

## Register for the "Try WinCE" Program

The main goal of this contest is to generate interest in the WinCE versions of MOXA's ready-to-run embedded computers. There are two steps to the registration procedure:

<b>Step 1:</b>	Provide us with your contact information. → <a href="#">Click here to complete Step 1</a> .
<b>Step 2:</b>	Create and execute a "Hello" program on MOXA's UC-7420-CE embedded computer over the Internet.

### Step 2: Create and execute a simple "Hello" program

Step 2-1. Install an IDE tool on your workstation

Step 2-2. Develop a "Hello" program

Step 2-3. Upload and execute your application on UC-7420-CE

#### Step 2-1. Install an IDE tool on your workstation

Thanks to well-known tools from Microsoft, developing applications on a UC-7420-CE embedded computer is easy and straightforward. The tools include eMbedded Visual C++ (eVC) 4.0 and Visual Studio 2005, which are easy for windows programmers to use.

#### C/C++ Applications: Use eMbedded Visual C++ (eVC) 4.0

The eVC 4.0 software is free and can be downloaded directly from Microsoft's download web pages. First install eVC 4.0, import the service pack, and then install MOXA's C/C++ SDK:

1. Install eMbedded Visual C++ 4.0 (230 MB)
2. Install Service Pack 4 for eVC 4.0 (68 MB)
3. Click here to download the SDK file UC7400CE\_SDK\_1.0.msi (28 MB), and then follow the MOXA SDK Installation Instructions:

MOXA SDK Installation Instructions	
1.	Double click on the MSI file UC7400CE_SDK_1.0.msi to start the installation process, and then click Next.
2.	If an error message appears, click Close to ignore the message.
3.	Accept the terms of the end-user license agreement (EULA) by choosing Accept, and then click Next.
4.	Type your name in the User Name box, and the name of your company in the Organization box. After that, choose Next.
5.	Install all functionality by selecting Complete, and then click Next.
6.	Specify a location for the SDK folder or select the default folder. Click Next to proceed.
7.	Choose Install to install the SDK.
8.	Wait for the installation to finish.
9.	Once the installation is complete, click Finish to exit the installer.

## VB.NET/C# Applications: Use Visual Studio 2005

You may use Microsoft Visual Studio 2005 to create Visual Basic, Visual C++, Visual C#, and Visual J# applications, all of which use the same IDE.

### Step 2-2. Develop a "Hello" program

After installing the IDE tools, write a program in the computer language that you have chosen. Leave your contact information in the registry in the following format:

```
[HKEY_LOCAL_MACHINE/SOFTWARE/MOXA/PARTNERS/<your company's name>]
LastName <your last name>
Email <your email>
Nation <your nation>
```

We will collect and save your contact information in our backup database. After you compile the program, do not to execute the program directly on your workstation. Use an ftp client to upload and execute the program on our demo computer.

## C/C++ Applications: Use eMbedded Visual C++ (eVC) 4.0

1. Open Microsoft eMbedded Visual C++ 4.0.
2. From the File menu, choose New.
3. Choose the Projects tab and then select the type of application that you want to build.
4. Enter a name for the project, and then click OK.
5. Choose the type of application that you want to create and then click Finish.
6. On the Build toolbar, choose the SDK (UC7420CE1.0 in this example), the type of run-time image (Release or Debug), and the target device (UC7420CE1.0 Device in the example).
7. Develop your application.
8. From the Build menu, choose Rebuild All to compile the application.
9. Ignore the resulting error dialog. The error message is from ActiveSync when developing a mobile application.

### Example of an application written in C

```
#include "stdafx.h"

#define PARTNER_PATH L"SOFTWARE\\MOXA\\PARTNERS"
#define PARTNER_NAME L"LastName"
#define PARTNER_EMAIL L"Email"
#define PARTNER_NATION L"Nation"

int CreatePartner(LPWSTR company, LPWSTR name, LPWSTR email, LPWSTR nation)
{
    LONG IRet;
    HKEY hKey = NULL;
    DWORD dwDisposition;
    WCHAR szRegPath[MAX_PATH];

    wsprintf( szRegPath, L"%s\\%s", PARTNER_PATH, company );
    IRet = RegCreateKeyEx(HKEY_LOCAL_MACHINE, szRegPath, 0, NULL, 0, 0, NULL, &hKey,
    &dwDisposition);
    if(ERROR_SUCCESS != IRet)
        return 1;

    IRet = RegSetValueEx(hKey, PARTNER_NAME, 0, REG_SZ, (LPBYTE)name,
    (wcslen(name) + 1) * sizeof(WCHAR));
    if(ERROR_SUCCESS != IRet)
    {
        RegCloseKey(hKey);
        return 2;
    }

    IRet = RegSetValueEx(hKey, PARTNER_EMAIL, 0, REG_SZ, (LPBYTE)email,
    (wcslen(email) + 1) * sizeof(WCHAR));
```

```

if(ERROR_SUCCESS != IRet)
{
RegCloseKey(hKey);
return 3;
}

IRet = RegSetValueEx(hKey, PARTNER_NATION, 0, REG_SZ, (LPBYTE)nation,
(wcslen(nation)+1)*sizeof(WCHAR));
if(ERROR_SUCCESS != IRet)
{
RegCloseKey(hKey);
return 4;
}

return 0;
}

int WINAPI WinMain( HINSTANCE hInstance, HINSTANCE hPrevInstance,
LPTSTR lpCmdLine, int nCmdShow)
{
CreatePartner( L"MyCompany", L"MyLastName", L"MyEmail@some.com", L"MyNation" );
return 0;
}

```

## VB.NET/C# Applications: Use Visual Studio 2005

1. Open Microsoft Visual Studio .Net 2005.
2. From the File menu, choose **New Project**.
3. Choose the Project Type and then select the "Smart Device Application" type of application.
4. Enter a name for the project, and then click **OK**.
5. Choose the "Windows CE" target platform.
6. Select the project type and click **OK**.
7. Develop your .Net application.
8. On the Device toolbar, choose the **Windows CE.Net Device**.
9. From the Build menu, choose Build Project or Rebuild Project.

### Example of an application written in Visual Studio 2005 C# project

```

using System;
using System.Collections.Generic;
using System.Text;
using Microsoft.Win32;

namespace WriteReg
{
class Program
{
static void Main(string[] args)
{
const string szCompany = "MOXA_Systems";
const string szNation = "Taiwan";
const string szName = "B.R. Liao";
const string szEMail = "freddy@moxa.com.tw";

RegistryKey regHKLM = Registry.LocalMachine;
RegistryKey regPartners = regHKLM.OpenSubKey("SOFTWARE").
OpenSubKey("MOXA").OpenSubKey("PARTNERS", true);
RegistryKey regCompany = regPartners.CreateSubKey(szCompany);
regCompany.SetValue("LastName", szName);
regCompany.SetValue("Email", szEMail);
regCompany.SetValue("Nation", szNation);
regCompany.Close();
regPartners.Close();
}
}

```

```
regHKLM.Close();
regHKLM.Flush();
}
}
}
```

### Step 2-3. Upload and Execute Your Application

1. Upload your application (e.g, myProg.exe) to the UC-7420-CE computer.

```
\> ftp 59.124.42.149
Login: admin
Password: admin
> put myProg.exe
```

2. Use a telnet client to establish a connection to the UC-7420-CE computer. Then, type the login name and password as requested to logon.

```
\> telnet 59.124.42.149
Login: admin
Password: admin
```

3. Execute your program.

```
> \myProg
```

NOTE: After you log on to the demo computer, please be careful and do not reboot or crash it intentionally. We will periodically collect all of the contact information from participants worldwide and save it in our database.