1.4 public int sensorType

1.5 public int templateType

2: 480 Bytes 3: 160 Bytes 4: 320 Byte

1.6 public int reserve

1.7 public unsafe string version

Version of the firmware, Read Only

1.8 public unsafe string deviceId



Wilson Technology Corporation

11F-2, No. 289, Sec. 2, Guang-Fu Rd., Hsin-Chu 300, Taiwan, R.O.C.

Tel: 886-3-5163339

Fax: 886-3-5163679

2. ConfigEx

Extended configuration for TCP/IP module

2.1 public **int** *contrast*

2.2 public **BAUDRATE_PARA** *baudrate*
no used

2.3 public **int** *autoAdjustImage*
Adjust image automatically

2.4 public **int** *sensorType*

2.5 public **int** *templateType*
2:480 Bytes 3: 160 Bytes 4: 320 Byte

2.6 public **int** *reserve*

2.7 public unsafe **string** *version*
Version of the firmware, Read Only

2.8 public unsafe **string** *deviceId*

2.9 public unsafe **byte[]** *IP*

2.10 public unsafe **byte[]** *NetMask*

2.11 public unsafe **byte[]** *Gateway*



Wilson Technology Corporation

11F-2, No. 289, Sec. 2, Guang-Fu Rd., Hsin-Chu 300, Taiwan, R.O.C.

Tel: 886-3-5163339

Fax: 886-3-5163679

3. UserInfo

User information

3.1 public **int** *role*

Administrator : 0x00 User : 0x01

3.2 public **SECURITY_PARA** *securityLevel*

3.3 public **int** *fingerNo*

1st, 2nd, 3rd, Finger

3.4 public **int** *fingerid*

0 ~ 9

3.5 public **QUALITY_PARA** *quality*

Enrolled quality

3.6 public unsafe **string** *id*

3.7 public unsafe **string** *password*

3.8 public unsafe **byte[]** *enrlTemplate*

3.9 public unsafe **byte[]** *reserved*

3.10 public unsafe **byte[]** *id_array*

3.11 public **int** *score*



Wislon Technology Corporation

11F-2, No. 289, Sec. 2, Guang-Fu Rd., Hsin-Chu 300, Taiwan, R.O.C.

Tel: 886-3-5163339

Fax: 886-3-5163679

4. WisLib

4.1 constructor :

public *WisLib*(int portNo, BAUDRATE_PARA baudRate, int timeOut)
RS232 mode constructor

public *WisLib*(byte[] ip, int port)
Tcp/ip mode constructor

4.2 member :

public **int** *TemplateType*
[set only]

public **ArrayList** *UserList*
[get only]

public **Config** *Configuration*
[get and set]

public **ConfigEx** *ConfigurationEx*
[get and set]

public **int** *Contrast*
[set only]

public **int** *GPIO*
[get and set]

public **DateTime** *Time*
[get and set] device' date and time

public **string** *DeviceId*
[set only]



Wison Technology Corporation

11F-2, No. 289, Sec. 2, Guang-Fu Rd., Hsin-Chu 300, Taiwan, R.O.C.

Tel: 886-3-5163339

Fax: 886-3-5163679

public **POWER_MODE** *PowerMode*

[set only]



Wison Technology Corporation

11F-2, No. 289, Sec. 2, Guang-Fu Rd., Hsin-Chu 300, Taiwan, R.O.C.

Tel: 886-3-5163339

Fax: 886-3-5163679

4.3 method

public **bool** *TestDevice*()

This function diagnoses your fingerprint device. Before testing, please clean the capture area and make sure that there is no finger on the reader.

public **bool** *CheckNoFinger*()

This function is mainly used in the enrollment process. To get the stable and real features of a fingerprint during the enrollment, the user must remove his finger from the reader after *Enroll*() has successfully been processed for this snapped fingerprint image. You can check if a fingerprint has actually been lifted off the reader by using this function.

public **void** *Enroll*(ref UserInfo info)

To generate a final fingerprint code

Enroll() use no input and stores the results to *UserInfo*'s **quality** and **enrlTemplate** members

public **void** *ReleaseEnroll*()

This function releases all the internal resource created during the enrollment process. Call this function only if *Enroll*() is no longer in use.

public **bool** *Verify*(ref UserInfo info)

To do 1-1 fingerprint verification by giving a specify user ID, security level and fingerprint

Before calling this function, you must prompt user to place his(her) finger as *Verify* () will capture the image from the fingerprint device to generate the feature.

Verify () use *UserInfo*'s **id** and **securityLevel** members as input , and stores the results to *UserInfo*'s **score**, **fingerNo** and **fingered** members



Wilson Technology Corporation

11F-2, No. 289, Sec. 2, Guang-Fu Rd., Hsin-Chu 300, Taiwan, R.O.C.

Tel: 886-3-5163339

Fax: 886-3-5163679

public **bool** *Identify*(ref *UserInfo* info)

To do 1-N fingerprint identification with the enrolled templates saved in the system..

Before calling this function, you must prompt user to place his(her) finger as *Identify*() will capture the image from the fingerprint device to generate the feature

Identify () use *UserInfo*'s **securityLevel** member as input , and stores the results to *UserInfo*'s **id**, **score**, **fingerNo** and **fingerId** members

public **int** *VerifyTemplate*(byte[] template, int security)

To do 1-1 verification by giving a specified template, Security level and fingerprint.

public **bool** *ReadUserData*(ref *UserInfo* info)

Load a specified user's information that stored in the fingerprint module.

ReadUserData () use *UserInfo*'s **id** member as input , and stores the results to *UserInfo*'s **fingerNo** and **fingerId** members

public **bool** *WriteUserData*(ref *UserInfo* info, DB_ACCESS access)

Create or update a specified user's information in the fingerprint module.

WriteUserData () use *UserInfo*'s **id** and **DB_ACCESS** member as input , and stores the results to the class *UserInfo*

public **bool** *DeleteUserData*(string id)

Delete the user's information that stored in the fingerprint module.

public **bool** *DeleteAllUser*()

Delete all the users information that stored in the fingerprint module



Wilson Technology Corporation

11F-2, No. 289, Sec. 2, Guang-Fu Rd., Hsin-Chu 300, Taiwan, R.O.C.

Tel: 886-3-5163339

Fax: 886-3-5163679

public **byte[]** *Snap*(IMAGE_PARA flag)

This function provides an interface for user to get the currently captured image buffer from the fingerprint module.

Once the function is called, the fingerprint module captures image from the reader and stores to a frame buffer and send to the client.

Snap() use IMAGE_PARA as input and return a byte array

public **byte[]** *GetImage*(IMAGE_PARA flag)

This function provides an interface for user to get the previously captured image buffer from the fingerprint module.

GetImage () use IMAGE_PARA as input and return a byte array

public **Size** *GetImageSize*(IMAGE_PARA para)

This function provides an interface for user to get the size of the currently captured image

GetImageSize () use IMAGE_PARA as input and return a Size

public **bool** *LoadDefaultSetting*()

This function provides an interface for user to reload default system configuration to the fingerprint module. It will take effect immediately without shutting down the system.

public **bool** *UpdateProgram*(string filename)

This function provides an interface for user to update user's program

public **void** *DisplayImage*(IntPtr handle, IMAGE_PARA flag, Rectangle rc)

This function provides an interface for user to display an finger-print image

DisplayImage() use Control.Handle , IMAGE_PARA and Rectangle as input



Wison Technology Corporation

11F-2, No. 289, Sec. 2, Guang-Fu Rd., Hsin-Chu 300, Taiwan, R.O.C.

Tel: 886-3-5163339

Fax: 886-3-5163679

public **bool** *SaveImage*(IMAGE_PARA flag, string Filename)

This function provides an interface for user to load image from device and store it to a bitmap file