

Scuola Universitaria Professionale
della Svizzera Italiana

SUPSI

Dipartimento
Tecnologie
Innovative

SpaceLab

Driver per Windows XP

Remo Radaelli

SUPSI SpaceLab e Dispositivi Periferici 03

Manno, 2 maggio 2007

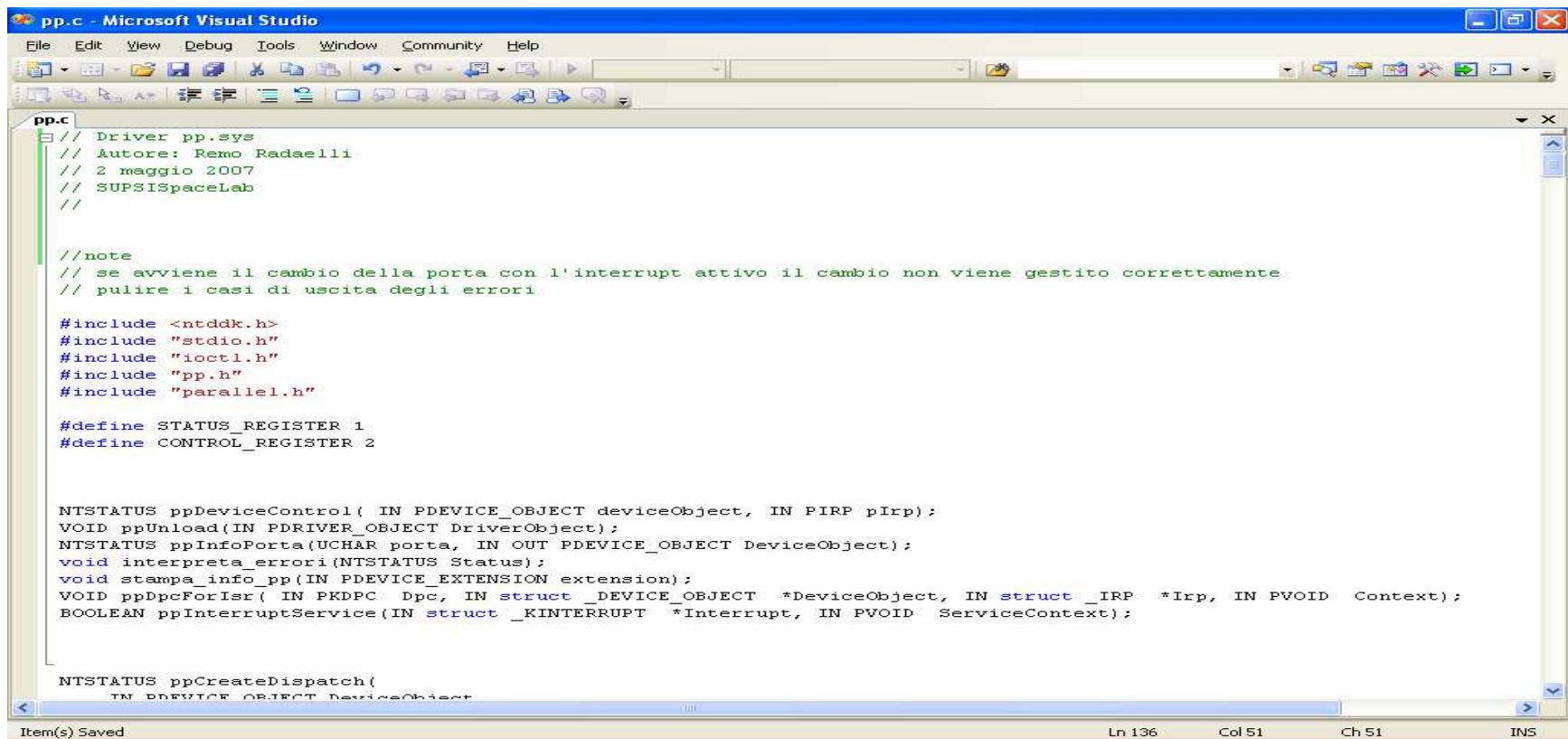
Motivazione

Lo SUPSI SpaceLab desidera poter utilizzare in modo non convenzionale la porta parallela per la gestione di strumentazione nell'acquisizione di dati e controllo, e nell'ambito della mia collaborazione ho costruito un driver XP.

Ho altresì pensato di studiare le problematiche della stesura di un driver al fine di mettere a disposizione le conoscenze acquisite a chi ne fosse interessato.

Driver pp.sys

(vedi codice completo)



```
pp.c - Microsoft Visual Studio
File Edit View Debug Tools Window Community Help
// Driver pp.sys
// Autore: Remo Radaelli
// 2 maggio 2007
// SUPSISpaceLab
//
//note
// se avviene il cambio della porta con l'interrupt attivo il cambio non viene gestito correttamente
// pulire i casi di uscita degli errori
#include <ntddk.h>
#include "stdio.h"
#include "ioctl.h"
#include "pp.h"
#include "parallel.h"
#define STATUS_REGISTER 1
#define CONTROL_REGISTER 2
NTSTATUS ppDeviceControl( IN PDEVICE_OBJECT deviceObject, IN PIRP pIrp);
VOID ppUnload(IN PDRIVER_OBJECT DriverObject);
NTSTATUS ppInfoPorta( UCHAR porta, IN OUT PDEVICE_OBJECT DeviceObject);
void interpreta_errori(NTSTATUS Status);
void stampa_info_pp(IN PDEVICE_EXTENSION extension);
VOID ppDpcForIsr( IN PKDPC Dpc, IN struct_DEVICE_OBJECT *DeviceObject, IN struct_IRP *Irp, IN PVOID Context);
BOOLEAN ppInterruptService(IN struct_KINTERRUPT *Interrupt, IN PVOID ServiceContext);
NTSTATUS ppCreateDispatch(
    IN PDEVICE_OBJECT DeviceObject
```

Item(s) Saved Ln 136 Col 51 Ch 51 INS

Prerequisiti

Preparazione dell'ambiente di compilazione:

- Tools for Windows 32-bit driver development kit (ddk)

• http://download.microsoft.com/download/9/0/f/90f019ac-8243-48d3-91cf-81fc4093ecfd/1830_usa_ddk.iso



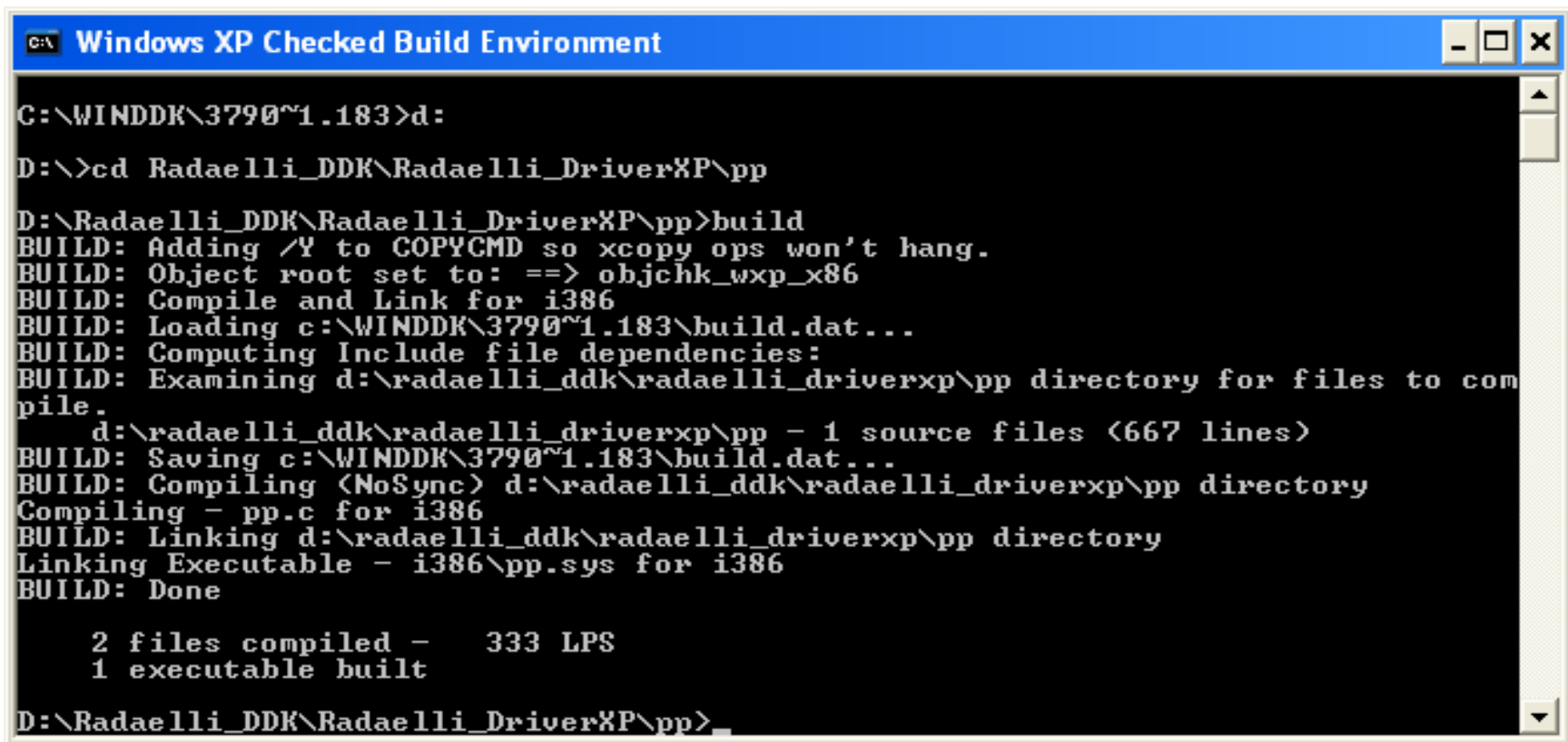
Per lo sviluppo Debugging Tools for Windows

- Windbg
- Dual boot per l'accesso a window in modalità di debug (modifica del boot.ini)
- http://msdl.microsoft.com/download/symbols/debuggers/dbg_x86_6.7.05.0.exe



```
boot.ini - Blocco note
File Modifica Formato Visualizza ?
[boot loader]
timeout=30
default=multi(0)disk(0)rdisk(0)partition(1)\WINDOWS
[operating systems]
multi(0)disk(0)rdisk(0)partition(1)\WINDOWS="Microsoft windows XP Professional"
/noexecute=optin /fastdetect
multi(0)disk(0)rdisk(0)partition(1)\WINDOWS="Microsoft windows XP Professional"
/noexecute=optin /debug
```

Compilazione Driver XP



```
C:\ Windows XP Checked Build Environment
C:\WINDDK\3790~1.183>d:
D:\>cd Radaelli_DDK\Radaelli_DriverXP\pp
D:\Radaelli_DDK\Radaelli_DriverXP\pp>build
BUILD: Adding /Y to COPYCMD so xcopy ops won't hang.
BUILD: Object root set to: ==> objchk_wxp_x86
BUILD: Compile and Link for i386
BUILD: Loading c:\WINDDK\3790~1.183\build.dat...
BUILD: Computing Include file dependencies:
BUILD: Examining d:\radaelli_ddk\radaelli_driverxp\pp directory for files to compile.
      d:\radaelli_ddk\radaelli_driverxp\pp - 1 source files (667 lines)
BUILD: Saving c:\WINDDK\3790~1.183\build.dat...
BUILD: Compiling (NoSync) d:\radaelli_ddk\radaelli_driverxp\pp directory
      Compiling - pp.c for i386
BUILD: Linking d:\radaelli_ddk\radaelli_driverxp\pp directory
      Linking Executable - i386\pp.sys for i386
BUILD: Done

      2 files compiled -      333 LPS
      1 executable built

D:\Radaelli_DDK\Radaelli_DriverXP\pp>
```

Installazione Driver

- Nome driver: *pp*
- Copiare il file *pp.sys* compilato nel fs in *windows\System32\drivers*
- Aprire l'editor del registro di sistema per registrare il nuovo driver:
My computer\HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services
- Creare una nuova chiave *pp* con i seguenti attributi
ErrorControl REG_DWORD 1
Start REG_DWORD 3
Type REG_DWORD 1
- Riavviare il sistema
- Per avviare o chiudere il driver: *net start xyz* o *net stop xyz*

Compilazione pp_dll.dll

```
pp_dll - Microsoft Visual C++ - [pp_dll.cpp]
File Edit View Insert Project Build Tools PureCoverage Purify Quantify Window Help
[Globals] | [All global members] | scriviPorta
Workspace 'pp_dll': 1 project(s)
  pp_dll files
    Source Files
      pp_dll.cpp
      pp_dll.def
      StdAfx.cpp
    Header Files
      ioctl.h
      StdAfx.h
    Resource Files
      ReadMe.txt
    External Dependencies
// Nome del programma: pp_dll.cpp
// Questo è un programma che è l'intermediario fra l'applicazione e il driver "pp.sys"
//
// Autore: Remo Radaelli
// 2 maggio 2007
// SUPSISpaceLab
//
#include "stdafx.h"
#include "ioctl.h"
// #include <winioctl.h> // errore nella libreria!
#include <windows.h>
#include <stdio.h>
// #include "prototipi_dll.h"
//
// PROTOTIPI
short _stdcall selezionaRisorsa(UCHAR num);
short _stdcall numeroDiPorte(PULONG Buffer);
short _stdcall leggiPorta(PUCHAR Buffer);
short _stdcall scriviPorta(UCHAR Numero);
short _stdcall gestioneInterrupt(void);
//
short _stdcall leggiControl(PUCHAR Buffer);
short _stdcall leggiStatus(PUCHAR Buffer);
//
BOOL collegamento();
//
HANDLE hDriver;
//
BOOL WINAPI DllMain( HANDLE hModule,
                    DWORD ul_reason_for_call,
                    LPVOID lpReserved
                    )
{
}

```

Creating library Debug/pp_dll.lib and object Debug/pp_dll.exp
pp_dll.dll - 0 error(s), 0 warning(s)

Build | Debug | Find in Files 1 | Find in Files 2 | Results | SQL Debugging |

Ready | Ln 159, Col 27 | REC | COL | DVR | READ

Test del Driver pp.sys

```
pp_test - Microsoft Visual C++ - [pp_test.cpp]
File Edit View Insert Project Build Tools PureCoverage Purify Quantify Window Help
[Globals] [All global members] main
Workspace 'pp_test': 1 project
  pp_test files
    Source Files
      pp_test.cpp
      Stdafx.cpp
    Header Files
    Resource Files
    ReadMe.txt
    External Dependencies
// Nome del programma: pp_test.cpp
// Questo è un programma di test per il driver XP "pp.sys" di porta parallela
// pp_test.cpp.
// Autore: Remo Radaelli
// 2 maggio 2007
// SUPSISpaceLab

#include "stdafx.h"
#include "stdlib.h"
#include <windows.h>
#include <stdio.h>

extern short _stdcall selezionaRisorsa(UCHAR num); // selezione lpt num
extern short _stdcall numeroDiPorte(PULONG Buffer); // ritorna numerii di lpt disponibili
extern short _stdcall leggiPorta(PUCHAR Buffer);
extern short _stdcall scriviPorta(UCHAR Numero);

extern short _stdcall leggiControl(PUCHAR Buffer);
extern short _stdcall leggiStatus(PUCHAR Buffer);

int main(int argc, char* argv[])
{
    ULONG NumPorte=99;
    UCHAR car;
    char ch=0;

    printf("Solo per test della funzione selezionaRisorsa 0=nessun errore Errore: %d\n", seleziona
    printf("Errore: %d\n", numeroDiPorte(&NumPorte));

    printf("Porte parallele trovate: %d\n", NumPorte);

    // menu di scelta
}

-----Configuration: pp_test - Win32 Debug-----
Linking...
pp_test.exe - 0 error(s), 0 warning(s)
Build Debug Find in Files 1 Find in Files 2 Results SQL Debugging
Ready
```


Funzioni

- Seleziona la porta lpt
short _stdcall selezionaRisorsa(UCHAR num);
- Visualizza il numero di porte lpt trovate
short _stdcall numeroDiPorte(PULONG Buffer);
- Legge un byte sulla porta lpt selezionata (BASE)
short _stdcall leggiPorta(PUCHAR Buffer);
- Scrive un byte sulla porta lpt selezionata (BASE)
short _stdcall scriviPorta(UCHAR Numero);
- Legge il registro status (BASE + 1)
short _stdcall leggiStatus(PUCHAR Buffer);
- Legge il registro control (BASE + 2)
short _stdcall leggiControl(PUCHAR Buffer);