Subject: Re: About RS232

Posted by nlneilson on Tue, 23 Nov 2010 02:50:48 GMT

View Forum Message <> Reply to Message

Looking at Serial.h and Serial.cpp they were not familiar to what I have previously used.

To help someone else this is what was done.

In my Python and Java code and the reason for doing it this way.

- 1. Loop through the ports from COM4 to COM20 (COM3 is the default for Win printer).
- 2. Loop through the ports to check the baud rate.
- 3. Open each port that returns the NMEA GPS GGA sentence that run concurrently, three is the most tried so far without threads, no problems.

Here is a snip of code to work with the Serial.h and Serial.cpp Thread work;

```
void OpenAction() {
if (!CommPort.Open(6,57600)) Exclamation("Cannot open COM6");
Sleep(1000);
work.Run(THISBACK(Work));
}
void Work(){
char ch;
char buf1[1];
String St1;
while (CommPort.ReadDataWaiting() ) {
    CommPort.ReadData(buf1, 1);
    ch = buf1[0];
    if(ch!="\n' || ch!="\r')St1 << ch;
    if(ch==='\n'){
        Data<<=St1;
        ...
        Sleep(980);
</pre>
```

The COM port and baud is hard coded here, the app will find and insert these parameters.

Note the Sleep(1000); before the work thread is run and Sleep(980);, otherwise the ReadDataWaiting() is empty.

This reads a char at a time but the parameter has to be a char buf, hence the necessity for char buf1[1];

This is very basic but shows how you can get data from a serial port you can work with. There is also a lot of try/catch, if(...), continue, break, parse(...), send(...), swear(), etc..