srp\_dut.OnReceiving += new EventHandler<DataStreamEventArgs>(receiveHandler); 註冊

private void OnSerialReceiving(byte[] res)

{

if (OnReceiving != null)

{

OnReceiving(this, new DataStreamEventArgs(res)); 最後的觸發

}

}

public SerialClient(string port)

{

\_port = port;

\_baudRate = 9600;

\_lastReceive = DateTime.MinValue;

serThread = new Thread(new ThreadStart(SerialReceiving)); 一 Thread

serThread.Priority = ThreadPriority.Normal;

serThread.Name = "SerialHandle" + serThread.ManagedThreadId;

}

private void SerialReceiving()

{

while (true)

{

int count = \_serialPort.BytesToRead;

/\*Get Sleep Inteval\*/

TimeSpan tmpInterval = (DateTime.Now - \_lastReceive);

/\*Form The Packet in The Buffer\*/

byte[] buf = new byte[count];

int readBytes = Receive(buf, 0, count);

if (readBytes > 0)

{

OnSerialReceiving(buf); 呼叫觸發的過程

}

#region Frequency Control

\_PacketsRate = ((\_PacketsRate + readBytes) / 2);

\_lastReceive = DateTime.Now;

if ((double)(readBytes + \_serialPort.BytesToRead) / 2 <= \_PacketsRate)

{

if (tmpInterval.Milliseconds > 0)

Thread.Sleep(tmpInterval.Milliseconds > freqCriticalLimit ? freqCriticalLimit : tmpInterval.Milliseconds);

/\*Testing Threading Model\*/

Diagnostics.Debug.Write(tmpInterval.Milliseconds.ToString());

Diagnostics.Debug.Write(" - ");

Diagnostics.Debug.Write(readBytes.ToString());

Diagnostics.Debug.Write("\r\n");

}

#endregion

}