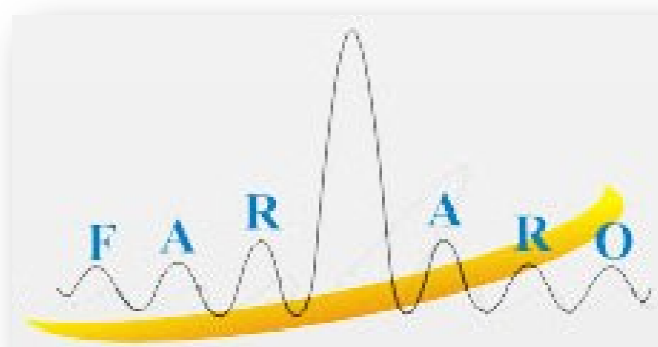


راهنمای استفاده از کتابخانه ارتباطی خانواده PAC در سی شارپ C#  
( PacTransaction.dll )



[www.fararopaya.com](http://www.fararopaya.com)

این کتابخانه به منظور ارتباط با دستگاه‌های خانواده PAC از طریق شبکه اترنت و شبکه RS-485 طراحی شده است و در این نوشتار نحوه استفاده از آن شرح داده خواهد شد. طراحی این کتابخانه بر مبنای پروتکل ارتباطی خانواده PAC بوده است که برای آگاهی از جزئیات آن می‌توانید به آدرس زیر مراجعه کنید :

[http://www.fararopaya.com/UserImage/PAC\\_Transaction\\_Protocol.pdf](http://www.fararopaya.com/UserImage/PAC_Transaction_Protocol.pdf)

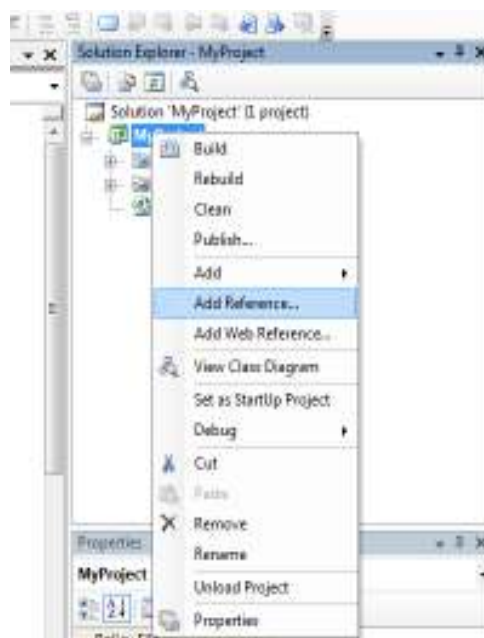
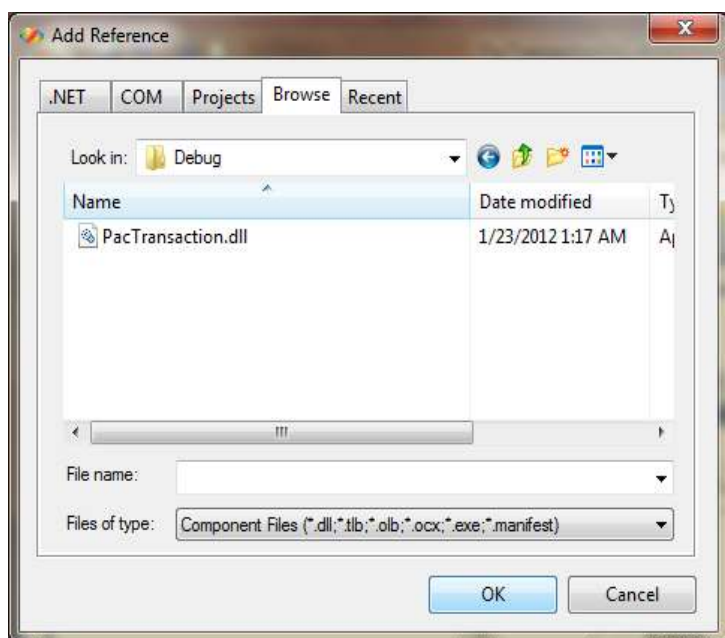
کدهای این کتابخانه و نیز یک مثال که با استفاده از همین کتابخانه نوشته شده از آدرس‌های زیر قابل دسترس است:

<http://www.fararopaya.com/UserImage/...>

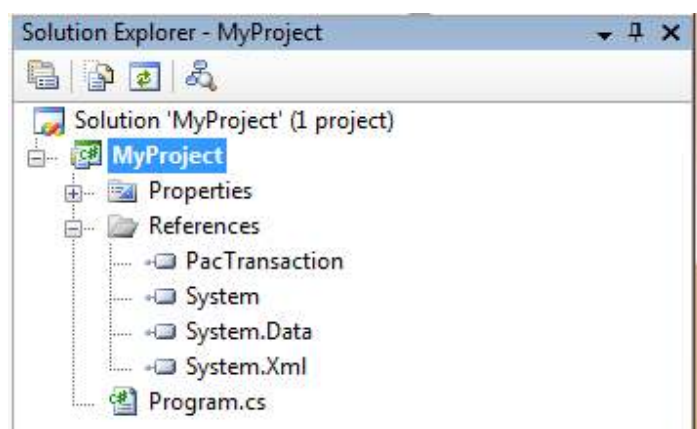
<http://www.fararopaya.com/UserImage/...>

نحوه استفاده:

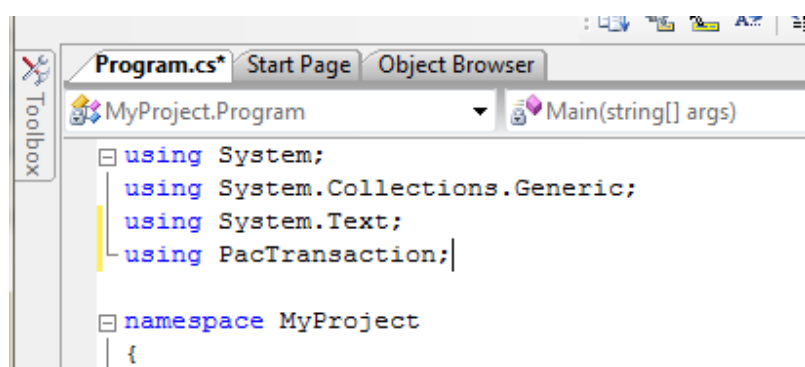
برای استفاده از این کتابخانه ابتدا بر روی نام پروژه کلیک راست کرده و از بخش Add Reference...\Browse آن را به پروژه خود اضافه کنید.



پس از اضافه شدن کتابخانه باید آن را در پوشه References مشاهده کنید.



سپس در ابتدای برنامه فضای نام PacTransaction را فراخوانی نمایید.



## ساختار توابع:

ابتدا یک شی از کلاس PAC ایجاد نمایید:

```
PAC MyPac = new PAC();
```

این شی شامل ۵ تابع SerialRead ، SerailWrite ، TcpRead ، TcpWrite و CRC16\_Ram است که در ادامه نحوه استفاده از کدام توضیح داده شده است.

### : SerialRead

این تابع ۵ ورودی دارد و خروجی آن یک آرایه بایتی است که فریم فرستاده شده از دستگاه به PC است.

```
byte[] SerialRead(string PortName, int BaudRate, byte DestinationID,
    UInt16 DesMemoryAddress, byte DataLength)
```

PortName : نام پورت سریالی که دستگاه از طریق آن به PC وصل شده است. این نام از Control Panel کامپیوتر قابل دسترس است.

BaudRate : نرخ تبادل داده بین دستگاه PAC و PC است که در صورت متوقف بودن دستگاه 38400 می باشد و در غیر اینصورت مقدار همانی است که از طریق برنامه FBD و هنگام پروگرام کردن دستگاه تعیین شده است.

DestinationID : در این بایت، شماره ID دستگاهی که قرار است داده از حافظه آن خوانده شود قرار می گیرد. این بایت مقداری از 0x01 تا 0xFF خواهد داشت.

DesMemoryAddress : آدرس حافظه ای که می خواهیم از آن داده را بخوانیم. در صورتیکه بخواهیم چند بایت بخوانیم باید آدرس اولین خانه را وارد کنیم. آدرس مذکور از نوع Word تعریف شده و مقداری از 0x0000 تا 0xFFFF خواهد داشت.

DataLength: طول داده که تعداد بایت‌هایی است که می‌خواهیم از حافظه بخوانیم و حداکثر ۲۵۵ می‌باشد.

مثال:

```
byte[] MyReceiveFrame = MyPac.SerialRead("com8", 38400, 12, 453, 15);
```

### : SerialWrite

این تابع ۵ ورودی دارد و خروجی آن یک آرایه بایتی است که Acknowledge فرستاده شده از دستگاه

به PC است.

```
byte[] SerialWrite(string PortName, int BaudRate, byte DestinationID, UInt16 DesMemoryAddress, byte[] Data)
```

Data: آرایه‌ای بایتی که شامل داده‌هایی است که می‌خواهیم در حافظه دستگاه بنویسم.

مثال:

```
byte[] MyData = {12, 13, 34, 234, 21};  
byte[] MyAcknowledge = MyPac.SerialWrite("com8", 19200, 13, 124, MyData);
```

### : TcpRead

```
byte[] TcpRead(string IP, UInt16 DesMemoryAddress, byte DataLength)
```

IP: آدرس شبکه دستگاه PAC در شبکه اترنت که به این صورت "192.168.1.15" باید وارد شود.

مثال:

```
byte[] MyReceiveFrame = MyPac.TcpRead("192.168.1.12", 100, 23);
```

**: *TcpWrite***

```
byte[] TcpWrite(string IP, UInt16 DesMemoryAddress, byte[] Data)
```

مثال:

```
byte[] MyAcknowledgeTcp = MyPac.TcpWrite("192.168.1.12", 2453, 2, 24, 1,  
4, 45, 56, 234);
```

**: *CRC16\_Ram***

```
UInt16 CRC16_Ram(int Length, byte[] Pointer)
```

مثال:

```
byte[] MyFrame = {34, 2, 24, 1, 4, 45, 56, 234};  
UInt16 MyCrc = MyPac.CRC16_Ram(MyFrame.Length, MyFrame);
```