



APPLICATION PROGRAMMING INTERFACE DESCRIPTION (API)

Version 5.0

(TECHNICAL & RELATED INFORMATION)

Compiled by:
iPulse Biometrics (Pty) Ltd.

-- PROPRIETARY INFORMATION --

THE TRADEMARKS AND REGISTERED TRADEMARKS MENTIONED IN THIS PUBLICATION ARE THE PROPERTY OF THEIR RESPECTIVE HOLDERS. UNLESS OTHERWISE NOTED, THE ENTIRE CONTENTS OF THIS PUBLICATION MAY NOT BE REPRODUCED, STORED IN A RETRIEVAL SYSTEM, OR TRANSMITTED IN ANY FORM OR BY ANY MEANS WITHOUT PRIOR WRITTEN CONSENT OF THE PUBLISHER.

Table of Content

APPLICATION PROGRAMMING INTERFACE (IDUOIF.DLL)	4
PROPERTIES	4
DataReceived	4
DataSend	4
GetFingerMinutiaeCancel	4
GetFingerMinutiaeRead	4
IDU2IPAddress	5
IDU2Port	5
PCCommandPort	5
PCIPAddress	5
PCTimePort	5
SerialNumber	5
SerialPort	6
METHODS	7
Clock	7
DisplayName	7
FingerDelay	7
Fingernam	7
FirmwareDownload	8
FirmwareInfo	8
FlagCardReader	8
FlagEnrollModeSet	8
FlagIdentifyOnServer	9
FlagLcdOn	9
FlagOtherUnitOpen	9
FlagRequireDirection	9
FlagRequirePin	10
FlagSensorLEDOn	10
FlagTimePush	10
FlagUseCamera	10
GetError	11
GetProxCardCode	11
IDDeleteAll	11
IDDeviceSettingsRead	11
IDDeviceSettingsSet	11
IDFingerMinutiaeGet	12
IDGetImage	12
IDGetVersion	12
IDIdentify	12
IDOpticleLed	13
IDPersonAdd	13
IDPersonDelete	13
IDPersonGet	13
IDTestSensor	14
IDUFirmwareVersion	14
IDUType	14
ListenForTimeStamps	14
NetSettings	15
OpenRelay	15
PING	15
RelayDelay	15
RelayDuration	15
ResetIDU2	16
SerialNoGet	16
StatCountersGet	16
TimeRecordsNumGet	16
TimeRecordGet	17
TimeRecordsClear	17
VerifyReply	17
WiegandFacilityCode	17
WiegandProtocol	18
EVENTS	19
AccessBlocked	19
AccessDenied	19
FWProgress	19
IDUError	19
MenuSelection	19
OpenOtherRelay	19
RequestFinger	19
ServerIdentify	19
TimeRecordGet	19
VerifyRequest	19
ENUMERATIONS USED BY INTERFACE	20

COMM_LAYER	20
DEVICE_TEST	20
FINGER_NUMBER	20
IDU_COMMAND_TYPE	20
IMAGE_SIZE	20
RELAY_ACCESS	20
RETURN_CODE	20
SECURITY_LEVEL	22
SWITCH_VALUE	22
USER_TYPE	22
USER_VERIFICATION	22
TYPES/STRUCTURES USED BY INTERFACE	23
DEVICE_SETTINGS	23
NET_SETTINGS	23
PERSON_RECORD	23

APPLICATION PROGRAMMING INTERFACE (IDUOIF.DLL)

The IDUOIF.DLL is provided with the standard admin/access control tool – mentioned ActiveX dll provides programmers with an API to easily interface into existing applications – also allowing programmers & designers to write their own custom 3rd party products. Please refer to addendum A for API detail.

PROPERTIES

DataReceived

Returns the data received by the IDU device.

Syntax: object.DataReceived

The DataReceived property syntax has these parts:

Part	Description
Object	An object expression that evaluates to an instantiation of the IDU interface class

DataSend

Returns the data sent to the IDU device.

Syntax: object.DataSend

The DataSend property syntax has these parts:

Part	Description
Object	An object expression that evaluates to an instantiation of the IDU interface class

GetFingerMinutiaeCancel

In the enrollment process the IDU device will fire a RequestFinger event, the GetFingerMinutiaeCancel flag can then be set to TRUE to indicate that the enrollment process must be aborted.

Syntax: object.GetFingerMinutiaeCancel [= value]

The GetFingerMinutiaeCancel property syntax has these parts:

Part	Description
Object	An object expression that evaluates to an instantiation of the IDU interface class
Value	A Boolean expression indicating whether the enrollment process must be aborted

Settings

The settings for value are:

Setting	Description
FALSE	Continue enrollment process
TRUE	Abort enrolment process

GetFingerMinutiaeRead

Sets a flag to indicate that a person has put a finger on the sensor.

Syntax: object.GetFingerMinutiaeRead = value

The GetFingerMinutiaeRead property syntax has these parts:

Part	Description
Object	An object expression that evaluates to an instantiation of the IDU interface class
Value	A Boolean expression indication that a finger is on the sensor and is ready to be read.

Settings

The settings for value are:

Setting	Description
---------	-------------

Setting	Description
TRUE	A finger is on the sensor and is ready to be read
FALSE	There is no finger on the sensor

IDU2IPAddress

Returns or set the IP Address of the IDU device to communicate with.

Syntax: object.IDU2IPAddress [= string]

The IDU2IPAddress property syntax has these parts:

Part	Description
Object	An object expression that evaluates to an instantiation of the IDU interface class
String	A string expression specifying the IDU device to communicate with

IDU2Port

Returns or set UDP Port of the IDU device to communicate with. Both an IP address and a UDP port must be specified to successfully communicate with and IDU device

Syntax: object.IDU2Port [= string]

The IDU2Port property syntax has these parts:

Part	Description
Object	An object expression that evaluates to an instantiation of the IDU interface class
String	A string expression specifying the IDU port the IDU device will listen on

PCCommandPort

Returns or set the UDP Port of the PC used to listen for command responses from the IDU device.

Syntax: object.PCCommandPort [= string]

The PCCommandPort property syntax has these parts:

Part	Description
object	An object expression that evaluates to an instantiation of the IDU interface class
String	A string expression specifying the UDP port will listen for responses from the IDU device

PCIPAddress

Returns the IP Address of the PC

Syntax: object.PCIPAddress

The PCIPAddress property syntax has these parts:

Part	Description
Object	An object expression that evaluates to an instantiation of the IDU interface class

PCTimePort

Returns or set the UDP Port of the PC used to get time records

Syntax: object.PCTimePort [= string]

The PCTimePort property syntax has these parts:

Part	Description
Object	An object expression that evaluates to an instantiation of the IDU interface class
String	An string expression specifying on which UDP port the PC must listen for time records send by the IDU devices

SerialNumber

Returns the serial number of the IDU device

Syntax: object. SerialNumber

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

Part	Description
Object	An object expression that evaluates to an instantiation of the IDU interface class
Value	A string value with the serial number of the IDU device

SerialPort

Return or set the communications port of the PC connected to the IDU device

Syntax: **object.SerialPort** [= Value]

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

Part	Description
Object	An object expression that evaluates to an instantiation of the IDU interface class
Value	A integer expression specifying the communications port (1 – 4) on which the IDU device is connected

METHODS

Clock

Real time clock (RTC) functions

Syntax: **object.Clock** DateTime, CommandType, COMM_LAYER

The **Clock** method syntax has these parts:

Part	Description
Object	An object expression that evaluates to an instantiation of the IDU interface class
DateTime	Date and time variable
CommandType	Type of command: 0 – IDU_GET, 1 – IDU_SET
COMM_LAYER	Communications type: 0 – IDU_SERIAL, 1- IDU_NETWORK

Return Value

This method will return a long value where 0 indicates success and a non-zero value indicates failure.

DisplayName

Name that is displayed on the IDU devices LCS

Syntax: **object.DisplayName** sName, CommandType, COMM_LAYER

The **DisplayName** method syntax has these parts:

Part	Description
Object	An object expression that evaluates to an instantiation of the IDU interface class
sName	Display string of LCD
CommandType	Type of command: 0 – IDU_GET, 1 – IDU_SET
COMM_LAYER	Communications type: 0 – IDU_SERIAL, 1- IDU_NETWORK

Return Value

This method will return a long value where 0 indicates success and a non-zero value indicates failure.

FingerDelay

Delay before finger is scanned

Syntax: **object.FingerDelay** TimeUnits, CommandType, COMM_LAYER

The **FingerDelay** method syntax has these parts:

Part	Description
Object	An object expression that evaluates to an instantiation of the IDU interface class
TimeUnits	Number of 500 milli-second periods to delay before finger is scanned
CommandType	Type of command: 0 – IDU_GET, 1 – IDU_SET
COMM_LAYER	Communications type: 0 – IDU_SERIAL, 1- IDU_NETWORK

Return Value

This method will return a long value where 0 indicates success and a non-zero value indicates failure.

Fingername

Return a name of a finger

Syntax: **object.Fingername** FingerNumber

The **Fingername** method syntax has these parts:

Part	Description
Object	An object expression that evaluates to an instantiation of the IDU interface class
FingerNumber	Number of finger (0-9) to get name for

Return Value

This method will return a string value with the fingername.

FirmwareDownload

Loads new firmware onto device(s).

Syntax: **object.FirmwareDownload FWFile, COMM_LAYER**

The **FirmwareDownload** method syntax has these parts:

Part	Description
Object	An object expression that evaluates to an instantiation of the IDU interface class
FWFile	Full path to firmware file.
COMM_LAYER	Communications type: 0 – IDU_SERIAL, 1- IDU_NETWORK

Return Value

This method will return a long value where 0 indicates success and a non-zero value indicates failure.

FirmwareInfo

Provides info of firmware of device.

Syntax: **object.FirmwareInfo FWFile, FWName, FWVersion, FWDate, FWComment, FWType**

The **FirmwareInfo** method syntax has these parts:

Part	Description
Object	An object expression that evaluates to an instantiation of the IDU interface class
FWFile	Firmware filename
FWName	Firmware name
FWVersion	Firmware version
FWDate	Firmware date
FWComment	Firmware comments
COMM_LAYER	Communications type: 0 – IDU_SERIAL, 1- IDU_NETWORK
FWType	Returns a string indicating the IDU version this firmware is intended for

Return Value

This method will return a long value where 0 indicates success and a non-zero value indicates failure.

FlagCardReader

Flag on the IDU device to indicate whether a proximity (Wiegand) card reader is connected.

Syntax: **object.FlagCardReader ReaderConnected, COMM_LAYER**

The **FlagCardReader** method syntax has these parts:

Part	Description
Object	An object expression that evaluates to an instantiation of the IDU interface class
ReaderConnected	Boolean value indicating whether proximity card reader is connected
COMM_LAYER	Communications type: 0 – IDU_SERIAL, 1- IDU_NETWORK

Return Value

This method will return a long value where 0 indicates success and a non-zero value indicates failure.

FlagEnrollModeSet

Flag on the IDU device to put the IDU in enrollment mode

Syntax: **object.FlagEnrollModeSet DisableComms, COMM_LAYER**

The **FlagEnrollModeSet** method syntax has these parts:

Part	Description
Object	An object expression that evaluates to an instantiation of the IDU interface class
DisableComms	Boolean value indicating whether to disable all communications except enrolment
COMM_LAYER	Communications type: 0 – IDU_SERIAL, 1- IDU_NETWORK

Return Value

This method will return a long value where 0 indicates success and a non-zero value indicates failure.

FlagIdentifyOnServer

Flag on the IDU device whether a string with the scanned person must be send to a DVR

Syntax: **object. FlagIdentifyOnServer ServerIdentify, CommandType, COMM_LAYER**

The **FlagIdentifyOnServer** method syntax has these parts:

Part	Description
Object	An object expression that evaluates to an instantiation of the IDU interface class
ServerIdentify	Boolean flag indicating whether device will send minutiae to server for verification
CommandType	Type of command: 0 – IDU_GET, 1 – IDU_SET
COMM_LAYER	Communications type: 0 – IDU_SERIAL, 1- IDU_NETWORK

Return Value

This method will return a long value where 0 indicates success and a non-zero value indicates failure.

FlagLcdOn

Flag on the IDU device whether LCD backlight must always be on

Syntax: **object. FlagLcdOn LcdOn, CommandType, COMM_LAYER**

The **FlagLcdOn** method syntax has these parts:

Part	Description
Object	An object expression that evaluates to an instantiation of the IDU interface class
LcdOn	Boolean flag indicating whether LCD backlight will always stay on
CommandType	Type of command: 0 – IDU_GET, 1 – IDU_SET
COMM_LAYER	Communications type: 0 – IDU_SERIAL, 1- IDU_NETWORK

Return Value

This method will return a long value where 0 indicates success and a non-zero value indicates failure.

FlagOtherUnitOpen

Flag on the IDU device whether another IDU will open door (release relay)

Syntax: **object. FlagOtherUnitOpen OtherUnitOpen, CommandType, COMM_LAYER**

The **FlagOtherUnitOpen** method syntax has these parts:

Part	Description
Object	An object expression that evaluates to an instantiation of the IDU interface class
OtherUnitOpen	Boolean flag indicating whether device will ask server to open relay
CommandType	Type of command: 0 – IDU_GET, 1 – IDU_SET
COMM_LAYER	Communications type: 0 – IDU_SERIAL, 1- IDU_NETWORK

Return Value

This method will return a long value where 0 indicates success and a non-zero value indicates failure.

FlagRequireDirection

Flag on the IDU device whether a string with the scanned person must be send to a DVR

Syntax: **object. FlagRequireDirection RequireDirection, CommandType, COMM_LAYER**

The **FlagRequireDirection** method syntax has these parts:

Part	Description
Object	An object expression that evaluates to an instantiation of the IDU interface class
RequireDirection	Boolean flag indicating whether device enforce direction button
CommandType	Type of command: 0 – IDU_GET, 1 – IDU_SET
COMM_LAYER	Communications type: 0 – IDU_SERIAL, 1- IDU_NETWORK

Return Value

This method will return a long value where 0 indicates success and a non-zero value indicates failure.

FlagRequirePin

Flag on the IDU device to force all persons to enter a PIN.

Syntax: **object.FlagRequirePin RequirePin, COMM_LAYER**

The **FlagRequirePin** method syntax has these parts:

Part	Description
Object	An object expression that evaluates to an instantiation of the IDU interface class
RequirePin	Boolean value indicating whether to disable all PIN requirement
COMM_LAYER	Communications type: 0 – IDU_SERIAL, 1- IDU_NETWORK

Return Value

This method will return a long value where 0 indicates success and a non-zero value indicates failure.

FlagSensorLEDon

Flag on the IDU device whether sensor LED must always be on

Syntax: **object. FlagSensorLEDon LEDOn, CommandType, COMM_LAYER**

The **FlagSensorLEDon** method syntax has these parts:

Part	Description
Object	An object expression that evaluates to an instantiation of the IDU interface class
LEDon	Boolean flag indicating whether sensor LED will always stay on
CommandType	Type of command: 0 – IDU_GET, 1 – IDU_SET
COMM_LAYER	Communications type: 0 – IDU_SERIAL, 1- IDU_NETWORK

Return Value

This method will return a long value where 0 indicates success and a non-zero value indicates failure.

FlagTimePush

Flag on the IDU device to indicate whether successful finger scans should be send to the server

Syntax: **object.FlagTimePush DateTimePush , CommandType, COMM_LAYER**

The **FlagTimePush** method syntax has these parts:

Part	Description
Object	An object expression that evaluates to an instantiation of the IDU interface class
DateTimePush	Boolean value indicating whether device must send time record on successful scan
CommandType	Type of command: 0 – IDU_GET, 1 – IDU_SET
COMM_LAYER	Communications type: 0 – IDU_SERIAL, 1- IDU_NETWORK

Return Value

This method will return a long value where 0 indicates success and a non-zero value indicates failure.

FlagUseCamera

Flag on the IDU device whether a string with the scanned person must be send to a DVR

Syntax: **object.FlagUseCamera UseCamera, CommandType, COMM_LAYER**

The **FlagUseCamera** method syntax has these parts:

Part	Description
Object	An object expression that evaluates to an instantiation of the IDU interface class
UseCamera	Boolean flag indicating whether device must send string to DVR
CommandType	Type of command: 0 – IDU_GET, 1 – IDU_SET
COMM_LAYER	Communications type: 0 – IDU_SERIAL, 1- IDU_NETWORK

Return Value

This method will return a long value where 0 indicates success and a non-zero value indicates failure.

GetError

Returns error description

Syntax: **object.GetError** ErrorNumber

The **GetError** method syntax has these parts:

Part	Description
Object	An object expression that evaluates to an instantiation of the IDU interface class
ErrorNumber	Error number for which error description must be retrieved

Return Value

This method will return a string value with the error description.

GetProxCardCode

Returns error description

Syntax: **object.GetProxCardCode** ProxCardCode, WiegandBits, COMM_LAYER

The **GetProxCardCode** method syntax has these parts:

Part	Description
Object	An object expression that evaluates to an instantiation of the IDU interface class
ProxCardCode	String value in which the proximity card number will be returned
WiegandBits	Number of bits of Wiegand card: 26 or 44
COMM_LAYER	Communications type: 0 – IDU_SERIAL, 1- IDU_NETWORK

Return Value

This method will return a string value with the error description.

IDDeleteAll

Clear all persons from the archive.

Syntax: **object.IDDeleteAll**, COMM_LAYER

The **ClearArchive** method syntax has these parts:

Part	Description
Object	An object expression that evaluates to an instantiation of the IDU interface class
COMM_LAYER	Communications type: 0 – IDU_SERIAL, 1- IDU_NETWORK

Return Value

This method will return a long value where 0 indicates success and a non-zero value indicates failure.

IDDeviceSettingsRead

Retrieves sensor device settings

Syntax: **object.IDDeviceSettingsRead** DeviceSettings, COMM_LAYER

The **IDDeviceSettingsRead** method syntax has these parts:

Part	Description
Object	An object expression that evaluates to an instantiation of the IDU interface class
DeviceSettings	DEVICE_SETTINGS structure, settings will be retrieved in this structure
COMM_LAYER	Communications type: 0 – IDU_SERIAL, 1- IDU_NETWORK

Return Value

This method will return a long value where 0 indicates success and a non-zero value indicates failure.

IDDeviceSettingsSet

Sets sensor device settings

Syntax: **object.IDDeviceSettingsSet** DeviceSettings, COMM_LAYER

The **IDDeviceSettingsSet** method syntax has these parts:

Part	Description
------	-------------

Part	Description
Object	An object expression that evaluates to an instantiation of the IDU interface class
DeviceSettings	DEVICE_SETTINGS structure, settings to set is saved n this structure
COMM_LAYER	Communications type: 0 – IDU_SERIAL, 1- IDU_NETWORK

Return Value

This method will return a long value where 0 indicates success and a non-zero value indicates failure.

IDFingerMinutiaeGet

Starts the enrollment process

Syntax: **object.IDFingerMinutiaeGet** , FingerData, COMM_LAYER

The **IDFingerMinutiaeGet** method syntax has these parts:

Part	Description
Object	An object expression that evaluates to an instantiation of the IDU interface class
FingerData	Fingerminutiae
COMM_LAYER	Communications type: 0 – IDU_SERIAL, 1- IDU_NETWORK

Return Value

This method will return a long value where 0 indicates success and a non-zero value indicates failure.

IDGetImage

Scans a finger and returns the image

Syntax: **object.IDGetImage** ImageString, ImageSize, COMM_LAYER

The **IDGetImage** method syntax has these parts:

Part	Description
Object	An object expression that evaluates to an instantiation of the IDU interface class
ImageString	String in which image will be saved
ImageSize	Size of image to return: 1 – full size, 2 – half size, 4 – quarter size
COMM_LAYER	Communications type: 0 – IDU_SERIAL, 1- IDU_NETWORK

Return Value

This method will return a long value where 0 indicates success and a non-zero value indicates failure.

IDGetVersion

Returns sensor firmware version

Syntax: **object.IDGetVersion** Version, TextVersion, COMM_LAYER

The **IDGetVersion** method syntax has these parts:

Part	Description
Object	An object expression that evaluates to an instantiation of the IDU interface class
Version	Single value with the firmware version
TextVersion	String with firmware version
COMM_LAYER	Communications type: 0 – IDU_SERIAL, 1- IDU_NETWORK

Return Value

This method will return a long value where 0 indicates success and a non-zero value indicates failure.

IDIdentify

Test identification function on the IDU device

Syntax: **object.IDIdentify** PersonID, COMM_LAYER

The **IDIdentify** method syntax has these parts:

Part	Description
Object	An object expression that evaluates to an instantiation of the IDU interface class
PersonID	ID of person identified

Part	Description
COMM_LAYER	Communications type: 0 – IDU_SERIAL, 1- IDU_NETWORK

Return Value

This method will return a long value where 0 indicates success and a non-zero value indicates failure.

IDOpticleLed

Turns sensor led on or off

Syntax: **object.IDOpticleLed LEDOn, COMM_LAYER**

The **IDOpticleLed** method syntax has these parts:

Part	Description
Object	An object expression that evaluates to an instantiation of the IDU interface class
LEDOn	Integer value to set sensor led: 0 – Off, 1 - On
Surname	Surname of person retrieved
COMM_LAYER	Communications type: 0 – IDU_SERIAL, 1- IDU_NETWORK

Return Value

This method will return a long value where 0 indicates success and a non-zero value indicates failure.

IDPersonAdd

Add a person to the archive

Syntax: **object.IDPersonAdd PersonID, PersonRecord, Comms**

The **IDPersonAdd** method syntax has these parts:

Part	Description
Object	An object expression that evaluates to an instantiation of the IDU interface class
PersonID	ID of person to add
PersonRecord	Type that contains all person data including biometrics data
COMM_LAYER	Communications type: 0 – IDU_SERIAL, 1- IDU_NETWORK

Return Value

This method will return a long value where 0 indicates success and a non-zero value indicates failure.

IDPersonDelete

Delete a person from the archive

Syntax: **object.IDPersonDelete PersonID, COMM_LAYER**

The **IDPersonDelete** method syntax has these parts:

Part	Description
Object	An object expression that evaluates to an instantiation of the IDU interface class
PersonID	ID of person to delete
COMM_LAYER	Communications type: 0 – IDU_SERIAL, 1- IDU_NETWORK

Return Value

This method will return a long value where 0 indicates success and a non-zero value indicates failure.

IDPersonGet

Retrieves a persons data from the archive

Syntax: **object.IDPersonGet PersonID, PersonRecord, COMM_LAYER**

The **IDPersonGet** method syntax has these parts:

Part	Description
Object	An object expression that evaluates to an instantiation of the IDU interface class
PersonID	ID of person to get
PeronRecord	Type containing all person data
Surname	Surname of person retrieved

Part	Description
COMM_LAYER	Communications type: 0 – IDU_SERIAL, 1- IDU_NETWORK

Return Value

This method will return a long value where 0 indicates success and a non-zero value indicates failure.

IDTestSensor

Test Secugen sensor

Syntax: **object.IDTestSensor Test, COMM_LAYER**

The **IDTestSensor** method syntax has these parts:

Part	Description
Object	An object expression that evaluates to an instantiation of the IDU interface class
Test	Structure to specify what to test
COMM_LAYER	Communications type: 0 – IDU_SERIAL, 1- IDU_NETWORK

Return Value

This method will return a long value where 0 indicates success and a non-zero value indicates failure.

IDUFirmwareVersion

Firmware version on the IDU device

Syntax: **object.IDUFirmwareVersion Version, COMM_LAYER**

The **IDUFirmwareVersion** method syntax has these parts:

Part	Description
Object	An object expression that evaluates to an instantiation of the IDU interface class
Version	Version of firmware retrieved as string
COMM_LAYER	Communications type: 0 – IDU_SERIAL, 1- IDU_NETWORK

Return Value

This method will return a long value where 0 indicates success and a non-zero value indicates failure.

IDUType

Returns the IDU type

Syntax: **object.IDUType IType, COMM_LAYER**

The **IDUType** method syntax has these parts:

Part	Description
Object	An object expression that evaluates to an instantiation of the IDU interface class
IType	Version of IDU: 3, 4 or 5
COMM_LAYER	Communications type: 0 – IDU_SERIAL, 1- IDU_NETWORK

Return Value

This method will return a long value where 0 indicates success and a non-zero value indicates failure.

ListenForTimeStamps

Listens for time stamps send by the IDU devices

Syntax: **object.ListenForTimeStamps Listen**

The **ListenForTimeStamps** method syntax has these parts:

Part	Description
Object	An object expression that evaluates to an instantiation of the IDU interface class
Listen	Put program into listen mode, all timestamps send to the computer will be captured

Return Value

This method will return a long value where 0 indicates success and a non-zero value indicates failure.

NetSettings

IDU device's network settings

Syntax: **object.NetSettings NetworkSettings, CommandType, COMM_LAYER**

The **NetSettings** method syntax has these parts:

Part	Description
Object	An object expression that evaluates to an instantiation of the IDU interface class
NetworkSettings	Structure with network settings
CommandType	Type of command: 0 – IDU_GET, 1 – IDU_SET
COMM_LAYER	Communications type: 0 – IDU_SERIAL, 1- IDU_NETWORK

Return Value

This method will return a long value where 0 indicates success and a non-zero value indicates failure.

OpenRelay

Opens one of the IDU relays

Syntax: **object.OpenRelay RelayNumber, COMM_LAYER**

The **OpenRelay** method syntax has these parts:

Part	Description
Object	An object expression that evaluates to an instantiation of the IDU interface class
RelayNumber	Relay number, 1 or 2.
COMM_LAYER	Communications type: 0 – IDU_SERIAL, 1- IDU_NETWORK

Return Value

This method will return a long value where 0 indicates success and a non-zero value indicates failure.

PING

Pings the IDU device on network to see if it is available

Syntax: **object.PING sIPadr, TimeOutMs**

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

Part	Description
Object	An object expression that evaluates to an instantiation of the IDU interface class
sIPadr	IP Address to ping
TimeOutMs	Milliseconds before ping timeouts

RelayDelay

Delay before the relay will be opened

Syntax: **object.RelayDelay RelayNumber, TimeUnits, CommandType, COMM_LAYER**

The **RelayDelay** method syntax has these parts:

Part	Description
Object	An object expression that evaluates to an instantiation of the IDU interface class
RelayNumber	Relay number, 1 or 2.
TimeUnits	Number of 500 milli-second periods to delay before the relay is triggered
CommandType	Type of command: 0 – IDU_GET, 1 – IDU_SET
COMM_LAYER	Communications type: 0 – IDU_SERIAL, 1- IDU_NETWORK

Return Value

This method will return a long value where 0 indicates success and a non-zero value indicates failure.

RelayDuration

Time the relay must stay open

Syntax: **object.RelayDuration RelayNumber, TimeUnits, CommandType, COMM_LAYER**

The **RelayDuration** method syntax has these parts:

Part	Description
Object	An object expression that evaluates to an instantiation of the IDU interface class
RelayNumber	Relay number, 1 or 2.
TimeUnits	Number of 500 milli-second periods the relay must be opened for
CommandType	Type of command: 0 – IDU_GET, 1 – IDU_SET
COMM_LAYER	Communications type: 0 – IDU_SERIAL, 1- IDU_NETWORK

Return Value

This method will return a long value where 0 indicates success and a non-zero value indicates failure.

ResetIDU2

Reset the IDU device

Syntax: **object.ResetIDU2 COMM_LAYER**

The **ResetIDU2** method syntax has these parts:

Part	Description
Object	An object expression that evaluates to an instantiation of the IDU interface class
COMM_LAYER	Communications type: 0 – IDU_SERIAL, 1- IDU_NETWORK

Return Value

This method will return a long value where 0 indicates success and a non-zero value indicates failure.

SerialNoGet

Serial number of the IDU device

Syntax: **object.SerialNoGet sSerial, COMM_LAYER**

The **SerialNoGet** method syntax has these parts:

Part	Description
Object	An object expression that evaluates to an instantiation of the IDU interface class
sSerial	Serial number of the IDU device
COMM_LAYER	Communications type: 0 – IDU_SERIAL, 1- IDU_NETWORK

Return Value

This method will return a long value where 0 indicates success and a non-zero value indicates failure.

StatCountersGet

Statistical counters kept by the IDU device on scanned fingers

Syntax: **object.StatCountersGet IAccepted, IRejected, COMM_LAYER**

The **StatCountersGet** method syntax has these parts:

Part	Description
Object	An object expression that evaluates to an instantiation of the IDU interface class
IAccepted	Number of accepted fingerprints
IRejected	Number of rejected fingerprints
COMM_LAYER	Communications type: 0 – IDU_SERIAL, 1- IDU_NETWORK

Return Value

This method will return a long value where 0 indicates success and a non-zero value indicates failure.

TimeRecordsNumGet

Get number of available time records from the IDU device

Syntax: **object.TimeRecordsNumGet NumRecords, COMM_LAYER**

The **TimeRecordsNumGet** method syntax has these parts:

Part	Description
Object	An object expression that evaluates to an instantiation of the IDU interface class

Part	Description
NumRecords	Number of available time records on the IDU device
COMM_LAYER	Communications type: 0 – IDU_SERIAL, 1- IDU_NETWORK

Return Value

This method will return a long value where 0 indicates success and a non-zero value indicates failure.

TimeRecordGet

Get a time record from the IDU device

Syntax: **object.TimeRecordGet** PersonID, DateTime, Direction, COMM_LAYER

The **TimeRecordGet** method syntax has these parts:

Part	Description
Object	An object expression that evaluates to an instantiation of the IDU interface class
PersonID	ID of person retrieved from IDU memory
DateTime	The date and time when the person was accepted
Direction	Direction button: 0 – not pressed, 1 – IN, 2 - OUT
COMM_LAYER	Communications type: 0 – IDU_SERIAL, 1- IDU_NETWORK

Return Value

This method will return a long value where 0 indicates success and a non-zero value indicates failure.

TimeRecordsClear

Clear all time records from the IDU device

Syntax: **object.TimeRecordsClear** COMM_LAYER

The **TimeRecordsClear** method syntax has these parts:

Part	Description
Object	An object expression that evaluates to an instantiation of the IDU interface class
COMM_LAYER	Communications type: 0 – IDU_SERIAL, 1- IDU_NETWORK

Return Value

This method will return a long value where 0 indicates success and a non-zero value indicates failure.

VerifyReply

Clear all time records from the IDU device

Syntax: **object.VerifyReply** PersonID, Accept, COMM_LAYER

The **VerifyReply** method syntax has these parts:

Part	Description
Object	An object expression that evaluates to an instantiation of the IDU interface class
PersonID	ID of person that was verified
Accept	Boolean indicating id person was granted or denied access.
COMM_LAYER	Communications type: 0 – IDU_SERIAL, 1- IDU_NETWORK

Return Value

This method will return a long value where 0 indicates success and a non-zero value indicates failure.

WiegandFacilityCode

Clear all time records from the IDU device

Syntax: **object.WiegandFacilityCode** FacilityCode, COMM_LAYER

The **WiegandFacilityCode** method syntax has these parts:

Part	Description
Object	An object expression that evaluates to an instantiation of the IDU interface class
FacilityCode	A value between 0 and 255 if only one facility code is used and 25 bit wiegand cards are used.

Part	Description
COMM_LAYER	Communications type: 0 – IDU_SERIAL, 1- IDU_NETWORK

Return Value

This method will return a long value where 0 indicates success and a non-zero value indicates failure.

WiegandProtocol

Wiegand protocol to use

Syntax: **object.WiegandProtocol WiegProtocol, COMM_LAYER**

The **WiegandProtocol** method syntax has these parts:

Part	Description
Object	An object expression that evaluates to an instantiation of the IDU interface class
WiegProtocol	Wiegand protocol to use: 0 – None, 1- 26 Bit, 2 – 44 Bit
COMM_LAYER	Communications type: 0 – IDU_SERIAL, 1- IDU_NETWORK

Return Value

This method will return a long value where 0 indicates success and a non-zero value indicates failure.

EVENTS

AccessBlocked

Public Event AccessBlocked(ByVal IPAddress As String, ByVal PersonID As Long, ByVal DateTime As Date, ByVal Direction As Integer)

AccessDenied

Public Event AccessDenied(ByVal IPAddress As String, ByVal DateTime As Date, ByVal Direction As Integer)

FWProgress

Public Event FWProgress(ByVal Percentage As Integer)

IDUError

Public Event IDUError(ByVal ErrorDescription As String)

MenuSelection

Public Event MenuSelection(ByVal IPAddress As String, OptionSelected As Integer)

OpenOtherRelay

Public Event OpenOtherRelay(ByVal IPAddress As String)

RequestFinger

Public Event RequestFinger(ByVal Finger As FINGER_NUMBER, ByVal ReadRetry As Boolean, ByVal RetryMessage As String)

ServerIdentify

Public Event ServerIdentify(ByVal IPAddress As String, ByVal PersonPin As Double, ByVal FingerMinutiae As String)

TimeRecordGet

Public Event TimeRecordGet(ByVal IPAddress As String, ByVal PersonID As Long, ByVal DateTime As Date, ByVal Direction As Integer, ByVal Pushed As Boolean)

VerifyRequest

Public Event VerifyRequest(ByVal IPAddress As String, ByVal PersonID As Long)

ENUMERATIONS USED BY INTERFACE

COMM_LAYER

SERIAL_RS232 = 0
NETWORK = 1
SERIAL_RS485 = 2

DEVICE_TEST

TEST_ALL = 0
TEST_SENSOR = 1
TEST_FLASHMEM = 2
TEST_FIRMWARE = 3

FINGER_NUMBER

LEFT_LITTLE = 0
LEFT_RING = 1
LEFT_MIDDLE = 2
LEFT_INDEX = 3
LEFT_THUMB = 4
RIGHT_THUMB = 5
RIGHT_INDEX = 6
RIGHT_MIDDLE = 7
RIGHT_RING = 8
RIGHT_LITTLE = 9

IDU_COMMAND_TYPE

IDU_GET = 0
IDU_SET = 1

IMAGE_SIZE

FULL_SIZE = 1
HALF_SIZE = 2
QUARTER_SIZE = 4

RELAY_ACCESS

RELAY_BOTH = 0
RELAY_1 = 1
RELAY_2 = 2

RETURN_CODE

RET_SUCCESS = &H0
RET_COMPORT_ERROR = &HA0
RET_NO_SERIAL_COMMS = &HA1
RET_NO_NETWORK_COMMS = &HA2
RET_INVALID_PERSON_ID = &HA3
RET_FINGER_CANCEL = &HA4
RET_INVALID_PERSON_RECORD = &HA5
RET_INVALID_FIRMWARE_FILE = &HA6
RET_FILE_NOT_FOUND = &HA7
RET_INCORRECT_SERIAL_NO = &HA8
RET_NO_SERIAL_FPU = &HA9
RET_INCORRECT_FIRMWARE = &HAA
RET_NO_BOOTLOADER = &HAB
RET_UNKNOWN_COMM_LAYER = &HAC
RET_NO_485_ADAPTER = &HAD
RET_NO_NETWORK = &HAF
RET_SELECTION_SHORT = &HE0

RET_SELECTION_LONG = &HE1

RET_FPU_FLASH_OPEN = &H1
RET_FPU_SENSOR_OPEN = &H2
RET_FPU_REGISTER_FAILED = &H3
RET_FPU_VERIFY_FAILED = &H4
RET_FPU_ALREADY_REGISTERED_USER = &H5
RET_FPU_USER_NOT_FOUND = &H6
RET_FPU_INVALID_PASSWORD = &H7
RET_FPU_TIMEOUT = &H8
RET_FPU_DB_FULL = &H9
RET_FPU_DB_WRONG_USERID = &HA
RET_FPU_DB_NO_DATA = &HB
RET_FPU_EXTRACT_FAIL = &HC
RET_FPU_MEMALLOC_FAILED = &HD
RET_FPU_SERIAL_OPEN = &HE
RET_FPU_NOT_IMPLEMENTED = &HF
RET_FPU_FUNCTION_FAIL = &H10
RET_FPU_INSUFFICIENT_DATA = &H11
RET_FPU_FLASH_WRITE_ERROR = &H12
RET_FPU_FLASH_READ_ERROR = &H13
RET_FPU_INVALID_PARAM = &H14
RET_FPU_MASTERFP_NOT_FOUND = &H15
RET_FPU_MASTERCOUNT_EXCEED = &H16
RET_FPU_AUTHENTICATION_FAIL = &H17
RET_FPU_ROOT_CANNOT_BE_DELETED = &H18
RET_FPU_ROOT_ALREADY_REGISTERED = &H19
RET_FPU_FPCHANGE_FAILED = &H1A
RET_FPU_IDENTIFY_FAILED = &H1B
RET_FPU_FLASH_ERASE_ERROR = &H1C
RET_FPU_VERIFY_FAKE = &H1D
RET_FPU_TIME_ERROR = &H1E
RET_FPU_SEARCHING_FOR_IDENTIFY = &H1F
RET_FPU_INVALID_USERDATA_SIZE = &H20
RET_FPU_INVALID_USERDATA_ADDRESS = &H21
RET_FPU_MUST_BE_SET_DATA_LENGTH = &H22
RET_FPU_AUTO_ON_MODE = &H23
RET_FPU_NOT_AUTO_ON_MODE = &H24
RET_FPU_SENSOR_READ_ERROR = &H25
RET_FPU_KEY_GEN_ERROR = &H26
RET_FPU_KEY_TIME_OVER = &H27
RET_FPU_CHECKSUM_ERROR = &H28
RET_FPU_CODE_CHECKSUM_ERROR = &H29
RET_FPU_INVALID_FPRECORD = &H30
RET_FPU_UNKNOWN_COMMAND = &HFF

RET_MPU_SUM_ERROR = &HB0
RET_MPU_INVALID_DATA = &HB1
RET_MPU_INVALID_CMD = &HB2
RET_MPU_NO_RECORDS = &HB3
RET_MPU_EXPIRE_CODE = &HB4
RET_MPU_FAILURE = &HBF

RET_IDU_NOIP = &HC0
RET_IDU_NOUDPTIME = &HC1
RET_IDU_NOUDPCMD = &HC2
RET_IDU_NOUDPIDU = &HC3
RET_IDU_INVALID_PASSWORD = &HC4
RET_IDU_INVALID_FW_FILE = &HC5
RET_IDU_INVALID_MAC = &HA012

RET_IDU_INVALID_SUBNET = &HC6
RET_IDU_INVALID_GATEWAY = &HC7
RET_IDU_FILE_NOT_FOUND = &HAC8
RET_IDU_LISTEN_MODE = &HC9
RET_IDU_INSUFFICIENT_DATA = &HCA
RET_IDU_INVALID_DATA_SEND = &HCB
RET_IDU_SERIAL_NUMBER = &HCC
RET_IDU_UNKNOWN_COMMAND = &HCF

SECURITY_LEVEL

LEVEL1_LOWEST = 1
LEVEL2_LOWER = 2
LEVEL3_LOW = 3
LEVEL4_BELOW_NORMAL = 4
LEVEL5_NORMAL = 5
LEVEL6_ABOVE_NORMAL = 6
LEVEL7_HIGH = 7
LEVEL8_HIGHER = 8
LEVEL9_HIGHEST = 9

SWITCH_VALUE

SWITCH_OFF = 0
SWITCH_ON = 1

USER_TYPE

TYPE_ACCESS_CONTROL = 1
TYPE_TIME_ATTENDANCE = 2
TYPE_BOTH = 3
TYPE_LOCAL_ENROLL = 4
TYPE_ACCESS_BLOCKED = 5

USER_VERIFICATION

VERIFY_NO = 0
VERIFY_YES = 1

TYPES/STRUCTURES USED BY INTERFACE

DEVICE_SETTINGS

SECURITY_LEVEL_UNIT	(SECURITY_LEVEL enumeration)
SECURIY_LEVEL_IDENTIFY	(SECURITY_LEVEL enumeration)
EXPOSURE_BRIGHTHNES	(0 – 100, 160 default)
EXPOSURE_GAIN	(0 – 64, 25 default)
EXPOSURE_CONTRAST	(9 – 100, 50 default)
REGISTER_QUALITY	(30 – 100, 50 default)
VERIFY_QUALITY	(10 – 100, 30 default)
USE_HIGH_SECURITY	(0 or 1, 0 default)
USE_SMART_CAPTURE	(0 or 1, 0 default)

NET_SETTINGS

IDUIPAddress	(String, IP address)
IDUMAC	(String, MAC address)
IDUSubnet	(String, Subnet mask)
IDUGateway	(String, gateway address)
IDUPort	(Long, UDP port, 5000 default)
ServerIP	(String, IP address of host PC)
ServerCommandPort	(Long, UDP port of host PC)
ServerTimePort	(Long, UDP port for receiving clocks from IDU)

PERSON_RECORD

PIN	(0 - 1099511627775)
PASSWORD	(0 - 1099511627775)
TYPE	(USER_TYPE enumeration)
VERIFY	(USER_VERIFICATION enumeration)
Relays	(RELAY_ACCESS enumeration)
DISPLAY_NAME	(Maximum 16 Character string)
FINGER0_DATA	(Finger minutiae)
FINGER1_DATA	(Finger minutiae)
FINGER2_DATA	(Finger minutiae)
FINGER3_DATA	(Finger minutiae)
FINGER4_DATA	(Finger minutiae)
FINGER5_DATA	(Finger minutiae)
FINGER6_DATA	(Finger minutiae)
FINGER7_DATA	(Finger minutiae)
FINGER8_DATA	(Finger minutiae)
FINGER9_DATA	(Finger minutiae)