

**Card Dispenser
F1-2000**

REV: V1.0 2014_03

Note:

This Document is for F1-2000 Card dispenser API instructions.

F1-2000 SDK included following files:

F1API.h	Interface declaration file
F1API.lib	F1API.DLL import library
F1API.dll	32 bit dynamic link library

API Reference

F1_Connect

F1_Connect

LONG

WINAPI

```
F1_Connect(  
    IN  DWORD dwPortNumber,  
    IN  DWORD dwBaudRate,  
    IN  BYTE  bAddress,  
    OUT PISSUERHANDLE phIssuer  
);
```

Parameter:

dwPortNumber COM number, valid value: 1 ~ 256.

dwBaudRate Baud rate, Valid value:

1200

2400

4800

9600

19200

38400

bAddress Dispenser address, value : 0~15.

phIssuer Return a connection handle

Return value:

Success, value at 0

Fail, return an error code (refer to ERROR CODE SHEET)

F1_Disconnect

```
LONG  
WINAPI  
F1_Disconnect(  
    IN  ISSUERHANDLE hIssuer  
);
```

Parameter:

hIssuer Quote F1_Connect return handle value

Return value:

Success, value at 0

Fail, return an error code (refer to ERROR CODE SHEET)

F1_SetBaudRate

```
LONG  
WINAPI  
F1_SetBaudRate(  
    IN  ISSUERHANDLE hIssuer,  
    IN  DWORD dwBaudRate  
);
```

Parameter:

hIssuer Quote F1_Connect return handle value

dwBaudRate Baud rate, valid value:
 1200
 2400
 4800
 9600
 19200
 38400

Return value:

Success, value at 0

Fail, return an error code (refer to ERROR CODE SHEET)

F1_Reset

```
LONG  
WINAPI  
F1_Reset(  
    IN  ISSUERHANDLE hIssuer  
);
```

Parameter:

hIssuer Quote F1_Connect return handle value

Return value:

Success, value at 0

Fail, return an error code (refer to ERROR CODE SHEET)

F1_SetCommAddress

To set card dispenser communication address:

```
LONG  
WINAPI  
F1_SetCommAddress(  
    IN  ISSUERHANDLE hIssuer,  
    IN  BYTE        bAddress  
);
```

Parameter:

hIssuer Quote F1_Connect return handle value

bAddress Dispenser address, valid value: 0~15.

Return value:

Success, value at 0

Fail, return an error code (refer to ERROR CODE SHEET)

F1_GetCommAddress

To get dispenser communication address:

```
LONG
WINAPI
F1_GetCommAddress(
    IN  ISSUERHANDLE hIssuer,
    OUT PBYTE pbAddress
);
```

Parameter:

hIssuer Quote F1_Connect return handle value

pbAddress Receive return dispenser address

Return value:

Success, value at 0

Fail, return an error code (refer to ERROR CODE SHEET)

F1_GetStatus

```
LONG
WINAPI
F1_GetStatus(
    IN  ISSUERHANDLE hIssuer,
    OUT PBYTE pbWorkStatus,
    OUT PBYTE pbCardBoxStatus
);
```

Parameter:

hIssuer Quote F1_Connect return handle value

pbWorkStatus Get dispenser working status, value may be following:

WS_READY	Dispenser ready to send card(standby)
WS_DISPENSING	Dispensing cards
WS_DISPENSE_SUCCESS	Dispense card success
WS_DISPENSE_ERROR	Dispense card error
WS_PREDISPENSING	Pre-dispense card

WS_DISABLED_COMMAND Dispense card command not available

pbCardBoxStatus Card stacker status, value may as following:
 CBS_SUFFICIENT has enough cards
 CBS_INSUFFICIENT less card
 CBS_NO_CARD has no card

Return value:

Success, value at 0
 Fail, return an error code (refer to ERROR CODE SHEET)

F1_Dispense

Dispense card out of dispenser

```
LONG
WINAPI
F1_Dispense(
    IN  ISSUERHANDLE hIssuer
);
```

Parameter:

hIssuer Quote F1_Connect return handle value

Return value:

Success, value at 0
 Fail, return an error code (refer to ERROR CODE SHEET)

F1_EnablePredispensing

Enable/disable pre-dispense card function:

```
LONG
WINAPI
F1_EnablePredispensing(
    IN  ISSUERHANDLE hIssuer,
    IN  BOOL        bEnable
);
```

Parameter:

hIssuer Quote F1_Connect return handle value

bEnable Enable/disable pre-dispense card function:

Return value:

Success, value at 0

Fail, return an error code (refer to ERROR CODE SHEET)

ERROR CODE SHEET:

F1_E_DEVICE_UNRECOGNIZED	Not detected device
F1_E_PORT_UNAVAILABLE	The specified COM port does not exist or is occupied by other programs
F1_E_UNKNOWN_ERROR	An internal error is detected, but the source is unknown.
F1_E_INTERNAL_ERROR	Internal consistency check fail (communication error)
F1_E_COMM_TIMEOUT	Specified communication timeout, Time has expired
F1_E_INVALID_HANDLE	Invalid handle provided
F1_E_INVALID_PARAMETER	One or more of the parameters provided cannot be interpreted correctly
F1_E_NO_MEMORY	There is not enough memory to complete the current operation.
F1_E_BUFFER_TOO_SMALL	The buffer that receives the returned data is too small