



# IntelliPay®

**V400C Managed DLL**

**Version 10.5.0.0**

**Revision Date: June 28, 2023**

**Prerequisites**

- Managed Code Environment
- o Visual Basic
- o .NET
- o C#
- USB Port on Host

**Connection**

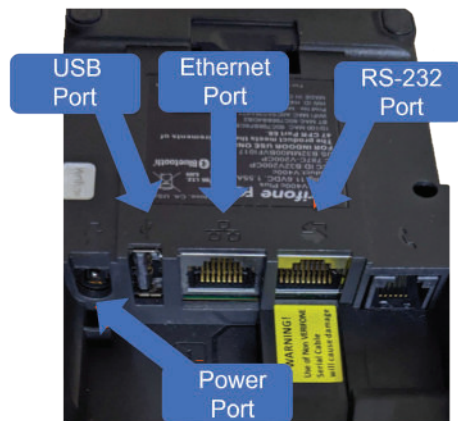
- Use the RJ50 to USB cable. (Provided by IntelliPay to connect the terminal to the Host)
- Take note of the COM port number that the Host assigns to this.

**Abstract**

This DLL will Manage and Open the connection to the V400, The responses are returned to an event handler defined below.



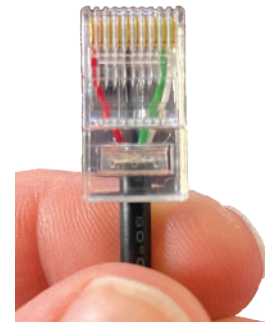
Bottom (back) of V400C



Close-up V400 C Connections



IntelliPay supplied USB to RJ50 cable



Close-up of the RJ50 Connector  
end that plugs into the  
Ethernet port on the Verifone  
V400C terminal

## Open the DLL Connection and Define its event handler

C#

```
IntelliPayDotNet.Terminal terminal = new Terminal("COM6");  
terminal.TerminalResponse += new Terminal.TerminalEventHandler(terminal_TerminalResponse);
```

VB

```
Dim terminal As IntelliPayDotNet.Terminal = new Terminal("COM6");  
terminal.TerminalResponse += New Terminal.TerminalEventHandler(terminal_TerminalResponse);
```

## Define an Event Handler

(Invoke if needed)

C#

```
private void terminal_TerminalResponse(object sender, IntelliPayDotNet.TerminalEventArgs e)  
{  
    string rsp = e.tsdData;  
    if (InvokeRequired)  
    {  
        BeginInvoke(new Action(() =>  
        {  
            txtRsp.Text = rsp;  
        }  
        )));  
    }  
    else  
    {  
        txtRsp.Text = rsp;  
    }  
}
```

VB

```
private Sub terminal_TerminalResponse(ByVal sender As object, ByVal e As IntelliPayDotNet.TerminalEventArgs)  
    DIM rsp = e.tsdData;  
    if InvokeRequired Then  
    {  
        BeginInvoke(new Action( Function()  
            txtRsp.Text = rsp;  
            End Function))  
    }  
    Else  
        txtRsp.Text = rsp;  
    End If  
End Sub
```

**SALE**

```
terminal.SALE("Amount");
```

Example

```
terminal.SALE("1.00");
```

Response Example

```
{
  "Response": {
    "Operation": "SALE",
    "ResponseCode": "",
    "ResponseMsg": "SALE COMPLETED",
    "PaymentID": "37736030",
    "Amount": "1.00",
    "Tip": "",
    "Invoice": "",
    "Clerk": "",
    "PaymentItem": "",
    "Card": {
      "Entry": "CHIP",
      "Brand": "Visa",
      "Last4": "8491"
    }
  },
  "RawResponse": "Will be the response from the Gateway"
}
```

## VOID

```
terminal.VOID("Payment ID");
```

Example

```
terminal.VOID("37736030");
```

### Response Example

```
{
  "Response": {
    "Operation": "VOID",
    "ResponseCode": "",
    "ResponseMsg": "VOID COMPLETED",
    "PaymentID": "37736030",
    "Amount": "",
    "Tip": "",
    "Invoice": "",
    "Clerk": "",
    "PaymentItem": "",
    "RawResponse": "Will be the response from the Gateway"
  }
}
```

## REFUND

```
terminal.REFUND("Amount", "Payment ID");
```

Example

```
terminal.REFUND("1.00", "37735158");
```

### Response Example

```
{
  "Response": {
    "Operation": "REFUND",
    "ResponseCode": "",
    "ResponseMsg": "REFUND COMPLTD",
    "PaymentID": "37735158",
    "Amount": "1.00",
    "Tip": "",
    "Invoice": "",
    "Clerk": "",
    "PaymentItem": "",
    "RawResponse": "Will be the response from the Gateway"
  }
}
```

## CLEAR

terminal.CLEAR(**bool Suppress**);

Example

terminal.REFUND(**False**);

## Response Example

```
{
  "Response": {
    "Operation": "CLEAR",
    "ResponseCode": "",
    "ResponseMsg": "CLEAR",
    "PaymentID": "37735158",
    "Amount": "1.00",
    "Tip": "",
    "Invoice": "",
    "Clerk": "",
    "PaymentItem": "",
    "RawResponse": "Will be the response from the Gateway"
  }
}
```

## Errors

If there are any errors opening up the port specified, it will be returned in a standard string with the error.

## Sale Error

```
{
  "Response": {
    "Operation": "SALE",
    "ResponseCode": "Error",
    "ResponseMsg": "The port 'COM6' does not exist.",
    "PaymentID": "",
    "Amount": "",
    "Tip": "",
    "Invoice": "",
    "Clerk": "",
    "PaymentItem": "",
    "Card": {
      "Entry": "",
      "Brand": "",
      "Last4": ""
    },
    "RawResponse": "Will be the response from the Gateway"
  }
}
```

**Void Error**

```
{
  "Response": {
    "Operation": "VOID",
    "ResponseCode": "Error",
    "ResponseMsg": "The port 'COM6' does not exist.",
    "PaymentID": "",
    "Amount": "",
    "Tip": "",
    "Invoice": "",
    "Clerk": "",
    "PaymentItem": "",
    "RawResponse": "Will be the response from the Gateway"
  }
}
```

**Refund Error**

```
{
  "Response": {
    "Operation": "RFUND",
    "ResponseCode": "Error",
    "ResponseMsg": "Error Message",
    "PaymentI": "37735158",
    "Amount": "1.00",
    "Tip": "",
    "Invoice": "",
    "Clerk": "",
    "PaymentItem": "",
    "RawResponse": "Will be the response from the Gateway"
  }
}
```

## Classes

### Sale Return Class

```
public class Card
{
    public string Entry { get; set; }
    public string Brand { get; set; }
    public string ExpDate { get; set; }
    public string Last4 { get; set; }
}

public class Response
{
    public string Operation { get; set; }
    public string ResponseCode { get; set; }
    public string ResponseMsg { get; set; }
    public string PaymentID { get; set; }
    public string Amount { get; set; }
    public string Tip { get; set; }
    public string Invoice { get; set; }
    public string Clerk { get; set; }
    public string PaymentItem { get; set; }
    public Card Card { get; set; }
}

public class Root
{
    public Response Response { get; set; }
}
```

### VOID Class

```
public class Response
{
    public string Operation { get; set; }
    public string ResponseCode { get; set; }
    public string ResponseMsg { get; set; }
    public string PaymentID { get; set; }
    public string Amount { get; set; }
    public string Tip { get; set; }
    public string Invoice { get; set; }
    public string Clerk { get; set; }
    public string PaymentItem { get; set; }
}

public class Root
{
    public Response Response { get; set; }
}
```



**REFUND Class**

```
public class Response
{
    public string Operation { get; set; }
    public string ResponseCode { get; set; }
    public string ResponseMsg { get; set; }
    public string PaymentI { get; set; }
    public string Amount { get; set; }
    public string Tip { get; set; }
    public string Invoice { get; set; }
    public string Clerk { get; set; }
    public string PaymentItem { get; set; }
}

public class Root
{
    public Response Response { get; set; }
}
```