



# **A PCDSP kártya programozása MS Windows alatt**

Csurgai-Horváth László  
BME-HVT  
[csurgai@mht.bme.hu](mailto:csurgai@mht.bme.hu)

# **Windows operációs rendszerek áttekintése**

- **Nem 32 bites Windows verziók**
- **Windows NT**
- **Windows 95**
- **Windows 98/ME**
- **Windows 2000**
- **Windows XP**
- **...**

# Hardware eszközök kezelése Windows alatt

- Windows memory manager
- A felhasználói szinten futó programok virtuális címeket használnak
- Direkt fizikai címek nem érhetők el
- Meghajtóprogramok – drivererek
  - Interface a felhasználói programok és a hardware között
  - Driver szinten a direkt hardware címek elérhetőek

# ISA, PCI eszközök; plug&play

- ISA buszos eszközök
  - Általában jumperrel konfigurálhatóak
  - A rendszer nem ismeri fel automatikusan
  - Manuális installálás szükséges
- ISA Plug & Play – már nem használt
- PCI eszközök
  - Automatikus erőforrás allokáció (PCI BIOS)
  - Automatikus Windows installálás

# PCI eszközök installálása

## Windows alatt

1. Hardware behelyezése a számítógépbe
2. Operációs rendszer indítása
3. Új hardware detektálás – installációs file-ok megadása

Operációs rendszer függés:

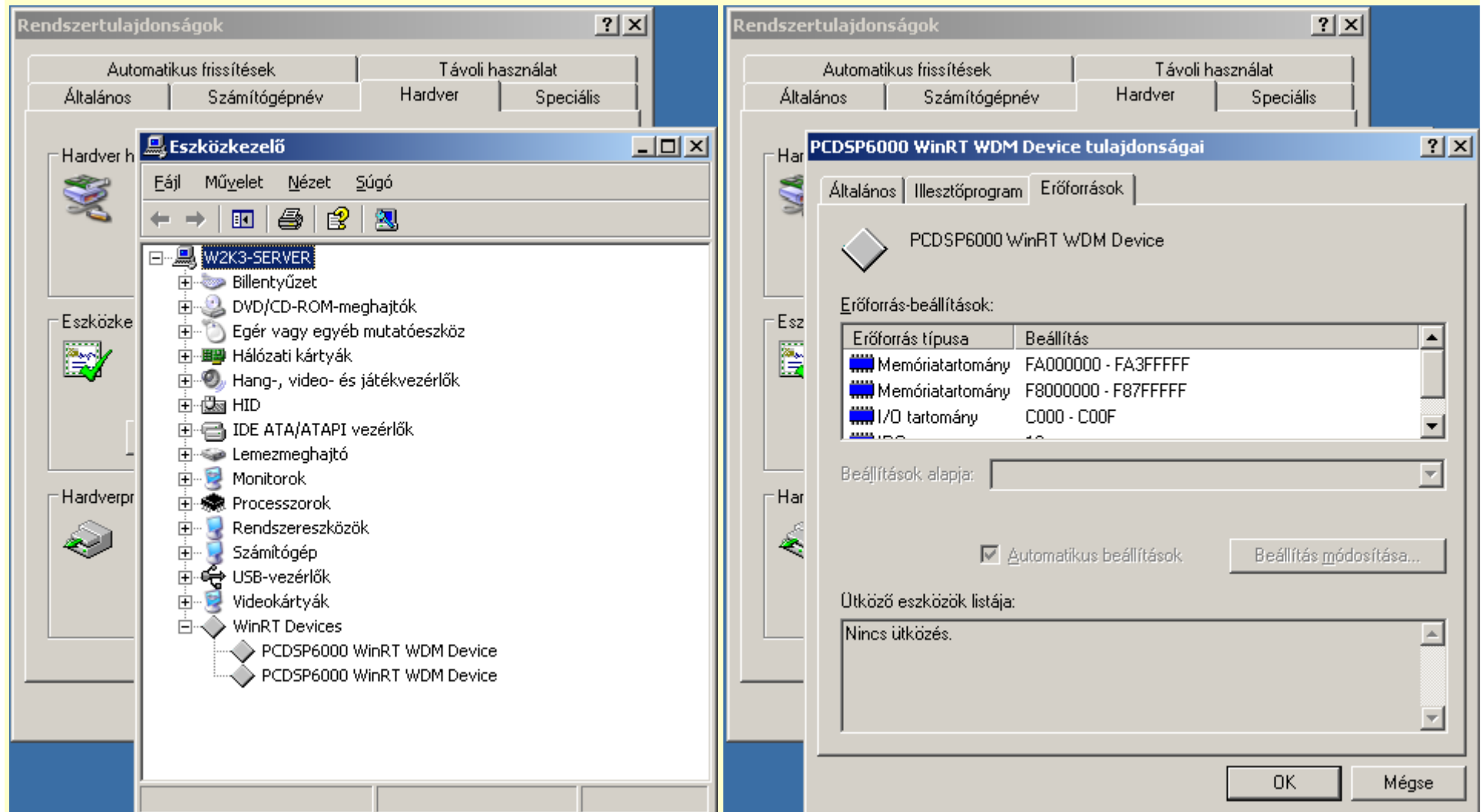
Windows 95: VXD

Windows NT: REG file, nem Plug&Play

Windows 98-ME: WDM modell (Windows Driver Model)

Windows XP: továbbfejlesztett WDM, WDF (Windows Driver Foundation)

# Rendszertulajdonságok ellenőrzése



# Konfiguráció és driver file-ok

.INF file	Information File (txt) (Win95+)
.CAT file	Microsoft digital signature file (Win98+)
.SYS file	Driver code (bin)
(.VXD file	Windows 95)
(.REG file	Windows NT)

# Az INF file szerkezete 1.

[Version]

Signature="\$Windows NT\$" **\$Windows NT\$, \$Windows 95\$, or \$Chicago\$**  
Class=WinRTDevices  
ClassGUID={D695ED6A-630D-4d83-8D8B-F1F0AC107AD0} **registry device setup class subkey**  
Provider=%PCDSP6000.Provider%  
DriverVer=01/31/2002,1.0.0.0  
CatalogFile=PCDSP600.cat

[Manufacturer]

%PCDSP6000.Manufacturer%=PCDSP6000

[ClassInstall32]

AddReg=PCDSP6000.AddClassReg

[PCDSP6000]

%PCDSP6000.DeviceDesc%=PCDSP6000.INSTALL, PCI\VEN\_104C&DEV\_6001 **vendor & device ID**

[ControlFlags]

ExcludeFromSelect=PCI\VEN\_104C&DEV\_6001 **Found New Hardware and Hardware Update wizard displays**

[SourceDisksNames]

1=%PCDSP6000.DriverDiskName%, "",

[SourceDisksFiles]

PCDSP600.sys=1



## Az INF file szerkezete 2.

[DestinationDirs]

DefaultDestDir = 10,system32\drivers ; WINDOWS\SYSTEM32\DRIVERS

PCDSP6000.CopyFiles = 10,system32\drivers ; WINDOWS\SYSTEM32\DRIVERS

[PCDSP6000.INSTALL]

CopyFiles=PCDSP6000.CopyFiles

AddReg=PCDSP6000.AddRegLoader

AddReg=PCDSP6000.AddRegWinRT

[PCDSP6000.INSTALL.NT]

CopyFiles=PCDSP6000.CopyFiles

[PCDSP6000.INSTALL.NT.HW]

AddReg=PCDSP6000.AddRegWinRT

[PCDSP6000.INSTALL.HW]

AddReg=PCDSP6000.AddRegWinRT

[PCDSP6000.AddClassReg]

HKR,,, %PCDSP6000.ClassName%

[PCDSP6000.AddRegLoader]

HKR,,DevLoader,,\*ntkern

HKR,,NTMPDriver,,PCDSP600.sys

## Az INF file szerkezete 3.

[PCDSP6000.AddRegWinRT]

HKR,,DMACommonBufferSize,0x00010001,4096

HKR,,DebugBitMask,0x00010001,7

HKR,,BusType,0x00010001,5

HKR,,SymbolicLinkName,0,"PCDSP6000"

WinRT registry section

DMA buffer size

[PCDSP6000.INSTALL.NT.Services]

AddService = PCDSP6000,0x00000002,PCDSP6000\_Service\_Inst,PCDSP6000\_EventLog\_Inst

[PCDSP6000\_Service\_Inst]

ServiceType = 1

StartType = 3 ; Start type is irrelevant on WDM

ErrorControl = 1

ServiceBinary = %12%\PCDSP600.sys

DisplayName = %PCDSP6000.DeviceDesc%

## Az INF file szerkezete 4.

[PCDSP6000\_EventLog\_Inst]

AddReg = PCDSP6000\_EventLog\_AddReg

[PCDSP6000\_EventLog\_AddReg]

HKR,,EventMessageFile,0x00020000,"%%SystemRoot%%\System32\loLogMsg.dll;%%SystemRoot%%\System32\drivers\PCDSP600.sys"

HKR,,TypesSupported,0x00010001,7

[PCDSP6000.CopyFiles]

PCDSP600.sys

[Strings]

**device manager-ben megjelenő értékek**

PCDSP6000.Provider="BSQUARE CORPORATION"


PCDSP6000.Manufacturer="BSQUARE CORPORATION"

PCDSP6000.DeviceDesc="PCDSP6000 WinRT WDM Device"

PCDSP6000.ClassName="WinRT Devices"

PCDSP6000.DriverDiskName="PCDSP6000 WinRT WDM Installation Diskette"

# A Windows driver fejlesztés lehetséges módjai, eszközei, DDK és third-party eszközök

- Windows DDK – Driver Development Kit
    - C programozási nyelv
  - NuMega DriverWorks
    - C++ fejlesztés
  - Jungo Windriver
    - C++ nyelv
  - Bsquare – WinRT
    - C nyelv
- 
- + C/C++ compiler

# A PCDSP driver áttekintése

- Kapcsolatot biztosít a hardware és a felhasználói program között
- WinRT fejlesztőrendszerrel készült
- Részei:
  - PCDSP600.SYS - Kernel-Mode driver
  - PCDSP6000.DLL - Driver API DLL
  - PCDSP6000.H - Driver API header file
  - PCDSP6000.LIB - Driver API library file

# A PCDSP driver használata

- Felhasználói program írása C/C++ nyelven
- PCDSP6000.H header file és PCDSP6000.LIB használata fordításnál
- PCDSP6000.DLL szükséges a futtatáshoz

# Függvények és hívásaik 1

## Általános függvények

### Kártya megnyitás:

```
int OpenBoard6000( int device_id, int number, BoardConfig6000 *boardconfig );
```

6001/6002      0-5

### Kártya bezárás:

```
BOOL CloseBoard6000 (PCDSPHANDLE hBoard, LPSTR szBuf);
```

boardconfig.device\_no

### Hibakezelés:

```
BOOL GetErrorStatus6000 ( PCDSPHANDLE hBoard, LONG *ErrorCode, LPSTR  
ErrorString );
```

# **Függvények és hívásaik 2**

## **Regiszter és memória elérés, I/O**

### **BASE 0/1 olvasás:**

```
uint32 GetBASE06000 ( PCDSPHANDLE hBoard, uint32 offset, BOOL  
    byteaddress );
```

```
uint32 GetBASE16000 ( PCDSPHANDLE hBoard, uint32 offset, BOOL  
    byteaddress );
```

### **BASE 0/1 írás:**

```
void SetBASE06000 ( PCDSPHANDLE hBoard, uint32 offset, uint32 data,  
    BOOL byteaddress );
```

```
void SetBASE16000 ( PCDSPHANDLE hBoard, uint32 offset, uint32 data,  
    BOOL byteaddress );
```



# Függvények és hívásaik 3

## IRQ

### Counter IRQ handler installálás:

```
void InstallCounterIRQHandler6000(PCDSPHANDLE hBoard, IRQSetup6000 *irqsetup);
```

### Counter lekérdezés:

```
void GetIRQCounter6000(PCDSPHANDLE hBoard, IRQSetup6000 *irqsetup);
```

### Callback IRQ handler installálás:

```
void InstallCallbackIRQHandler6000 (PCDSPHANDLE hBoard, isr_t handler);
```

### Callback függvény:

```
void (*isr_t)();
```

### IRQ handler eltávolítás:

```
void RemoveIRQHandler6000(PCDSPHANDLE hBoard);
```

# Függvények és hívásaik 4

## DMA

### DMA handler installálás:

```
BOOL InstallDMA6000(PCDSPHANDLE hBoard, DMASetup6000 *dmasetup, BOOL fill,  
    UCHAR fill_byte);
```

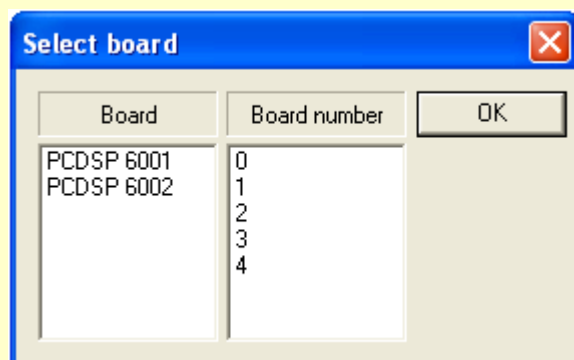
### DMA handler eltávolítás:

```
BOOL DeInstallDMA6000(PCDSPHANDLE hBoard);
```

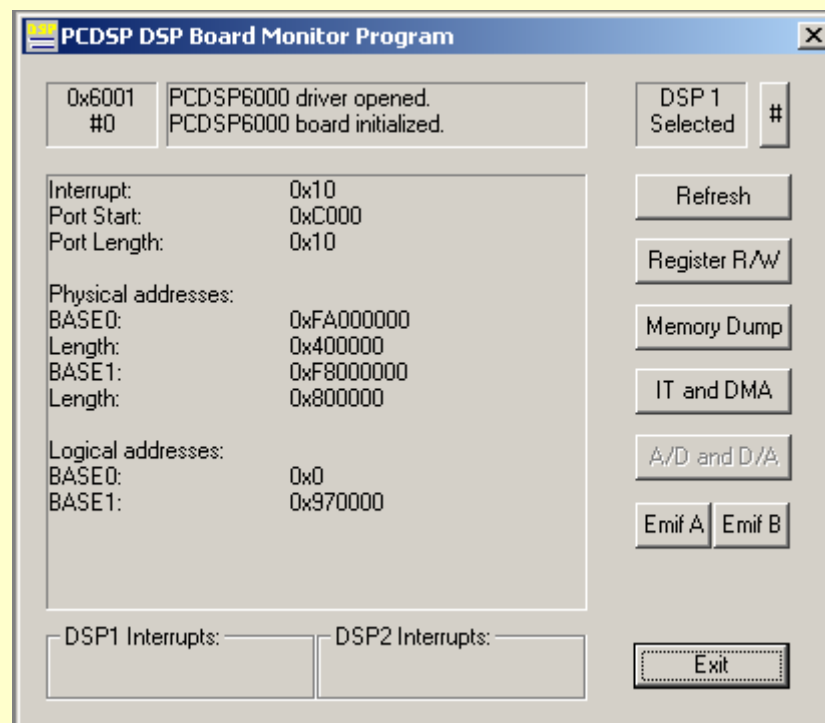
# A PCDSP monitorprogram 1.

## Egy alkalmazás a driver használatára

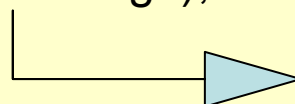
Kártya kiválasztás:



Főablak:



```
OpenBoard6000(0x6001,1, &boardconfig1); hDevice = boardconfig1.device_no;  
OpenBoard6000(0x6001,2, &boardconfig2);
```

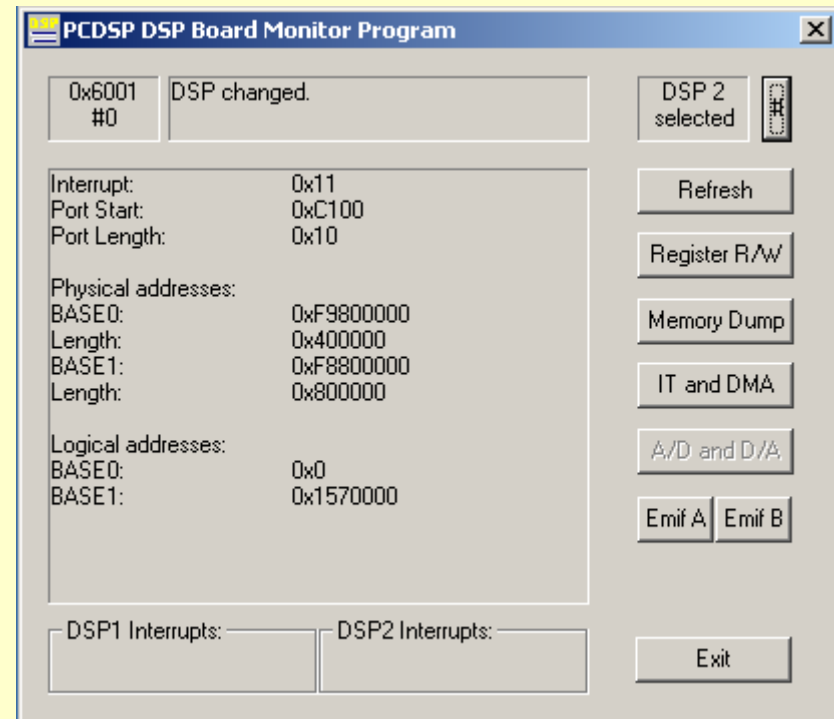


Kártya paraméterek

# A PCDSP monitorprogram 2.

## DSP váltás:

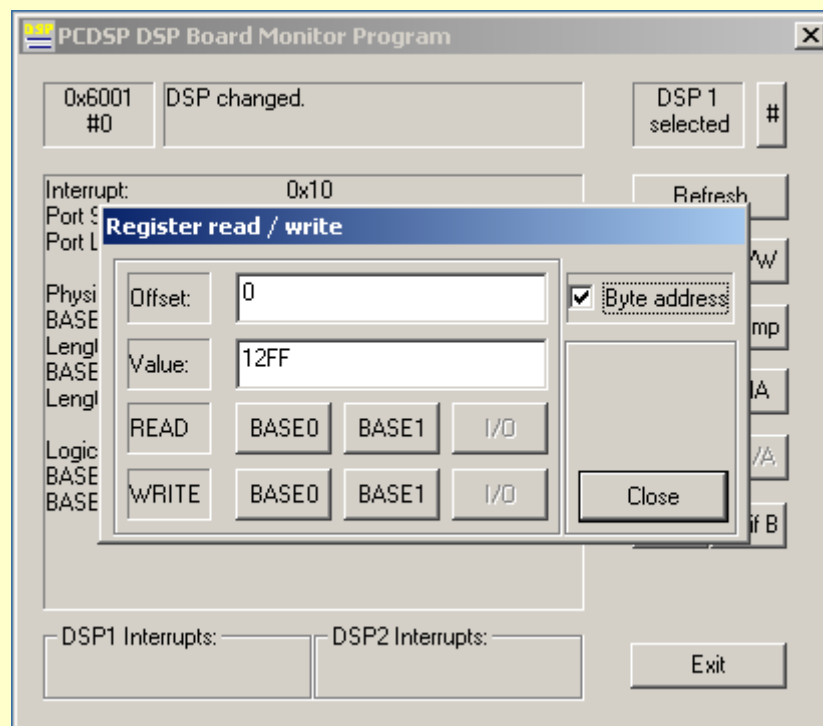
```
hDevice = boardconfig1.device_no;  
/  
hDevice = boardconfig2.device_no;
```



# A PCDSP monitorprogram 3.

## Regiszter írás/olvasás:

```
if(!m_byteaddress)
    rval = GetBASE06000 ( hDevice, (ULONG)offset/4, FALSE );
else
    rval = GetBASE06000 ( hDevice, (ULONG)offset, TRUE );    BYTE cím
```



# A PCDSP monitorprogram 4.

Memory dump/edit:  
DSP BASE 0

DSP Memory Dump / Edit

DSP BASE 0:

@00000000	C54927B3	775F34E5	DB64D5F7	EDDA4AB9	54998EED	58DCF679	8D9EBCA0	7F8E41F9
@00000020	3FECB3F3	FD9DB888	9F36FBBB	D069B99A	5D7F2ED3	5D5F57DF	4E6BEAFF	D19D12F4
@00000040	AD6E66AF	F6BA41EB	FD6DD82D	E8AE1BD5	2A32BEE5	1F7B754E	3A3B7F79	DF55F3BA
@00000060	24AAD365	7B6FBAAF	33974CA9	9EF6FE1F	1CEB7EAF	DCE7B547	C08BD46D	FA7CD3E6
@00000080	C73E2CBB	9A9E79DF	BEED3FE7	3455FBF3	FF596EF6	382A43AB	BE694FDB	F4773FBD
@000000A0	03EFD86F	FA1EC0F6	EA30FFDE	7C7637BF	3AFF6673	1F3DF5F8	FAFBB2D6	7FD677EB
@000000C0	6E6D456B	70B2C5D7	378EEEF9	70532F3F	361DF06F	168EBF16	3BB6959E	7AFDF1C5
@000000E0	F2BB7DF3	4BC7BE83	D14CF5EA	67CDD7C0	2CFB92BD	04C621EB	A6BBCBCB	C0F4BB7B
@00000100	EBBF73BB	3FD3EEDF	6ED8DBFF	97CCA6B7	55FCD7B7	AFE5F2FA	4127D27E	4CE7E5AD
@00000120	1F5FA637	779D5158	FB37770D	D15BFFB3	C6C2657B	EBFB338E	FFB99FC5	FFB335B4
@00000140	F12E09E9	A7626DDF	19733A2A	AFA59EF3	F3DA920C	986FFF9B	4DDB774A	F32F9BF9
@00000160	D78E71CD	53407E77	57543F8D	D4B3F5D5	26E5DB1F	7DEFA67B	B89FB3F3	B6F90B9A
@00000180	9FC575C0	3ABFFE4B	1E8DD3DB	7F3E2EB4	DBD0F71A	DFDEFFEB	23A3F6A0	956DFB5B
@000001A0	6B273E31	D11AA5FF	D0DE7B78	EF3FAECF	E09FDDBF	C5EDED66	C7FB36FC	719353FE
@000001C0	F5DA2D91	DE5ED3EB	F039F38A	A3FB02FB	B6B5ACAC	9238CE59	6275B77A	9AAE4F7D
@000001E0	6DCB15E7	CFAEDE1A	82C8793E	66617B7F	60474CBD	5F2F9B6C	861202DF	9578EC87

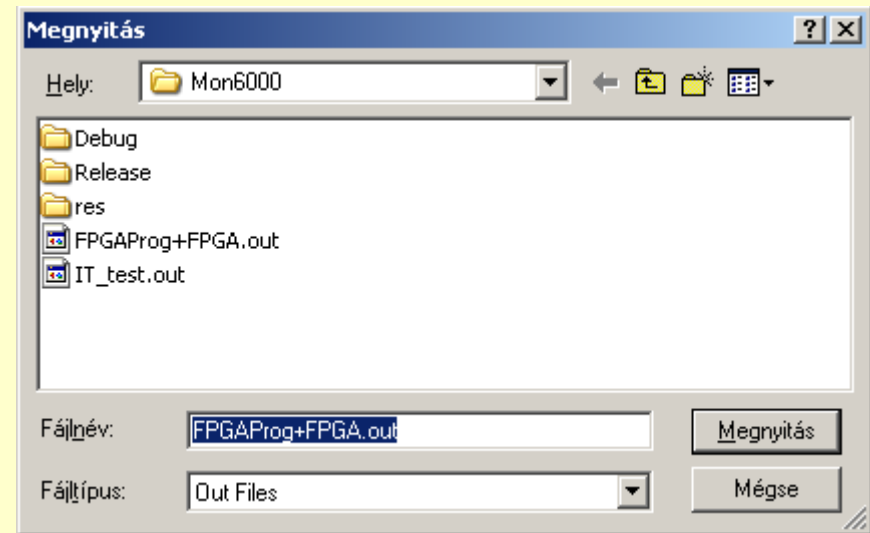
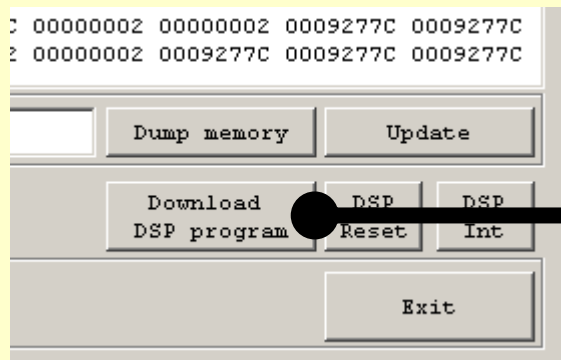
DSP: 0    Update    Offset: 0    Dump memory    Update

BASE 0    BASE 1    Download DSP program    DSP Reset    DSP Int

DMA Memory    Exit

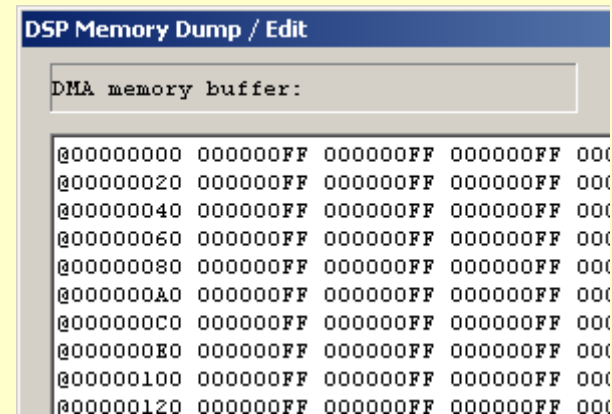
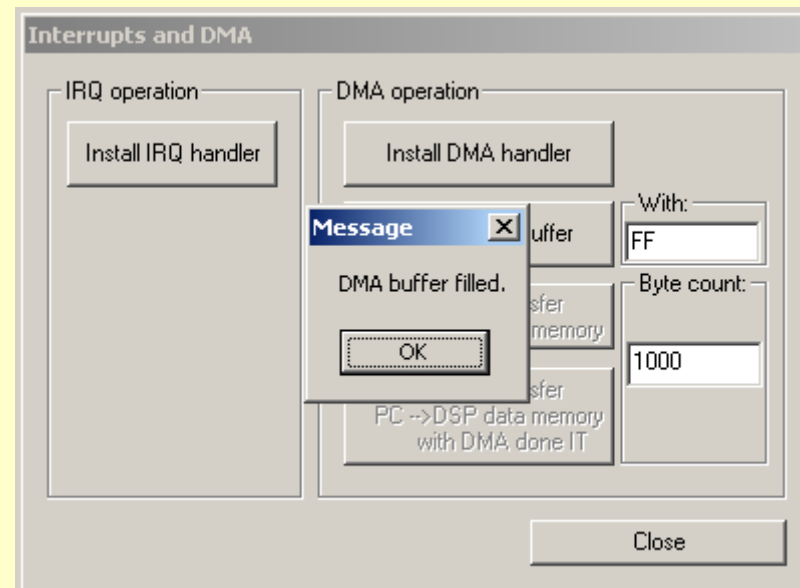
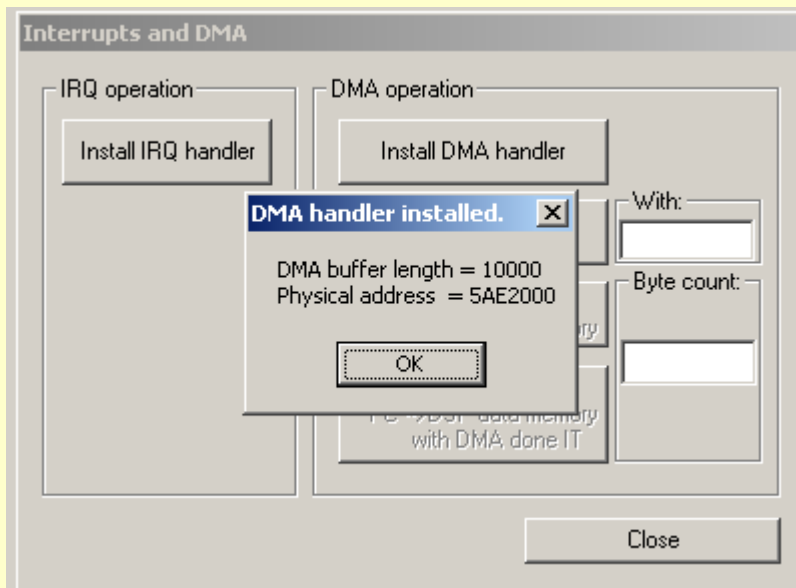
# A PCDSP monitorprogram 5.

DSP program letöltése:



# A PCDSP monitorprogram 6.

## IRQ és DMA buffer foglalás / feltöltés:





# **Windows C/C++ programfejlesztési eszközök: API, MFC**

- MS Visual Studio 6.0 / .NET
- MFC
- Borland C/C++