

USB CASH DRAWER INTERFACE

Introduction

The USB interface module is fitted in the rear of the cash drawer. The interface opens the drawer and reports the open/closed status of the drawer. Power is supplied by the USB port; external power supplies are not required.

Installation

Our USB cash drawer interface is identified as an HID device (Human Interface Device) by the operation system. Windows operating systems will automatically install drivers. An installation program must be downloaded (see Setup and Testing)

Setting DIP Switches on Cash Drawer

There is no need to change the factory settings, which are all "0", signifying: DRAWER NO 1, and drawer status setting = "Normally Open" (NO).** If settings are changed the drawer must be disconnected and then re-connected to the computer to register the change.

DRAWER NO.	<u>DRAWER ID*</u>			<u>DRAWER STATUS**</u>
	SW1	SW2	SW3	SW4
1	0	0	0	0
2	1	0	0	0
3	0	1	0	0
4	1	1	0	0
5	0	0	1	0
6	1	0	1	0
7	0	1	1	0
8	1	1	1	0

0=Off, 1=On

*Up to 8 drawers may be addressed.

**SW4 specifies the type of microswitch in the cash drawer used to read the open / closed status of the drawer.

SW4=0 means NO ("normally open"):switch reads signal (switch closed) when drawer is closed.

SW4=1 means NC ("normally closed"):switch reads signal (switch closed) when drawer is open.

Setup and Testing

An installation program containing a DLL file must be downloaded. The DLL file acts between the POS software and system driver. A test utility is included in the download.

1. Connect Cash Drawer and perform download from www.mscashdrawer.com/support.
2. Select start > Programs >M.S Cash Drawer Controller >M.S Cash Drawer USB Controller.
3. You should be able to trigger the Cash Drawer and register drawer status using the test utility.

POS Software

The following DLL functions are used. An example in Visual Basic may be downloaded from our website.

1. GetDrawerHandle

This function determines if the cash drawer controller has been added to the bus. If so it returns the handle to the controller, else it returns a 0.

Calling Structure in C

ULONG GetDrawerHandle (BYTE drawer_number):

Example:

```
Handle = GetDrawerHandle(0);  
If (handle) drawer_online = TRUE;  
else drawer_online = FALSE;
```

Calling Structure In Visual Basic

Private Declare Fuction GetDrawerHandle Lib "MSPOS_USB.dll" (ByVal Handle As Long) As Integer

Example:

```
Handle = GetDrawerHandle(0)  
If Handle > 0 Then  
    Drawer_Online = TRUE  
Else  
    Drawer_Online = FALSE  
End If
```

2. OpenDrawer

This function opens the cash drawer. You must give the function the handle to the cash drawer you are using. The solenoid will fire **ONLY** if the drawer is seen by the computer to be closed. (see Setting DIP Switches – Drawer Status). The function will return: drawer opened (2), drawer open (3) or, failure (0).

Calling Structure In C

```
int OpenDrawer (ULONG device_handle);
```

Example:

```
Result = OpenDrawer (handle);  
If (!result) drawer_online = 0
```

Calling Structure In Visual Basic

```
Private Declare Function OpenDrawer Lib "MSPOS_USB.dll" (ByVal Handle As Long) As Integer
```

Example:

```
Result = OpenDrawer(Handle)  
If Result = 0 Then Drawer_Online = False
```

3. GetDrawerStatus

This function returns the state of the microswitch in the cash drawer which signals drawer open or closed (see Setting DIP Switches).

You must give the function the handle to the cash drawer you are using. The function will return failure (0), drawer closed (1), or drawer open (2).

Calling Structure In C

```
int GetDrawerStatus (ULONG device_handle);
```

Example:

```
Result = GetDrawerStatus (handle)  
If (result= =2) drawer_open = TRUE;  
Else if (result= =1) drawer_open = FALSE  
Else drawer_online = FALSE;
```

Calling Structure In Visual Basic

```
Private Declare Function GetDrawerStatus Lib "MSPOS_USB.dll" (ByVal Handle As Long) As Integer
```

Example:

```
Result = GetDrawerStatus (Handle)  
If Result = 2 Then  
    Drawer_Open = TRUE  
Else If Result = 1  
    Drawer_Open = FALSE  
Else  
    Drawer_Online = FALSE  
End If
```

4. ReleaseHandle

This function will release the device handle. Call this function when your program is finished using the device or when your program exits

Calling Structure In C

```
int ReleaseHandle (ULONG device_handle);
```

Example:

```
Result = ReleaseHandle (handle);
```

Calling Structure In Visual Basic

```
Private Declare Function ReleaseHandle Lib "MSPOS_USB.dll" (ByVal Handle As Long) As Integer
```

Example:

```
Result = ReleaseDrawerHandle(Handle)
```