

# Re: Serial Port hangs in release

---

*Source:*

<http://www.tech-archive.net/Archive/WindowsCE/microsoft.public.windowsce.platbuilder/2007-06/msg00769.html>

---

- *From:* "Martin Howell" <[martin@xxxxxxxxxxxxxxxxxxxxxx](mailto:martin@xxxxxxxxxxxxxxxxxxxxxx)>
  - *Date:* Mon, 25 Jun 2007 11:59:44 +0100
- 

We had a similar problem which was caused by the Debug messages trying to go through the same physical COM port, that had a driver loaded for it.

Ensure that in for your release build, within OEMInitDebugSerial() you do not init any com port that will subsequently have a CE driver loaded.

Martin

"Doug Allender" <[doug.allender@xxxxxxxx](mailto:doug.allender@xxxxxxxx)> wrote in message [news:791FF164-E385-466A-8B2C-1EAF1C40579B@xxxxxxxxxxxxxxxx](mailto:news:791FF164-E385-466A-8B2C-1EAF1C40579B@xxxxxxxxxxxxxxxx)

Hi All

I have a PXA255 platform using 3 serial ports (COM1,2,3). In debug mode I have no problems, however my serial ports move number, COM1 hardware is accessed by debug using address COM3:, COM3 hardware is accessed by COM2:... etc.

My problem occurs when we go to release, when accessing COM3: causes my programs to hang. I have looked at the registry values and these seem OK.

We

also have a modified serial driver for COM1 and COM3, with added IOCTL's so I

can tell the driver is OK, by accessing COM1 instead of COM3 from my code.

any ideas? see below for COM3: registry and code snippet for line which causes crash.

```
REG_SZ Dll xsc1bd_serial.Dll
REG_DWORD IoBase a6300000
REG_DWORD Irq 20
REG_SZ Tsp Unimodem.dll
REG_DWORD IoLen 24
REG_DWORD DeviceArrayIndex 2
REG_SZ Prefix COM
REG_DWORD DeviceType 0
REG_SZ FriendlyName Serial Cable on COM3
REG_BINARY DevConfig 10 00 00 00 05 00 00 00 10 01 00 00 00 4b 00 00 00 00
08 00 00 00 00 00 00
```

## Re: Serial Port hangs in release

Code

```
Settings.DCBlength = sizeof(DCB);
GetCommState(m_hComPort, &Settings);
Settings.BaudRate = lBaud; // current baudrate
Settings.fBinary = TRUE; // binary mode, no EOF check
Settings.fParity = TRUE;//FALSE // parity checking
Settings.fOutxCtsFlow = FALSE; // CTS output flow control
Settings.fOutxDsrFlow = FALSE; // DSR output flow control
Settings.fDtrControl = DTR_CONTROL_DISABLE; // DTR flow controltype
Settings.fDsrSensitivity = FALSE; // DSR sensitivity
Settings.fTXContinueOnXoff = TRUE; // XOFF continues Tx
Settings.fOutX = FALSE; // XON/XOFF out flow control
Settings.fInX = FALSE; // XON/XOFF in flow control
Settings.fErrorChar = FALSE; // error replacement
Settings.fNull = FALSE; // null stripping
Settings.fRtsControl = RTS_CONTROL_DISABLE; // RTS flow control
Settings.fAbortOnError = FALSE; // abort reads/writes on error
//Settings.fDummy2 = 0; // do not use !
Settings.wReserved = 0; // set to zero!
Settings.XonLim = 10; // transmit XON threshold
Settings.XoffLim = 10; // transmit XOFF threshold
Settings.ByteSize = 8; // number of bits/byte, 4-8
Settings.Parity = EVENPARITY;//NOPARITY // 04=no,odd,even,mark,space
Settings.StopBits = ONESTOPBIT; // 0,1,2 = 1, 1.5, 2
Settings.XonChar = 19; // Tx and Rx XON character
Settings.XoffChar = 16; // Tx and Rx XOFF character
Settings.ErrorChar = 0; // error replacement character
Settings.EofChar = (char)-1; // end of input character
Settings.EvtChar = 0; // received event character
// Settings.wReserved1 = 0; // do not use!
// Open serial port
m_szPort.Format(TEXT("COM%d:"),iPort);
// crash occurs here.....
m_hComPort = CreateFile(m_szPort, GENERIC_READ|GENERIC_WRITE,
0,NULL,OPEN_EXISTING,0,NULL);
--
Snr. Des. Eng.
Northern Hi-Tec
```