

HT 開発メディアキット

ファーストステップガイド

このガイドは、CASIO 製ハンディターミナルのアプリケーション開発者向けガイドブックです。端末と開発環境との接続手順、およびアプリケーションを開発する上で必要となる CASIO 製 SDK に関するインストール方法やデバッグ方法について記載します。

ご注意

- このソフトウェアおよびマニュアルの、一部または全部を無断で使用、複製することはできません。
- このソフトウェアおよびマニュアルは、本製品の使用許諾契約書のもとでのみ使用することができます。
- このソフトウェアおよびマニュアルを運用した結果の影響については、一切の責任を負いかねますのでご了承ください。
- このソフトウェアの仕様、およびマニュアルに記載されている事柄は、将来予告なしに変更することがあります。
- このマニュアルの著作権はカシオ計算機株式会社に帰属します。
- 本書中に含まれている画面表示は、実際の画面とは若干異なる場合があります。予めご了承ください。

© 2017 カシオ計算機株式会社

Microsoft, MS, Active Desktop, Outlook, Windows, Windows NT, および Windows ロゴは、米国 Microsoft Corporation の米国およびその他の国における登録商標または商標です。Microsoft 社の製品は、OEM 各社に、Microsoft Corporation の 100% 出資子会社である Microsoft Licensing, Inc.によりライセンス供与されています。

はじめに

このたびは、「HT 開発メディアキット」をお買い上げいただきありがとうございます。
本ガイドでは、下記の名称を略称で表記しています。

略称	正式名称
WEC7	Microsoft Windows Embedded Compact 7
WEH6.5	Microsoft Windows Embedded Handheld 6.5

目次

1.	概要	1
1.1	ライブラリの構成	1
1.2	マニュアルの構成	2
2.	アプリケーション開発要件	3
2.1	必要なプログラミング知識	3
2.2	必要なシステム	3
2.3	必要なソフトウェア	3
3.	共通開発環境のインストール	5
3.1	アプリケーション開発環境の構築手順	5
3.1.1	WEC7 の場合	5
3.1.2	WEH6.5 の場合	6
3.2	開発キットのインストール	7
3.2.1	ExportSDK のインストール	8
3.2.2	デバイスライブラリのインストール	13
3.2.3	デバイスライブラリの Path の設定	16
4.	端末と PC との接続	19
4.1	Windows Mobile デバイスセンターによる接続	19
4.1.1	Windows Mobile デバイスセンターのインストール	19
4.1.2	USB による接続	20
4.1.3	Bluetooth による接続	22
4.2	無線 LAN による接続	24
4.3	イーサネットによる接続	30
4.4	PC の共有ドライブへのアクセス方法	32
4.5	スクリーンショットの撮り方	34
4.6	Visual Studio からの TCP/IP 接続方法	35
5.	機種別開発環境のインストール	36
5.1	機種別デバイスライブラリのインストール	36
6.	デバイスエミュレータ	43
6.1	WEC7 デバイスエミュレータ	44
6.1.1	ExportSDK のインストール	44
6.1.2	デバイスライブラリのインストール	47
6.1.3	デバイスエミュレータのインストール	48
6.1.4	デバイスエミュレータの起動	51
6.1.5	デバイスエミュレータと ExportSDK の関連付け	52
6.1.6	デバイスエミュレータの画面解像度の設定	55
6.1.7	デバイスエミュレータのディスクサイズの拡張	58
6.2	WEH6.5 デバイスエミュレータ	64
6.2.1	デバイスエミュレータのインストール	64
6.2.2	デバイスエミュレータの起動	67
6.2.3	Windows Mobile デバイスセンターとの接続	67
6.3	IO シミュレータ	68
6.3.1	IO シミュレータのインストール	68
6.3.2	IO シミュレータの起動	69
6.3.3	IO シミュレータの操作	70
7.	Visual Studio によるアプリケーションの開発	75

7.1	システムライブラリを使用したプログラミング (Visual Basic)	76
7.2	システムライブラリを使用したプログラミング (Visual C#)	81
7.3	システムライブラリを使用したプログラミング (Visual C++)	86
8.	自動復旧	92
9.	端末の安定化	93
9.1	リセット運用	93
9.2	サービスパックとパッチファイル	93
9.3	メモリ	93
10.	テクニカルインフォメーション・オンラインサポートサービス	94
11.	注意事項	97
11.1	HWND_BROADCAST について	97
11.2	システムフォントについて	98

1. 概要

本書は、Windows OS がインストールされている端末のアプリケーション開発者向けの開発ガイドブックです。

端末と開発環境との接続手順、および端末のアプリケーションを開発する上で必要となる SDK に関するインストール方法やテスト方法について記載します。

1.1 ライブラリの構成

本商品で提供するライブラリの一覧を以下に示します。

ライブラリ名称	概要	C++	VB C#
システムライブラリ	システム動作の制御に使用します。	○	○
レーザスキャナライブラリ	レーザスキャナの制御に使用します。	○	○
Bluetooth ライブラリ	Bluetooth モジュールの制御に使用します。	○	○
カメラライブラリ	カメラモジュールの制御に使用します。	○	○
イメージャライブラリ	CMOS 型イメージャの制御に使用します。	○	○
NFC ライブラリ	IC カードとの通信に使用します。	○	○
NFCMifare ライブラリ	Mifare カードとの通信に使用します。	○	○
NFCFelica ライブラリ	Felica カードとの通信に使用します。	○	○
NFCHFTag ライブラリ	HFTag カードとの通信に使用します。	○	○
WANGPRS ライブラリ	GPRS モジュールの制御に使用します。	○	○
JPEG ライブラリ	JPEG 画像の操作に使用します。	○	—
FLINK ライブラリ	PC や他の端末とのファイル転送に使用します。	○	○
音声出力ライブラリ	数値文字列の音声出力に使用します。	○	—

※ 上の表では、下記の略語を使用しています。

- C++ : Visual C++
- VB : Visual Basic.NET
- C# : Visual C#
- : 対象
- : 対象外

ダイナミックリンクライブラリの名称は下記のとおりです。

ライブラリ名称	ダイナミックリンク ライブラリ	ダイナミックリンク クラスライブラリ (.NET 用)
システムライブラリ	SystemLib.dll	SystemLibNet.dll
レーザスキャナライブラリ	OBReadLib.dll	OBReadLibNet.dll
Bluetooth ライブラリ	BluetoothLib.dll	BluetoothLibNet.dll
カメラライブラリ	CameraLib.dll	CameraLibNet.dll
イメージャライブラリ	ImagerLib.dll	ImagerLibNet.dll
NFC ライブラリ	NFCLib.dll	NFCLibNet.dll
NFCMifare ライブラリ	NFCMifareLib.dll	NFCMifareLibNet.dll
NFCFelica ライブラリ	NFCFelicaLib.dll	NFCFelicaLibNet.dll
NFCHFTag ライブラリ	NFCHFTagLib.dll	NFCHFTagLibNet.dll
WANGPRS ライブラリ	WANGPRSLib.dll	WANGPRSLibNet.dll
JPEG ライブラリ	JpegCe.dll	なし
FLINK ライブラリ	FlinkLib.dll	MoFlinkLib.dll
音声出力ライブラリ	MoSpk.dll	なし

1.2 マニュアルの構成

本商品で提供するライブラリの一覧を以下に示します。

名称	概要	C++	VB C#
ファーストステップガイド	本書	—	—
デバイスライブラリマニュアル		—	—
システムライブラリマニュアル	システムライブラリの関数仕様です。	○	○
レーザスキャナライブラリマニュアル	レーザスキャナライブラリの関数仕様です。	○	○
Bluetooth ライブラリマニュアル	Bluetooth ライブラリの関数仕様です。	○	○
カメラライブラリマニュアル	カメラライブラリの関数仕様です。	○	○
イメージャライブラリマニュアル	イメージャライブラリの関数仕様です。	○	○
NFC ライブラリマニュアル	NFC ライブラリの関数仕様です。	○	○
NFCMifare ライブラリマニュアル	NFCMifare ライブラリの関数仕様です。	○	○
NFCFelica ライブラリマニュアル	NFCFelica ライブラリの関数仕様です。	○	○
NFCHFTag ライブラリマニュアル	NFCHFTag ライブラリの関数仕様です。	○	○
WANGPRS ライブラリマニュアル	WANGPRS ライブラリの関数仕様です。	○	○
JPEG ライブラリマニュアル	JPEG ライブラリの関数仕様です。	○	—
FLINK ライブラリマニュアル	FLINK ライブラリの関数仕様です。	○	○
音声出力ライブラリ	音声出力ライブラリの関数仕様です。	○	—

※ 上の表では、下記の略語を使用しています。

- C++ : Visual C++
- VB : Visual Basic.NET
- C# : Visual C#
- : 対象
- : 対象外

2. アプリケーション開発要件

2.1 必要なプログラミング知識

アプリケーションを開発するためには、以下のいずれかの開発言語によるプログラミング技術が必要です。

- Visual C++
- Visual Basic .NET
- Visual C#

C 言語で作成されたアプリケーションは、ネイティブ環境で動作するため、高速動作が可能です。対して、Visual Basic、Visual C#で開発されたアプリケーションは、マネージコードのため、多少動作が遅くなります。

さらに、以下の技術知識があることが望ましいです。

- OS (WEC、WEH 等)
- Windows Mobile デバイスセンター
- Active Server Pages、Web プログラミング
- SQL Server、SQL Server Compact Edition
- その他ネットワーク関連

2.2 必要なシステム

OS

- Microsoft Windows 7 Professional 32bit with ServicePack1
- Microsoft Windows 7 Professional 64bit with ServicePack1

コンピュータ本体

- CPU 2.4GHz 以上
- RAM 768MB 以上
- モニタ解像度 1024x768 以上

WEC7 のデバイスエミュレータを使用する場合は、下記の値が必要となります。

- RAM 1024MB 以上

2.3 必要なソフトウェア

アプリケーションを開発するためには、Microsoft 社の開発プラットフォーム、通信ソフトおよび SDK が必要です。

必須ソフトウェア

- Microsoft Visual Studio 2008 + SP1
- Microsoft Windows Mobile デバイスセンター 6.1
<https://support.microsoft.com/ja-jp/help/931937/description-of-windows-mobile-device-center>

WEC7 のアプリケーションを開発するためには、Visual Studio 2008 + SP1 に対して、以下のパッチが必須です。

(下記の URL は変更されている場合があります)

- Visual Studio 2008 update for Windows Embedded Compact 7
<http://www.microsoft.com/en-us/download/details.aspx?id=11935>
- Windows Embedded Compact 7 ATL Update for Visual Studio 2008 SP1
<http://www.microsoft.com/en-us/download/details.aspx?id=27729>

WEH6.5 のアプリケーションを開発するためには、Visual Studio 2008 + SP1 に対して、以下の SDK が必須です。

- Windows Mobile 6 Professional SDK Refresh
<http://www.microsoft.com/en-us/download/details.aspx?id=6135>

任意ソフトウェア

WEC7 のデバイスエミュレータを使用する場合は、以下のソフトウェアが必要です。

- Microsoft Windows Virtual PC
<http://www.microsoft.com/ja-jp/download/details.aspx?id=3702>

3. 共通開発環境のインストール

3.1 アプリケーション開発環境の構築手順

本商品を使用して、アプリケーションを作成する手順を以下に示します。
構築手順は使用する端末の OS により異なります。

3.1.1 WEC7 の場合

(1) 開発プラットフォームのインストール

PC に Microsoft 製の開発プラットフォームをインストールします。
本商品は Visual Studio 2008 に対応しています。
詳細については、「**2.3 必要なソフトウェア**」を参照してください。

(2) 開発キットの PC へのインストール

PC に ExportSDK、デバイスライブラリをインストールします。
インストール方法については、「**3.2 開発キットのインストール**」を参照してください。

(3) PC との接続 (Windows Mobile デバイスセンターのインストール)

PC に Microsoft 製の Windows Mobile デバイスセンターをインストールします。
接続方法については、「**4 端末と PC との接続**」を参照してください。

(4) 開発キットの端末へのインストール

端末にライブラリをインストールします。
インストール方法については、「**5 機種別開発環境のインストール**」を参照してください。

(5) デバイスエミュレータの準備

PC でデバイスエミュレータが利用できるように準備します。
操作方法については、「**6 デバイスエミュレータ**」を参照してください。
デバイスエミュレータが必要でない場合は、(6)へ進んでください。

(6) アプリケーションの開発

(1)の開発プラットフォームを使用して、アプリケーションを開発します。
開発したアプリケーションは、(3)のツールを用いて、端末あるいはデバイスエミュレータに転送し、動作を確認します。
操作方法については、「**7 Visual Studio によるアプリケーションの開発**」を参照してください。

3.1.2 WEH6.5 の場合

(1) 開発プラットフォームのインストール

PC に Microsoft 製の開発プラットフォームをインストールします。
本商品は Visual Studio 2008 に対応しています。
詳細については、「**2.3 必要なソフトウェア**」を参照してください。

(2) 開発キットの PC へのインストール

PC に Microsoft 製の Windows Mobile 6 Professional SDK Refresh、CASIO 製のデバイスライブラリをインストールします。
インストール方法については、「**3.2 開発キットのインストール**」を参照してください。

(3) PC との接続 (Windows Mobile デバイスセンターのインストール)

PC に Microsoft 製の Windows Mobile デバイスセンターをインストールします。
接続方法については、「**4 端末と PC との接続**」を参照してください。

(4) 開発キットの端末へのインストール

端末にライブラリをインストールします。
インストール方法については、「**5 機種別開発環境のインストール**」を参照してください。

(5) デバイスエミュレータの準備

PC でデバイスエミュレータが利用できるように準備します。
操作方法については、「**6 デバイスエミュレータ**」を参照してください。
デバイスエミュレータが必要でない場合は、(6)へ進んでください。

(6) アプリケーションの開発

(1)の開発プラットフォームを使用して、アプリケーションを開発します。
開発したアプリケーションは、(3)のツールを用いて、端末あるいはデバイスエミュレータに転送し、動作を確認します。
操作方法については、「**7 Visual Studio によるアプリケーションの開発**」を参照してください。

3.2 開発キットのインストール

「開発キット」の CD-ROM を PC のドライブに挿入すると、自動的に以下のメニュー画面が起動します。画面内のバージョン番号は CD によって異なります。



3.2.1 ExportSDK のインストール

WEC7 の端末をご利用の場合は、ExportSDK が必要となります。
ExportSDK をインストールする前に、必ず「2.3 必要なソフトウェア」に記載したパッチを当ててください。
お使いの端末により、インストールする ExportSDK が異なります。

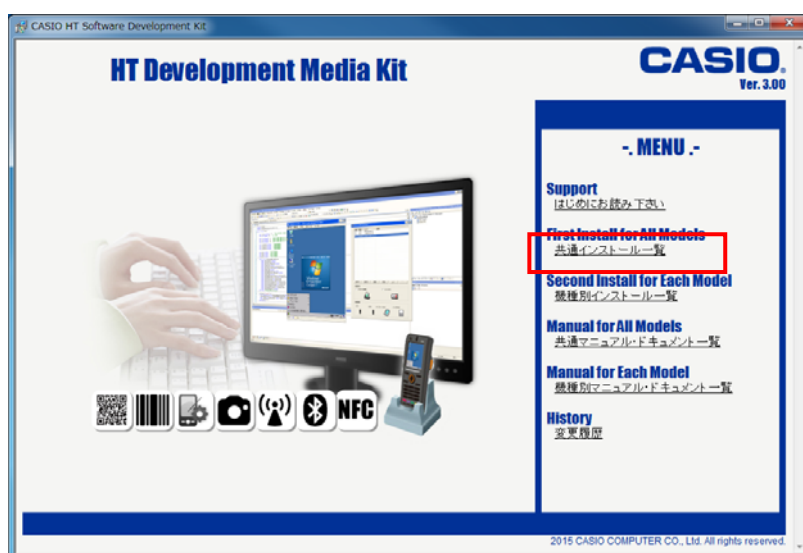
ExportSDK	端末
CASIO ARMv7 ExportSDK	IT-G500
CASIO ARMv5 ExportSDK	DT-X100 / DT-X200

注意

WEH6.5 の端末をご利用の場合は、CASIO 製の ExportSDK をインストールする必要はありません。
「2.3 必要なソフトウェア」を参照して、Microsoft 製の Windows Mobile 6 Professional SDK Refresh をインストールしてください。

ここでは、「CASIO ARMv7 ExportSDK」をインストールする手順について記載します。

1. 「共通インストーラー一覧」をクリックします。



2. 「CASIO ARMv7 ExportSDK」をクリックします。



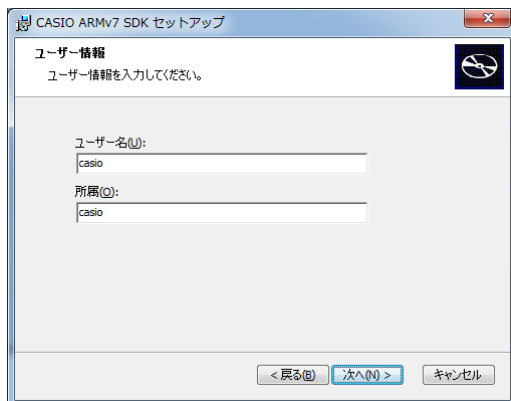
3. 「次へ」を押下します。



4. 使用許諾書に同意し、「次へ」を押下します。



5. 「ユーザ名」、「所属」を入力し、「次へ」を押下します。



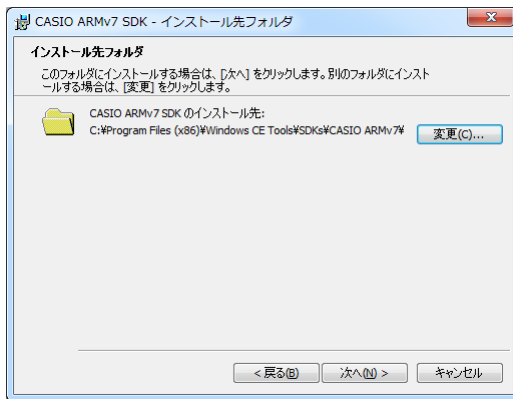
6. 「セットアップの種類を選択」で「完全」を選択します。



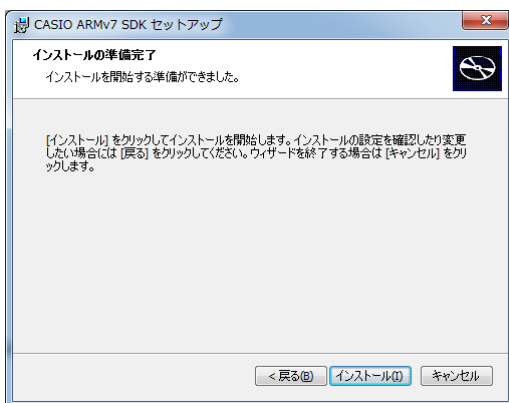
7. インストール先を変更する場合は、必要に応じて変更し、「次へ」を押下します。デフォルトでは以下にインストールします。

32bit: C:\Program Files\Windows CE Tools\SDKs\CASIO ARMv7

64bit: C:\Program Files (x86)\Windows CE Tools\SDKs\CASIO ARMv7



8. 「インストール」を押下します。



9. インストールが完了したら、「終了」を押下します。



3.2.2 デバイスライブラリのインストール

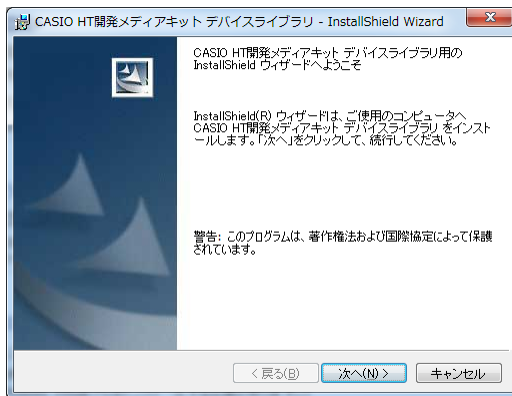
1. 「共通インストーラー一覧」をクリックします。



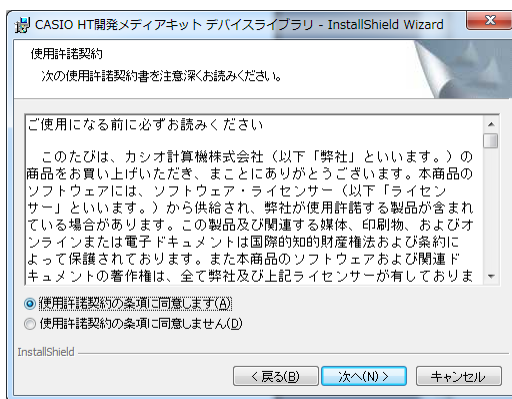
2. 「CASIO HT 開発メディアキット デバイスライブラリ」をクリックします。



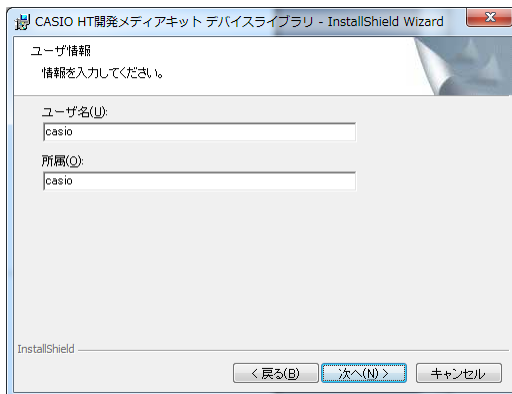
3. 「次へ」を押下します。



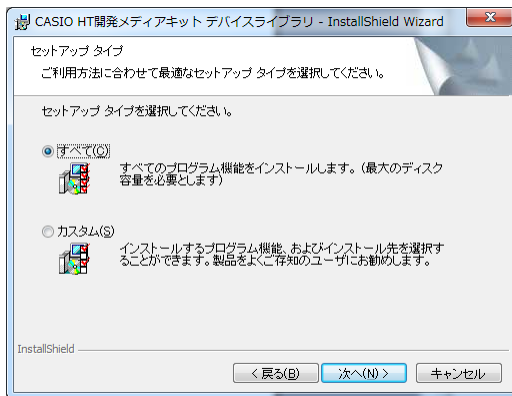
4. 使用許諾書に同意し、「次へ」を押下します。



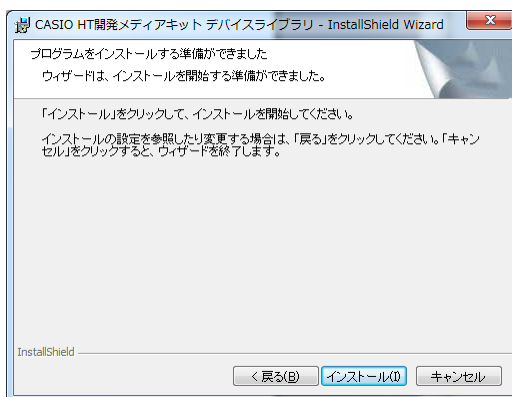
5. 「ユーザ名」、「所属」を入力し、「次へ」を押下します。



6. 「セットアップの種類を選択」で「すべて」を選択します。



7. 「インストール」を押下します。

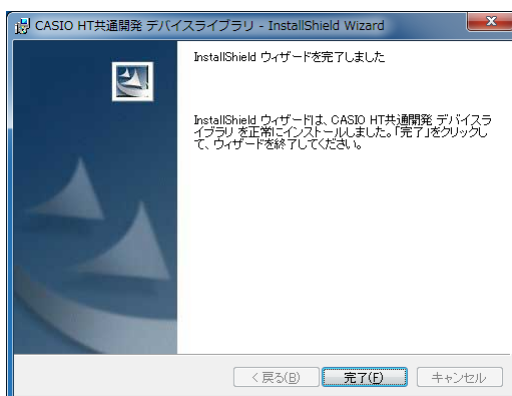


「CASIO HT 開発キット デバイスライブラリ」は下記にインストールされます。

32bit: C:¥Program Files¥CASIO¥MBSYS

64bit: C:¥Program Files (x86)¥CASIO¥MBSYS

8. インストールが完了したら、「終了」を押下します。

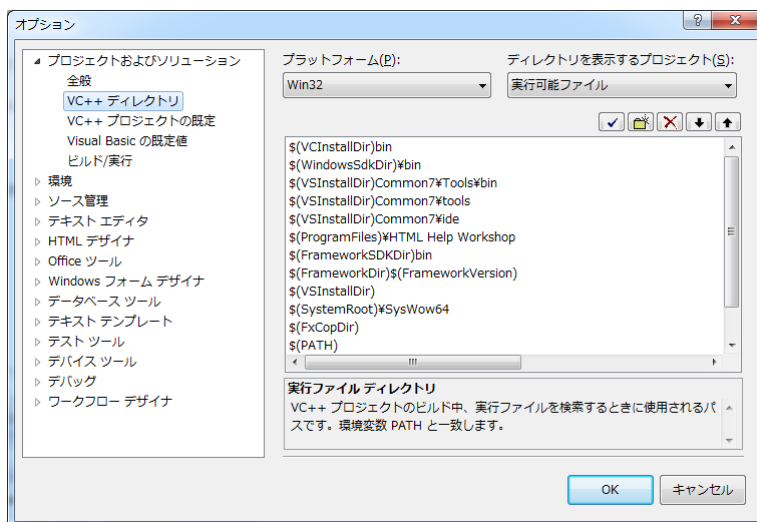


3.2.3 デバイスライブラリの Path の設定

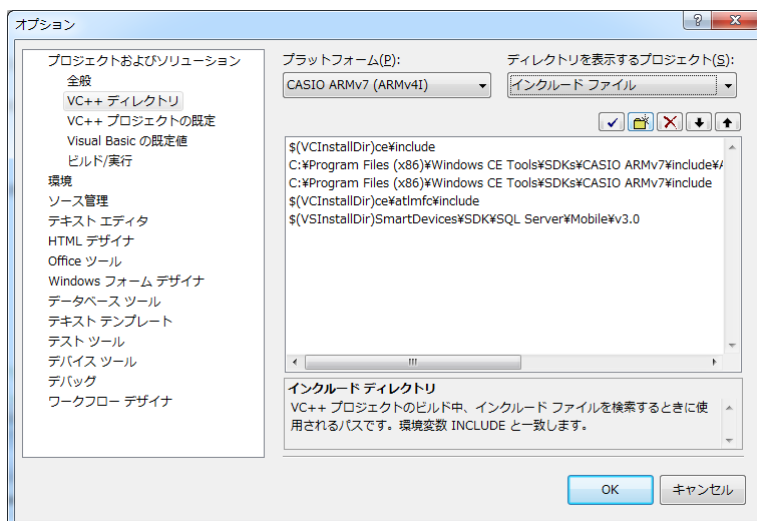
ExportSDK に対して、デバイスライブラリの Path を設定します。

ここでは、"CASIO ARMv7 ExportSDK"に Path を設定する手順について記載します。

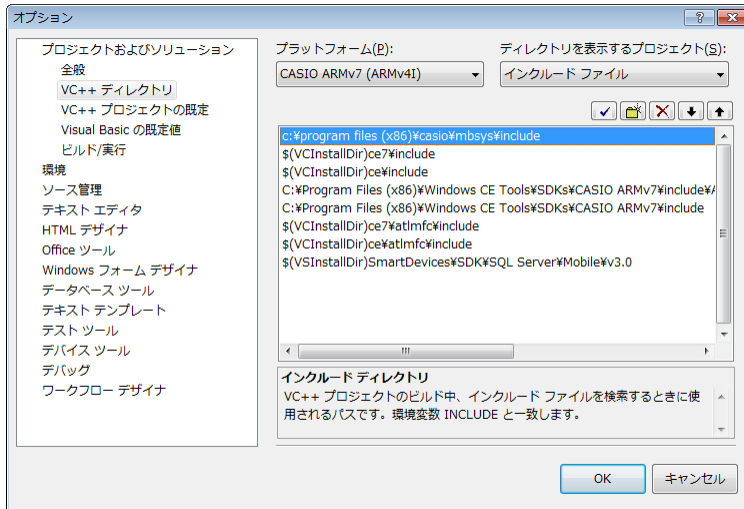
1. VisualStudio2008 のメニューから「ツール」>「オプション」を起動し、「プロジェクトおよびソリューション」の「VC++ディレクトリ」を選択します。



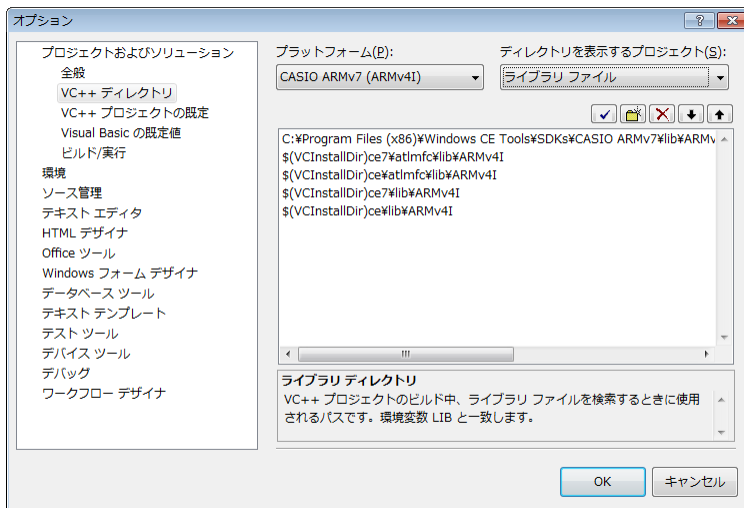
2. プラットフォームに「CASIO ARMv7 (ARMv4I)」、ディレクトリを表示するプロジェクトに「インクルード ファイル」をそれぞれ選択します。(CASIO ARMv5 ExportSDK をご利用の場合は「CASIO ARMv5 (ARMv4I)」を、Windows Mobile 6 Professional SDK をご利用の場合は「Windows Mobile 6 Professional SDK (ARMv4I)」を選択します)



3. 下図のように、「3.2.2 デバイスライブラリのインストール」でインストールしたフォルダを登録します。
- 32bit: C:\Program Files\CASIO\MBSYS\include
 64bit: C:\Program Files (x86)\CASIO\MBSYS\include



4. ディレクトリを表示するプロジェクトに「ライブラリ ファイル」をそれぞれ選択します。



5. 下図のように、「3.2.2 デバイスライブラリのインストール」でインストールしたフォルダを登録します。

32bit: C:\Program Files\CASIO\MBSYS\lib\ArmV7

64bit: C:\Program Files (x86)\CASIO\MBSYS\lib\ArmV7

CASIO ARMv5 ExportSDK をご利用の場合は下記のフォルダを登録します。

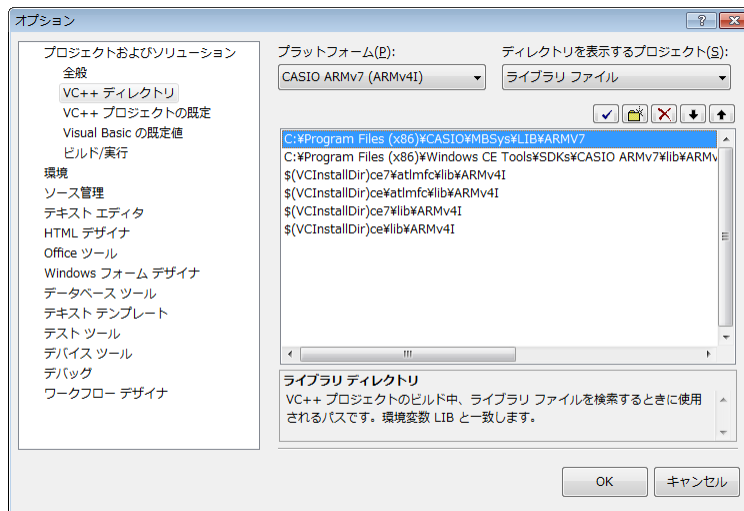
32bit: C:\Program Files\CASIO\MBSYS\lib\ArmV5

64bit: C:\Program Files (x86)\CASIO\MBSYS\lib\ArmV5

Windows Mobile 6 Professional SDK をご利用の場合は下記のフォルダを登録します。

32bit: C:\Program Files\CASIO\MBSYS\lib\ArmV4i

64bit: C:\Program Files (x86)\CASIO\MBSYS\lib\ArmV4i



4. 端末と PC との接続

アプリケーションの開発を行う際に必要な、端末と PC の接続方法を説明します。

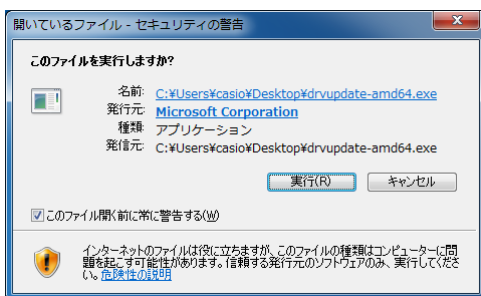
1. Windows Mobile デバイスセンターによる接続
2. 無線 LAN による接続
3. 有線 LAN による接続

4.1 Windows Mobile デバイスセンターによる接続

本節では、Windows Mobile デバイスセンター6.1 以降を使用して、端末と PC を接続する方法を説明します。

4.1.1 Windows Mobile デバイスセンターのインストール

1. Microsoft 社よりセットアップファイルをダウンロードし、起動します。



2. 内容を確認し、「同意する」を押下します。



3. インストールの進捗ウィンドウを表示します。ウィンドウが閉じるとインストール完了となります。

次に、Windows Mobile デバイスセンターで端末と PC を接続する通信方法について記載します。通信方法には、下記の 2 種類があります。

- USB による接続
- Bluetooth による接続

注意！「Bluetooth による接続」に関しましては、ご使用になる Bluetooth 機器との相性により、必ずしも安定した通信が可能とは言い切れません。

4.1.2 USBによる接続

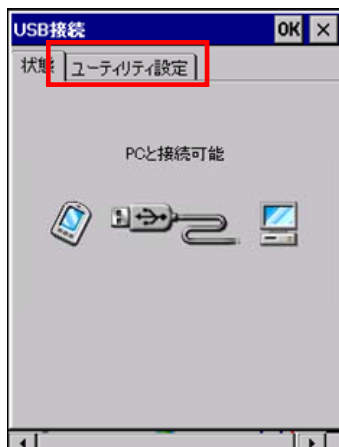
1. 使用中のアプリケーションをすべて終了します。
2. USB 接続を起動します。

WE7C の場合は、スタートメニューから「**設定(S)**」>「**コントロールパネル(C)**」>「**USB 接続**」を選択します。

WEH6.5 の場合は、スタートメニューから「**設定**」>「**システム**」>「**USB 接続**」を選択します。



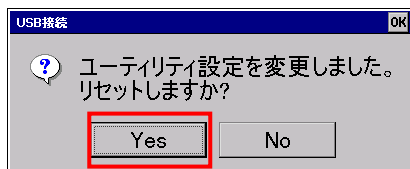
3. 「**ユーティリティ設定**」タブをクリックします。



4. 「Windows Mobile Device Center」にチェックを入れ、「OK」をクリックします。




5. リセット確認ダイアログが表示されるので、「はい(Y)」をクリックしてください。



6. 端末が再起動した後、端末を I/O ボックスに装着し、PC 上の Windows Mobile デバイスセンター画面の指示に従ってください。

4.1.3 Bluetooth による接続

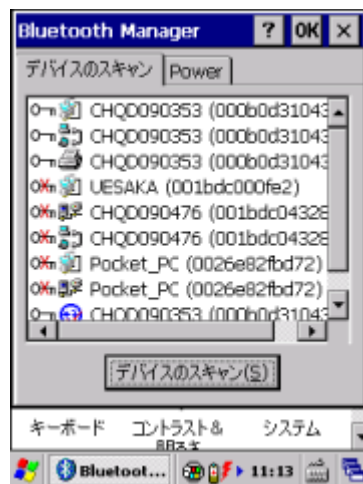
WEC7 の場合

1. PC に Bluetooth デバイスをインストールしていない場合には、Bluetooth をインストールします。
2. 端末のコントロールパネルから「Bluetooth デバイスのプロパティ」を選択します。
3. 「Power」タブ画面で Power 設定にチェックを入れて Bluetooth の電源を ON にします。
4. 「デバイスのスキャン」タブ画面から、「デバイスのスキャン」ボタンを押下して近くの Bluetooth デバイスを表示させます。
5. スキャンされた Bluetooth 一覧の中から  というマーク (ActiveSync) がついているものをダブルタップし「信頼あり」を選択して PIN コードを入力します。「信頼あり」にチェックがついたことを確認します。又「作動中」にもチェックをつけてください。
6. 端末のコントロールパネルから「PC との接続」を選択し接続方法を Bluetooth に変更して下さい。
7. スタートメニューから「プログラム(P)」>「通信」>「WMDC」を起動して PC と接続して下さい。

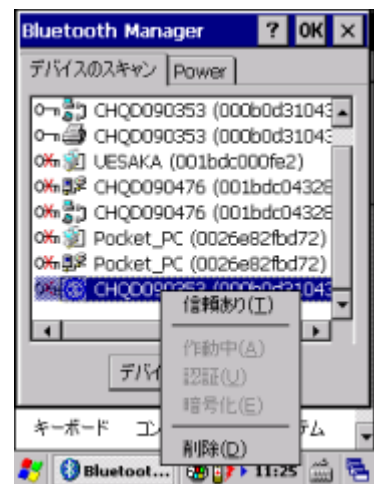
3.



4.



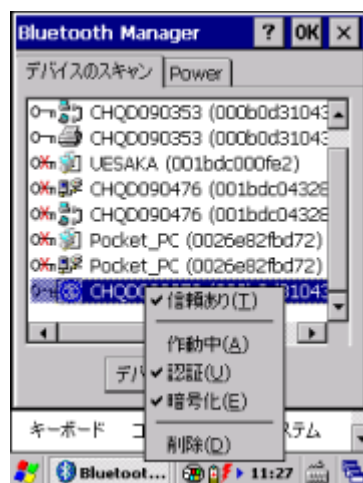
5.



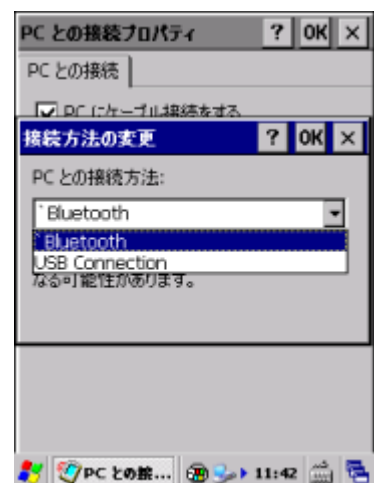
5.



5.



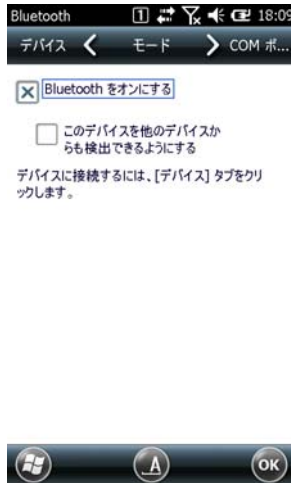
6.



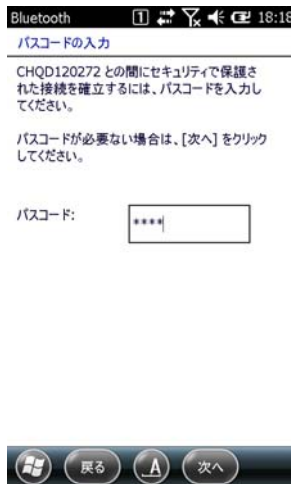
WEH6.5 の場合

1. PC に Bluetooth デバイスをインストールしていない場合には、Bluetooth をインストールします。
2. 端末の設定から「Bluetooth」を選択します。
3. 「モード」タブ画面の「Bluetooth をオンにする」チェックボックスにチェックを入れます。
4. 「デバイス」タブ画面から、「新しいデバイスの追加...」より接続する PC を選択して接続設定をします。
5. 「ActiveSync」にチェックが入っていることを確認します。
6. 「ActiveSync」プログラムよりメニューの「Bluetooth から接続」を選択して接続します。

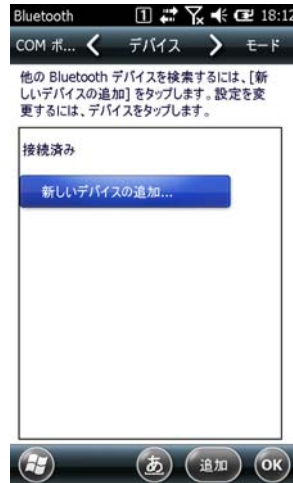
3.



4.



4.



5.



4.



6.



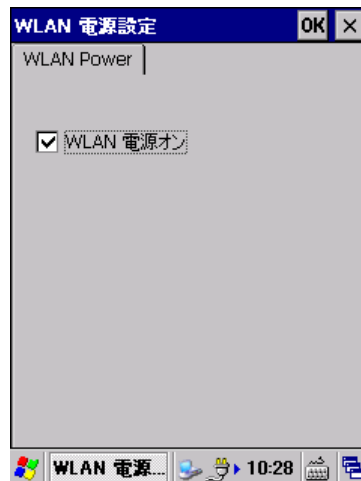
4.2 無線 LAN による接続

端末を無線 LAN 接続するには、以下の手順に従って操作してください。

※ 無線 LAN を利用する前に、サイトサーベイを行ってから環境を構築してください。

[無線 LAN 接続設定方法]

WEC7 の場合は「設定(S)」>「コントロールパネル(C)」>「WLAN 電源設定」を、WEH6.5 の場合は「設定」>「接続」>「WLAN 電源設定」を選択し、「WLAN 電源オン」にチェックを入れます。



[IP 設定]

WEC7 の場合は「設定(S)」>「コントロールパネル(C)」>「無線 LAN 設定」を、WEH6.5 の場合は「設定」>「接続」>「無線 LAN 設定」を選択します。



項目	値
DHCP 有効	DHCP の有効/無効を指定します。
IP 指定	
IP	IP アドレスを指定します。 [例: 192.168.1.100]
マスク	サブネットマスクを指定します。
ゲートウェイ	デフォルトゲートウェイを指定します。
DNS1	プライマリ DNS サーバアドレスを指定します。
DNS2	セカンダリ DNS サーバアドレスを指定します。
WINS1	プライマリ WINS サーバアドレスを指定します。
WINS2	セカンダリ WINS サーバアドレスを指定します。

省略時の動作は以下のように処理します。

項目	空白時の処理	範囲外の値の処理
DHCP 有効	有効と見なします。	有効と見なします。
IP	設定しません。	そのまま書き込みます。
マスク	設定しません。	そのまま書き込みます。
ゲートウェイ	設定しません。	そのまま書き込みます。
DNS1	設定しません。	そのまま書き込みます。
DNS2	設定しません。	そのまま書き込みます。
WINS1	設定しません。	そのまま書き込みます。
WINS2	設定しません。	そのまま書き込みます。

[基本設定]

「基本設定」に移動します。



項目		値
セキュリティ	無効	なし
	WEP	オープン
	WPA WPA2	PSK (キー設定も設定必要)
		EAP-PEAP
キー		EAP-TLS
		16 進数表記 (HexDecimal) で入力する場合、32 バイト分 (64 文字) 固定 半角英数字で入力する場合、最大 63 文字 入力済の文字数を表示します。 ini ファイルからキーを読み込んだ場合は*****で表示します。*****表示を 編集した場合は、新たにキーが設定されたとみなします。一度も編集されな かった場合は、ini ファイルから読み込んだキーがそのまま有効です。

EAP-PEAP 設定時は以下の入力も必要です。

項目	設定内容	デフォルト値
ユーザ名	半角英数字で入力（最大 100 文字）	なし
パスワード	半角英数字で入力（最大 100 文字） ini ファイルからパスワードを読み込んだ場合は*****で表示します。*****表示を編集した場合は、新たにキーが設定されたとみなします。一度も編集されなかった場合は、ini ファイルから読み込んだキーがそのまま有効です。	なし
ドメイン	半角英数字で入力（最大 100 文字）	なし
サーバーの有効化	サーバー証明書チェックのためのルート証明書の必要性を設定します。 チェックあり = 証明書必要 チェックなし = 証明書不要	証明書必要

EAP-TLS 設定時は以下の入力も必要です。

項目	設定内容	デフォルト値
ユーザ名	半角英数字で入力（最大 100 文字）	なし
証明書	半角英数字で入力（最大 100 文字） インストール済のクライアント証明書を選択します。 「検索」ボタンをタップすると、インストール済のクライアント証明書一覧が表示されるので、選択します。	なし
ドメイン	半角英数字で入力（最大 100 文字）	なし
サーバーの有効化	サーバー証明書チェックのためのルート証明書の必要性を設定します。 チェックあり = 証明書必要 チェックなし = 証明書不要	証明書必要

[WLAN 設定]

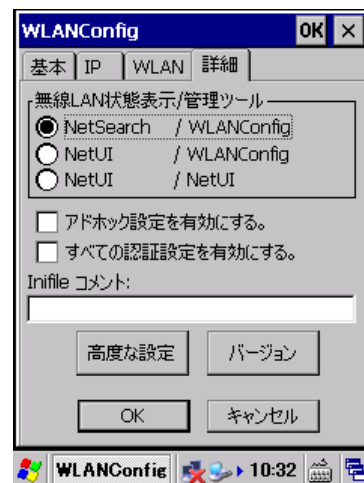
「WLAN 設定」に移動します。



項目	設定内容	デフォルト値
WLAN 電源	オン	○
	オフ	
省電力	有効	○
	無効	
規格	無線 LAN 規格	abgn
ローミング開始閾値	ローミングしない	
	無線 LAN 通信が実質的に行えない-100dBm をローミング開始閾値として設定する事により、ローミング動作しなくなります。	
	標準	○
	推奨値である-78dBm をローミング開始閾値として設定します。	
ローミングしやすい	標準より高い電界強度である-72dBm でローミング動作が開始します。	
	指定	
	ローミング開始閾値を手動で設定します。	
優先周波数帯	なし	○
	2.4GHz	
	5GHz	

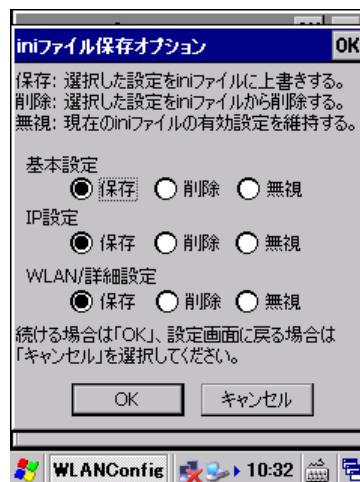
[詳細設定]

「詳細設定」に移動します。



項目	設定内容	デフォルト値
無線 LAN 状態表示/ LAN 管理ツール選択	[NetSearch/WLANConfig] 状態表示 : NetSearch 管理ツール : WLANConfig	○
	[NetUI/WLANConfig] 状態表示 : NetUI 管理ツール : WLANConfig	
	[NetUI/NetUI] 状態表示 : NetUI 管理ツール : NetUI	
アドホック設定を有効	チェックあり = 設定可能	
	チェックなし = 設定不能	○
すべてのセキュリティ設定 を有効	チェックあり = 設定可能	
	チェックなし = 設定不能	○
コメント	ini ファイルに記入するコメントを指定します。 半角 100 文字(全角 50 文字)まで設定可能です。	なし

「OK」ボタンを押すと右図のメッセージが表示されるので「OK」を押してください。
リセット後、設定が反映されます。

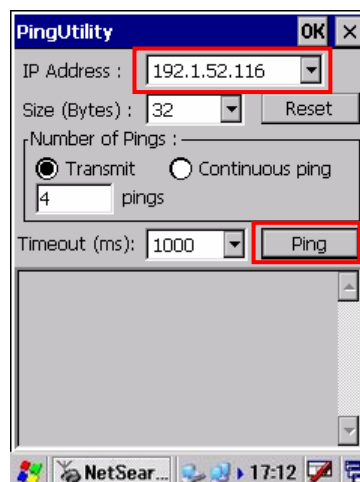


[接続の確認]

ネットサーチを起動して「Ping(4)」を選択してください。



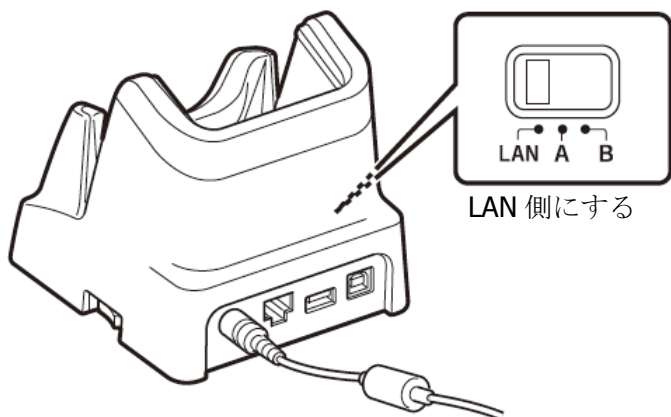
IP アドレスを入力してから「Ping」を押し、接続を確認してください。



4.3 イーサネットによる接続

LAN I/O ボックスを利用することにより、高速な LAN 接続が使用可能です。

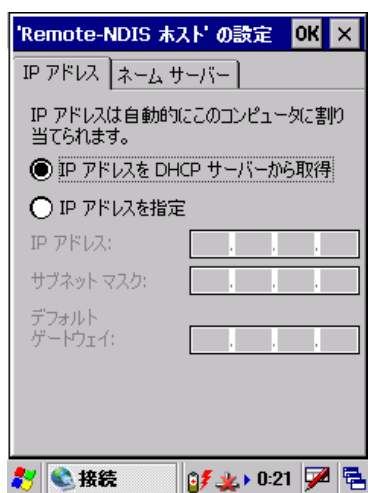
1. LAN I/O ボックスに AC アダプタを接続します。
2. ネットワークケーブルを LAN I/O ボックスに接続し、反対側をネットワークハブに接続します。
3. LAN I/O ボックス側面のスイッチを、「LAN」側に設定します。



4. 端末を、LAN I/O ボックスに装着します。
5. 端末で、[スタート]→[設定]→[コントロールパネル]→[ネットワークとダイヤルアップ接続] を選択します。
6. 下記画面が表示されるので、AX887721 をダブルクリックします。



7. 下記 TCP/IP 設定画面が表示されるので、必要な項目の設定を行い[OK]をクリックします。



正常に接続が成功すると、タスクトレイ上のネットワークアイコンから、“×”が消えます。

4.4 PC の共有ドライブへのアクセス方法

端末がネットワークで接続されている場合、端末のエクスプローラからPCで共有されているドライブへのアクセスが可能になります。

以下の手順に従って、操作してください。

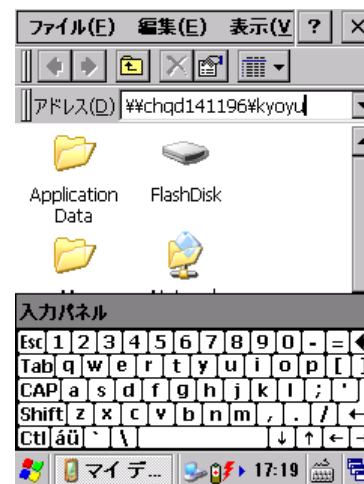
WEC7 の場合

1. 端末でネットワークへ接続するための設定を行います。
2. マイデバイスを起動します。
3. アドレスバーを表示。
4. ネットワーク上での PC 名を入力します。(例: ¥¥<hostname> IP アドレスでの指定はできません)
5. ネットワークのログオンダイアログボックスが表示されたら、ユーザ ID、パスワード、およびネットワークドメインを入力します。
6. PC で共有されているドライブが表示されるので、PC と端末間でファイルのコピーが可能となります。

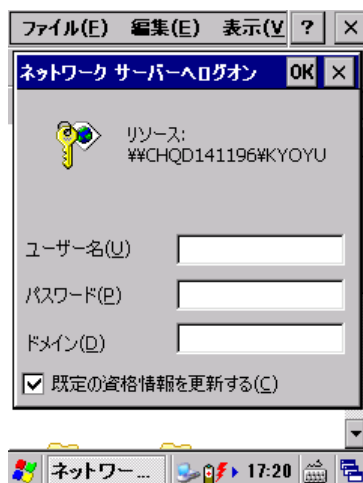
3. アドレスバーを表示



4. ネットワーク PC 名を入力



5. ユーザ名、パスワードを入力



WEH6.5 の場合

1. 端末でネットワークへ接続するための設定を行います。
2. エクスプローラーを起動します。
3. 「メニュー」→「移動」→「パスを開く」→「新しいパス...」を表示。
4. ネットワーク上での PC 名を入力(例: *hostname* IP アドレスでの指定はできません)します。
5. ネットワークのログオンダイアログボックスが表示されたら、ユーザ名、パスワード、およびドメインを入力します。
6. PC で共有されているドライブが表示されるので、PC と端末間でファイルのコピーが可能となります。

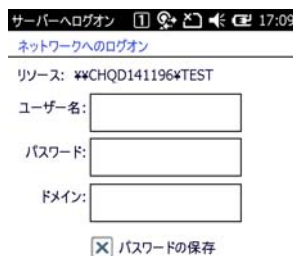
3. 新しいパス...を表示



4. ネットワーク PC 名を入力



5. ユーザ名、パスワードを入力



4.5 スクリーンショットの撮り方

端末と PC を Windows Mobile デバイスセンターで接続していると、Visual Studio を使用して端末のスクリーンショットを撮ることができます。

以下の手順に従って、操作してください。

1. スタートメニュー > Microsoft Visual Studio 2008 > Visual Studio Remote Tools > リモートズームインを起動します。



2. 接続している端末に合わせたデバイスを選択して OK を押下します。

デバイス	端末
CASIO ARMv7 Device	IT-G500 (WEC7 モデル)
CASIO ARMv5 Device	DT-X100 / DT-X200
CASIO x86 Device	WEC7 のデバイスエミュレータ
Windows Mobile 6 Professional Device	IT-G500 (WEH6.5 モデル) / WEH6.5 のデバイスエミュレータ



4.6 Visual Studio からの TCP/IP 接続方法

1. Visual Studio 2008 がインストールされている PC から、Windows Mobile デバイスセンター等を用いて、下記のファイルを端末へ転送します。
 - Clientshutdown.exe
 - ConmanClient2.exe
 - CMAccept.exe
 - DeviceDMA.dll
 - eDbgTL.dll
 - TcpConnectionA.dll

<コピー元(PC)>

¥Program Files¥Common Files¥Microsoft Shared¥CoreCon¥1.0¥Target¥wce400¥arm4i

<コピー先(端末)>

¥Windows

2. 端末側で ConmanClient2.exe を実行します。
3. Visual Studio 2008 上でデバイスの IP をセットします。
 1. “ツール”→“オプション”→“デバイスツール”→“デバイス”
 2. “Window CE デバイス”を選択し、“プロパティ”をクリックします。
 3. トランスポートの“構成”を押します。
 4. 特定の IP アドレスを使用し端末に割り振った IP アドレスを記述し OK をクリックします。
4. 端末側で CMAccept.exe を実行します。
5. Visual Studio 2008 上で“ツール”→“デバイスの接続”を行います。
6. “Windows CE デバイス”を選択し接続を行います。

5. 機種別開発環境のインストール

5.1 機種別デバイスライブラリのインストール

機種別デバイスライブラリをインストールすると、PC の下記フォルダに、機種別の CAB ファイルがインストールされます。

インストールの際には、機種別のライセンスが必要となります。

CAB ファイルの内容

デフォルトでは下記にインストールされます。

32bit: C:\Program Files\CASIO\MBSYS\CAB\XXX

64bit: C:\Program Files (x86)\CASIO\MBSYS\CAB\XXX

(XXX:機種名)

CAB ファイルには、下の表に示すランタイムライブラリが格納されています。

ライブラリ名	CAB ファイル名
システムライブラリ	SystemLib[XXXXYY]J.[ZZZ].CAB
レーザスキャナライブラリ	OBReadLib[XXXXYY]J.[ZZZ].CAB
Bluetooth ライブラリ	BluetoothLib[XXXXYY]J.[ZZZ].CAB
カメラライブラリ	CameraLib[XXXXYY]J.[ZZZ].CAB
イメージャライブラリ	ImagerLib[XXXXYY]J.[ZZZ].CAB
JPEG ライブラリ	JpegCe[XXXXYY]J.[ZZZ].CAB
FLINK ライブラリ	Flink[XXXXYY]J.[ZZZ].CAB
音声出力ライブラリ	MoSpk[XXXXYY]J.[ZZZ].CAB
NFC ライブラリ	NFCLib[XXXXYY]J.[ZZZ].CAB

(XXX:機種名、YYY:OS 名、ZZZ:バージョン情報)

例) IT-G500 の WEC7 のシステムライブラリ Ver.3.00 は、「SystemLibITG500WECJ.300.cab」となります。

※ デバイスエミュレータをご使用になる場合、すでにインストールされているので、CAB ファイルを転送する必要はありません。

※ CAB ファイルの名称や種類等は機種により異なります。

※ 機種で利用可能なライブラリについては、デバイスライブラリ基本マニュアルを参照してください。

※ 最新のデバイスライブラリは常に ServicePack で配布しています。最新の ServicePack については、製品サポートサイトを参照してください。

インストール手順

ここでは、WEC7 の IT-G500 のデバイスライブラリを端末にインストールする手順を説明します。

(1) PC へのインストール

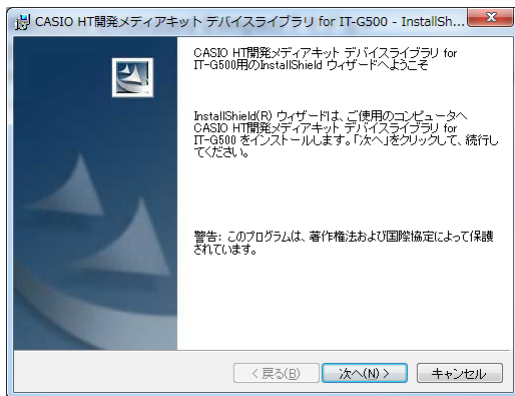
1. 「機種別インストール一覧」を押下します。



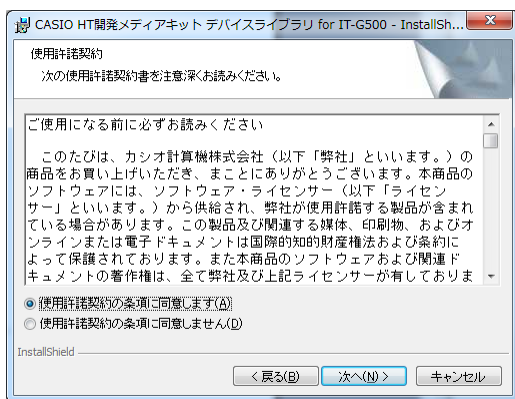
2. ご利用の端末に合わせたデバイスライブラリを選択します。ここでは、WEC7 の「CASIO HT 開発メディアキット デバイスライブラリ for IT-G500」を押下します。



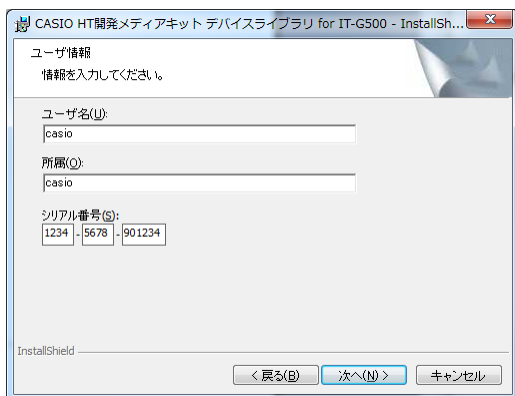
3. 「次へ」を押下します。



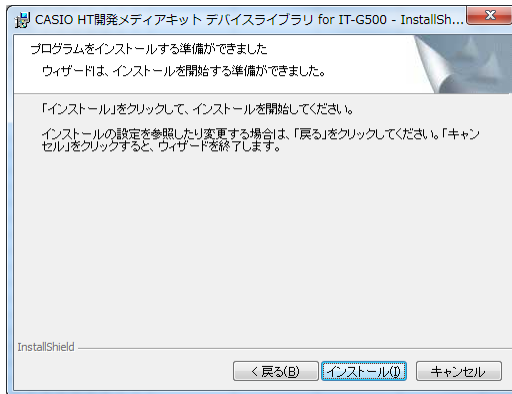
4. 使用許諾書に同意し、「次へ」を押下します。



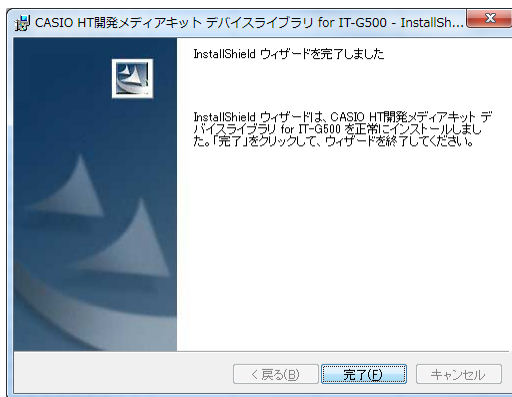
5. 「ユーザ名」、「所属」、「シリアル番号」を入力し、「次へ」を押下します。



6. 「インストール」を押下します。

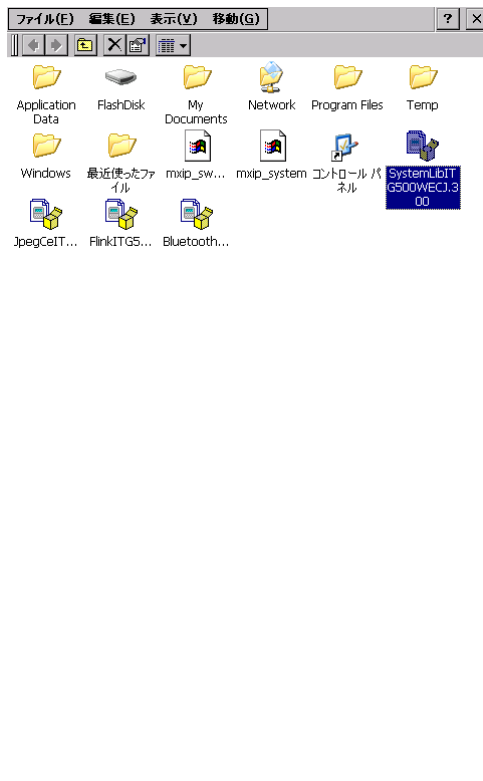


7. インストールが完了したら、「完了」を押下します。



(2) CAB ファイルの転送

Windows Mobile デバイスセンターを利用して、IT-G500 上の任意の場所に各 CAB ファイルを転送します。

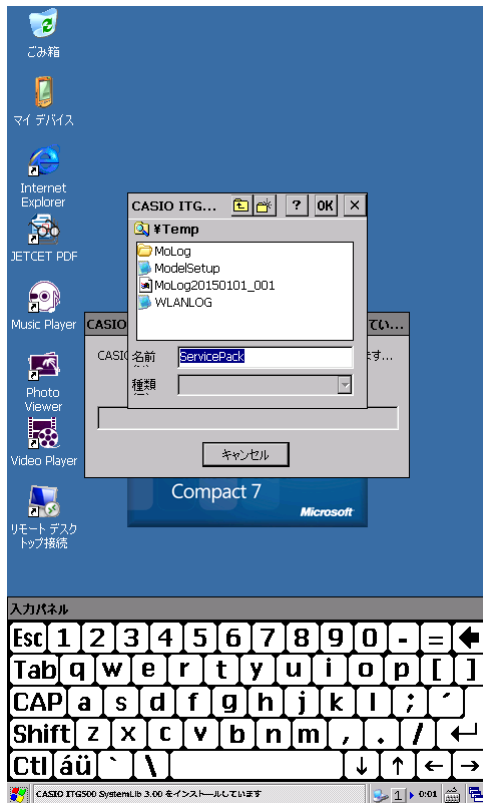


※ 本ガイドでは、CAB ファイルを IT-G500 のルートフォルダに配置しています。

※ Windows Mobile デバイスセンターの利用方法は「**4 端末と PC との接続**」を参照してください。

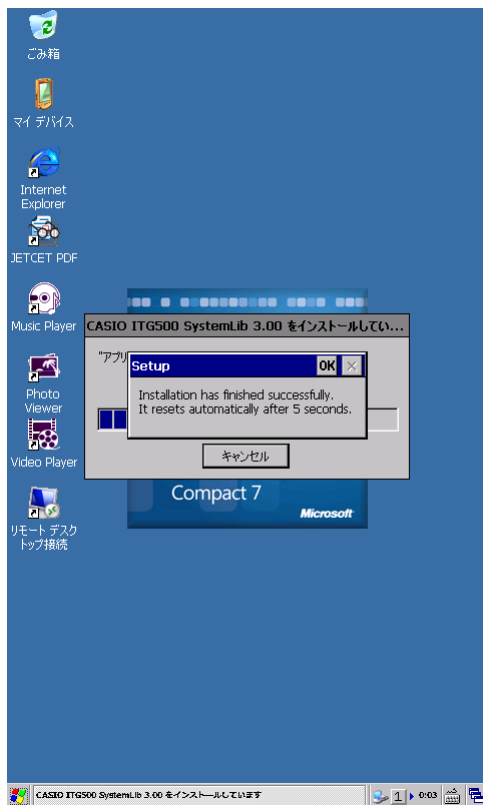
(3) CAB ファイルのインストール

転送した CAB ファイルを実行します。



IT-G500 の ENT キーあるいはソフトキーボード上の ENT キーを押下します。

(4) インストール完了



※ 本ガイドでは、システムライブラリのインストール手順を示しています。他のライブラリも同様の手順でインストールしてください。

※ IT-G500 のフルリセットを実行した場合は、再度インストールを行ってください。

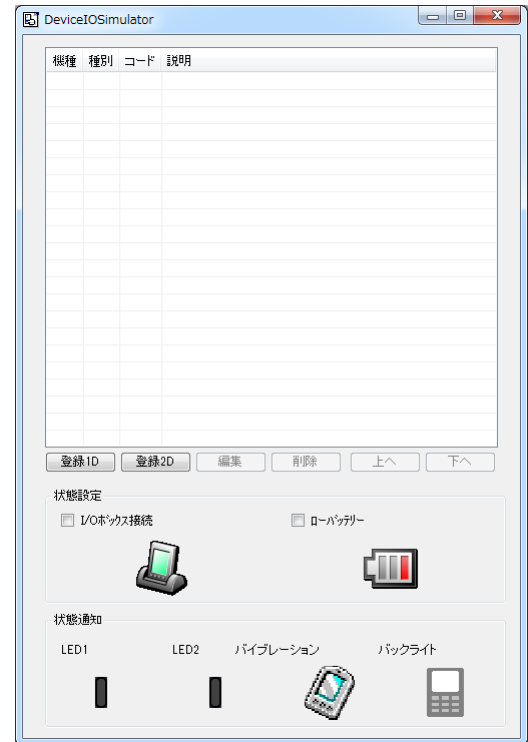
以上で、システムライブラリをご使用になるための準備が整いました。

6. デバイスエミュレータ

デバイスエミュレータは、「実機がなくても、アプリケーションの基本動作確認やソースレベルデバッグができる環境」をアプリケーション開発者に提供します。



デバイスエミュレータ



I/O シミュレータ

6.1 WEC7 デバイスエミュレータ

6.1.1 ExportSDK のインストール

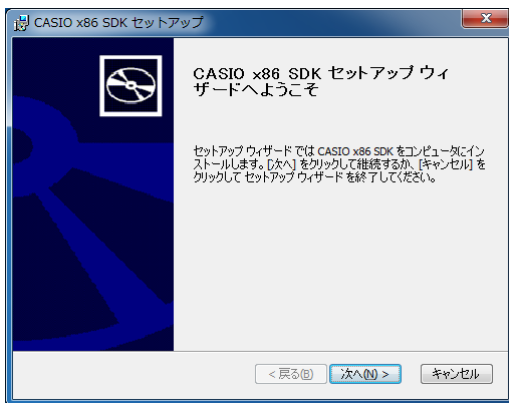
1. 「共通インストーラ一覧」をクリックします。



2. 「CASIO x86 ExportSDK」をクリックします。



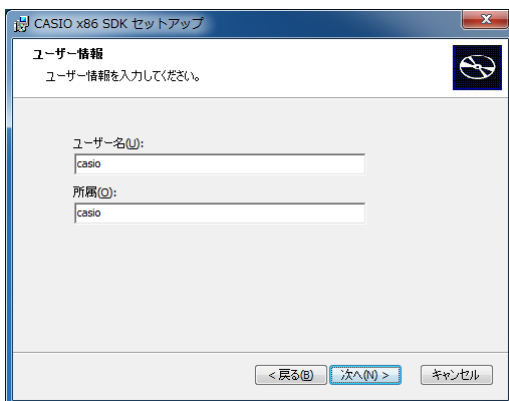
3. 「次へ」を押下します。



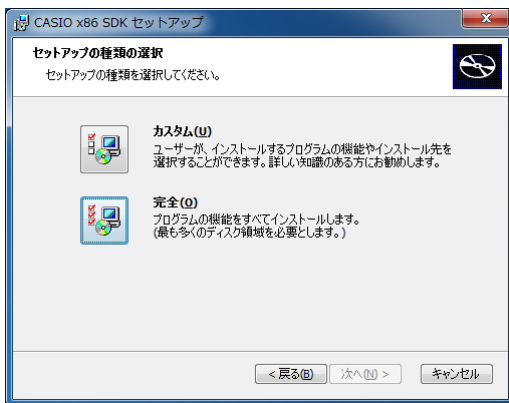
4. 使用許諾書に同意し、「次へ」を押下します。



5. 「ユーザ名」、「所属」を入力し、「次へ」を押下します。



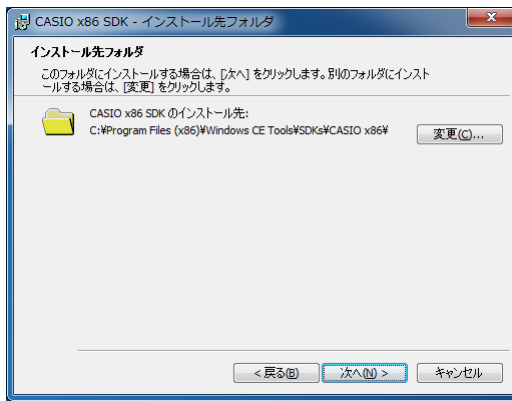
6. 「セットアップの種類を選択」で「完全」を選択します。



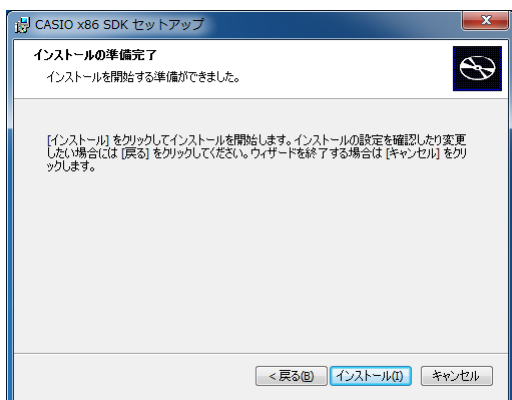
7. インストール先を変更する場合は、必要に応じて変更し、「次へ」を押下します。デフォルトでは以下にインストールします。

32bit: C:\Program Files\Windows CE Tools\SDKs\CASIO x86

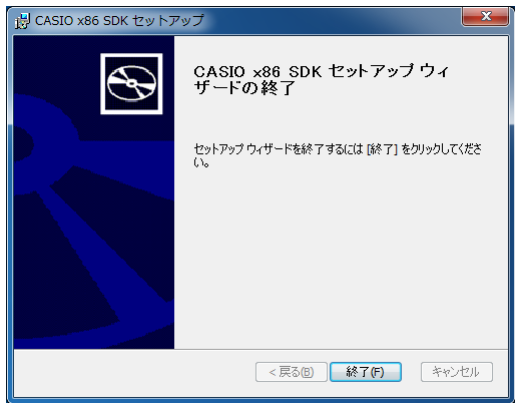
64bit: C:\Program Files (x86)\Windows CE Tools\SDKs\CASIO x86



8. 「インストール」を押下します。



9. インストールが完了したら、「終了」を押下します。



6.1.2 デバイスライブラリのインストール

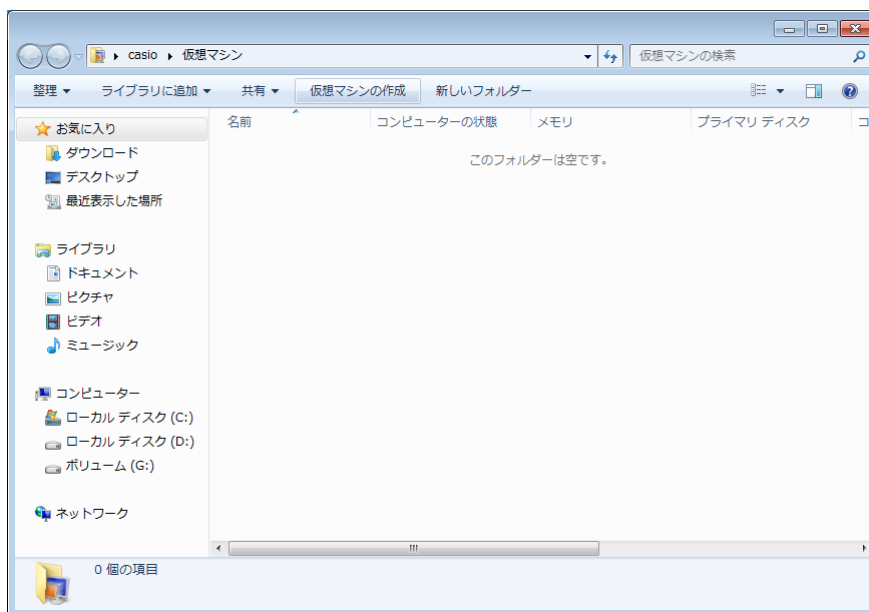
デバイスライブラリをインストールしていない場合は、「**3.2.2 デバイスライブラリのインストール**」にしたがって、インストールしてください。

6.1.3 デバイスエミュレータのインストール

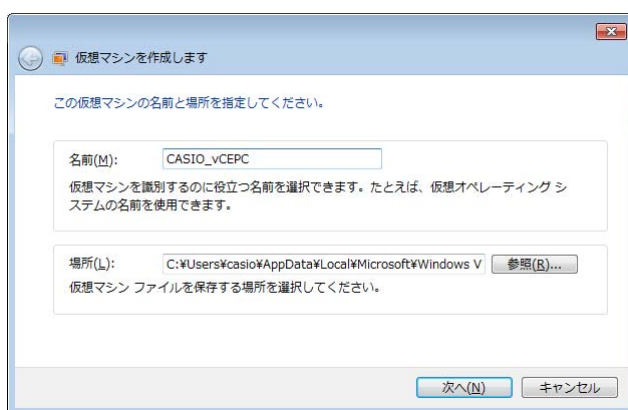
「Windows Embedded Compact 7 Emulator Image for CASIO」をインストールします。
インストールイメージは CD の下記にあります。

SETUP¥COMMON¥DEVICEEMULATOR¥CE

1. 上記フォルダ内の ZIP ファイルを解凍後に生成されるすべてのファイル(エミュレータイメージ)を PC の任意のディレクトリにコピーしてください。
本ガイドでは、C:¥IT-G500¥にコピーしたものとします。
2. スタートメニューから「Windows Virtual PC」を起動し、「仮想マシンの作成」を選択します。



3. デバイスエミュレータの「名前」と「場所」に任意の文字列を入力し、「次へ」を押下します。
ここでは名前を「CASIO_vCEPC」とします。

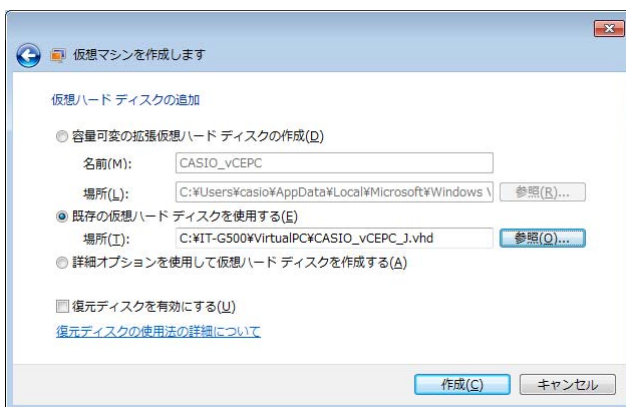


4. 「RAM」に 256MB 以上の値を入力し、「コンピュータのネットワーク接続を使用する」にチェックを入れ、「次へ」を押下します。ここで指定した値により、デバイスエミュレータの「データ記憶用メモリ」および「プログラム実行用メモリ」のサイズが変化します。

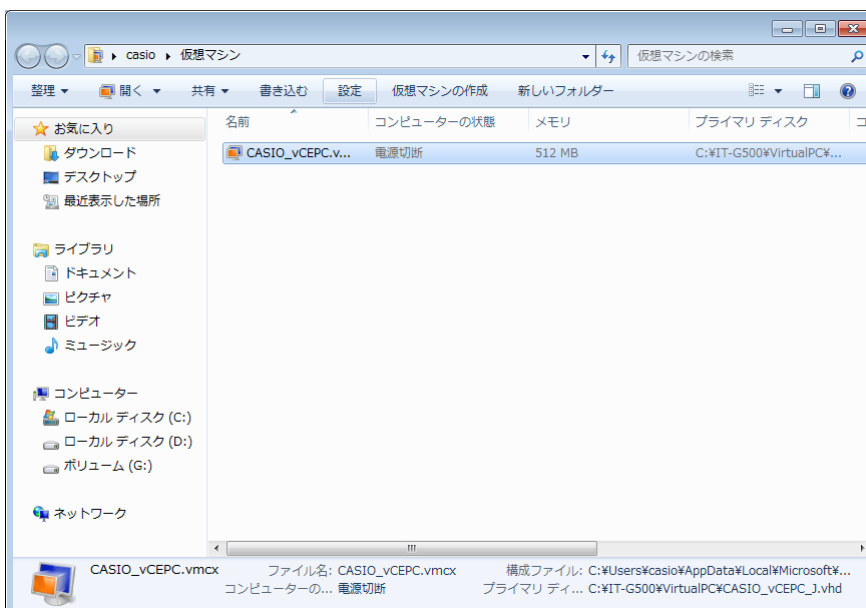


5. 「既存の仮想ハードディスクを使用する」の「場所」に、1.でインストールしたファイル(エミュレータイメージ)を解像度に応じて選択し、「作成」を押下します。

WQVGA : CASIO_vCEPC_WQVGA_J.vhd
その他の解像度 : CASIO_vCEPC_J.vhd



6. 1.から 5.で作成したデバイスエミュレータのファイル「CASIO_vCEPC.vmcx」を選択した状態で「設定」を押下します。



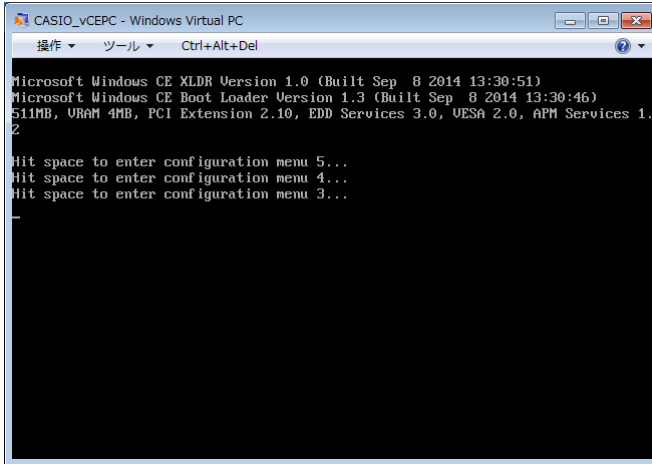
7. 「ハードディスク 2」に 1. でインストールしたファイル(エミュレータイメージ)「CASIO_FLASHDISK_J.vhd」を選択し、「OK」を押下します。



6.1.4 デバイスエミュレータの起動

「CASIO VirtualPC」を起動します。

1. 「6.1.3 デバイスエミュレータのインストール」で作成したデバイスエミュレータのファイル「CASIO_vCEPC.vmcx」をエクスプローラ上でダブルクリックすると、下記の画面が表示されます。




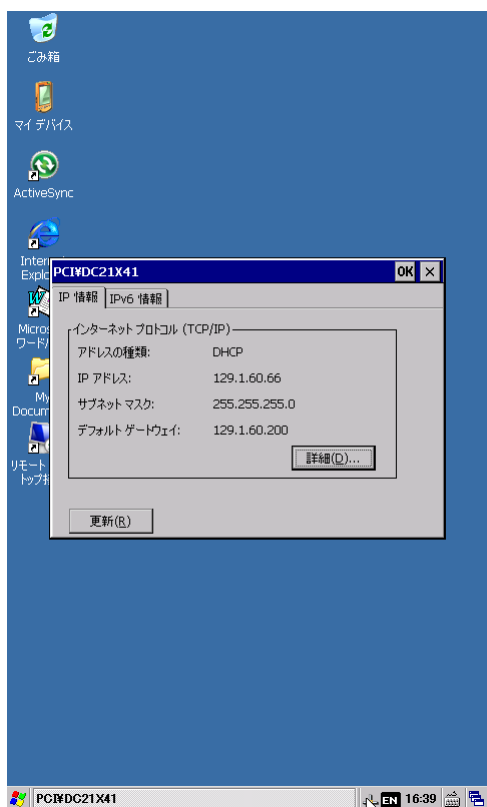
2. しばらくすると、デバイスエミュレータが起動します。(環境によっては起動に数分かかる場合があります。)



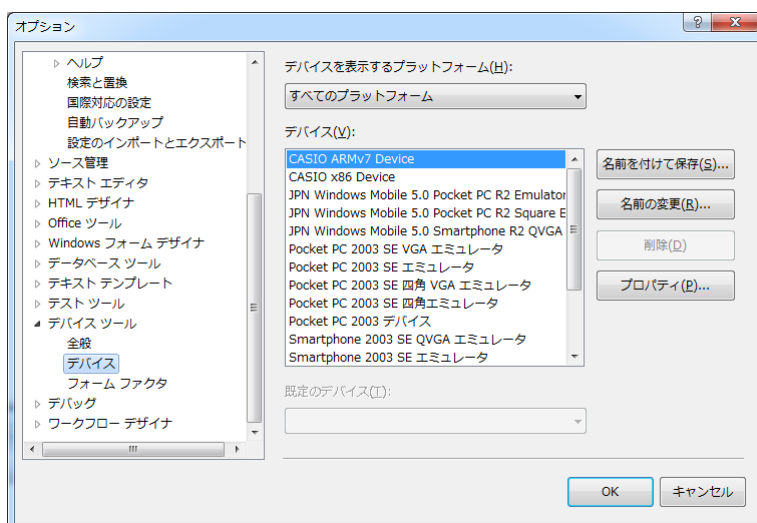
6.1.5 デバイスエミュレータと ExportSDK の関連付け

「6.1.1 ExportSDK のインストール」でインストールした「CASIO x86 SDK」と、デバイスエミュレータの関連付けを行います。

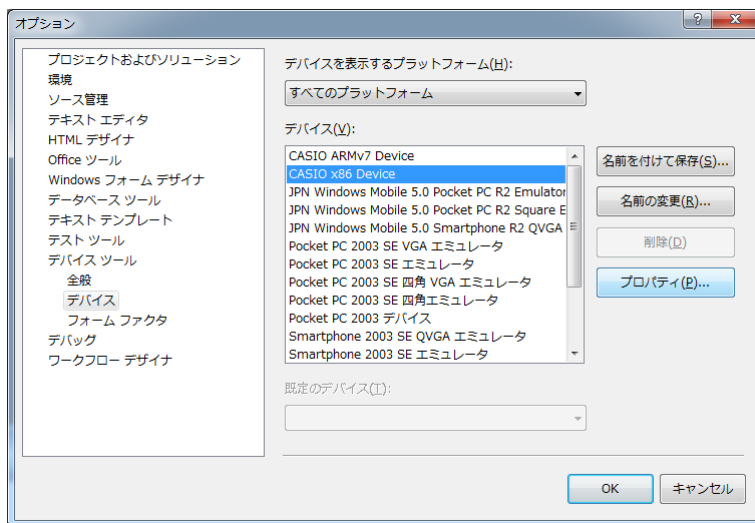
1. デバイスエミュレータ (CASIO_vCEPC) を起動し、タスクバーにある  をダブルクリックし、「IP アドレス」を控えます。



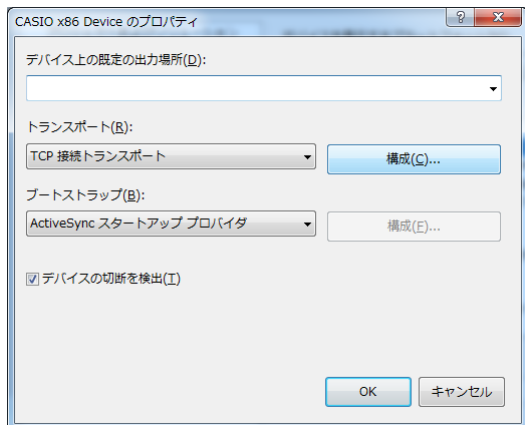
2. VisualStudio2008 のメニューから「ツール」>「オプション」を起動し、「デバイスツール」の「デバイス」を選択します。



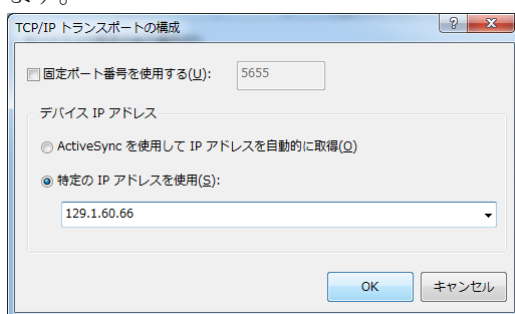
3. デバイスの一覧から「CASIO x86 Device」を選択し、「プロパティ」を押下します。



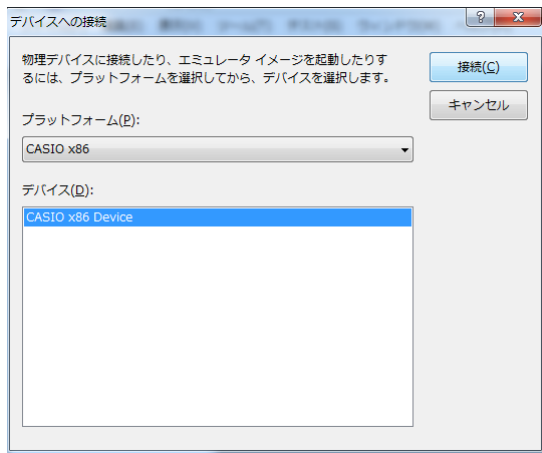
4. 「トランスポート」の「構成」を押下します。



5. 1.で控えたデバイスエミュレータの IP アドレスを「特定の IP アドレスを使用」に入力し、「OK」を押下します。



6. VisualStudio2008 のメニューから「ツール」>「デバイスへの接続」を起動し、プラットフォームに「CASIO x86」、デバイスに「CASIO x86 Device」を選択した状態で「接続」を押下します。



7. 下図の「接続に成功しました。」が表示されればデバイスエミュレータの設定は成功です。下図以外のメッセージが表示されている場合は、デバイスエミュレータの IP アドレスと入力した IP アドレスが一致しているか確認してください。



接続に失敗した場合は下記の項目を確認してください。

- デバイスエミュレータ上で IP アドレスの設定
- PC とデバイスエミュレータ間の通信確認 (ping)

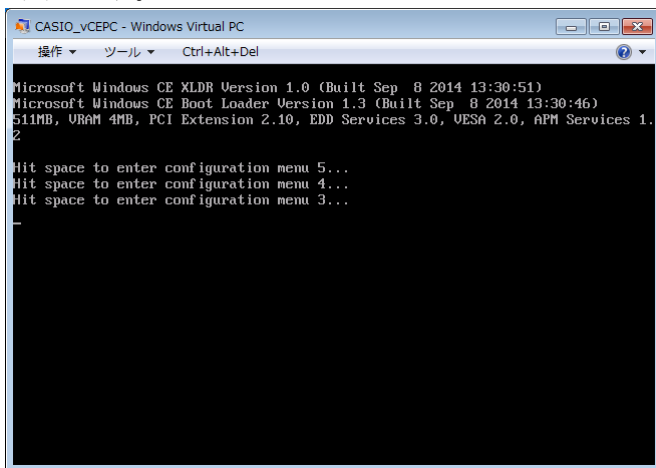
6.1.6 デバイスエミュレータの画面解像度の設定

デバイスエミュレータの画面解像度を設定します。

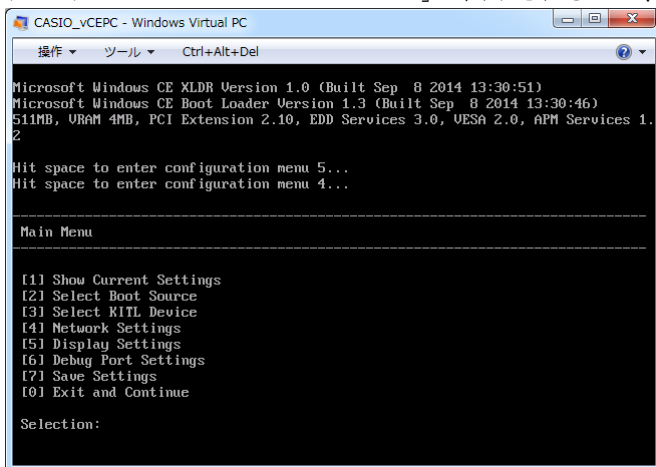
デフォルトでは、**480x800** で起動します。

ここでは **480x640** の画面解像度で起動する手順について記載します。

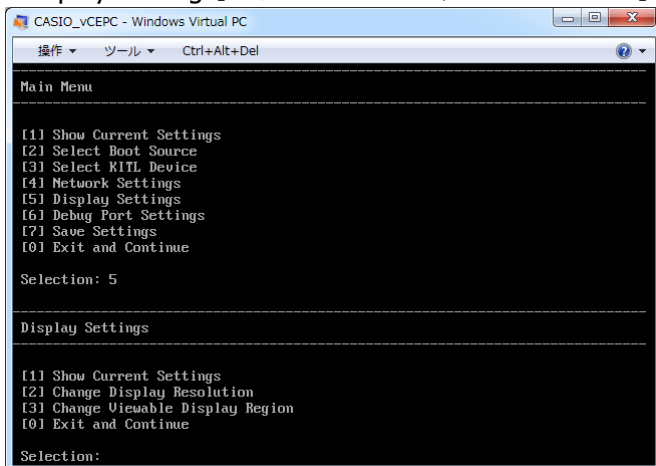
1. デバイスエミュレータを起動し、下記の画面が表示されている間に、キーボードの「スペース」キーを押下します。



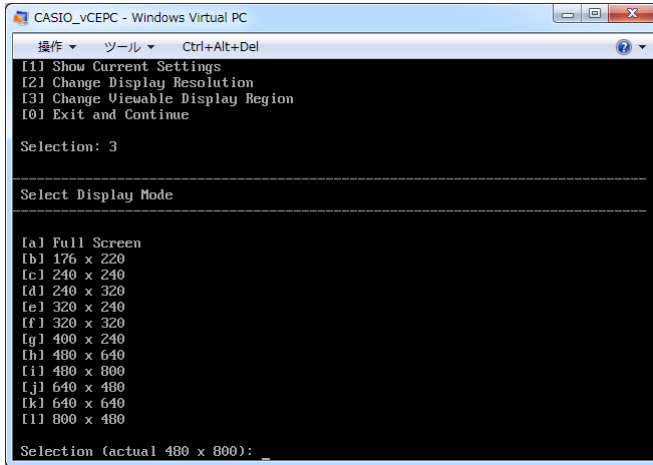
2. デバイスエミュレータの「Main Menu」が表示されるので、キーボードの「5」キーを押下します。



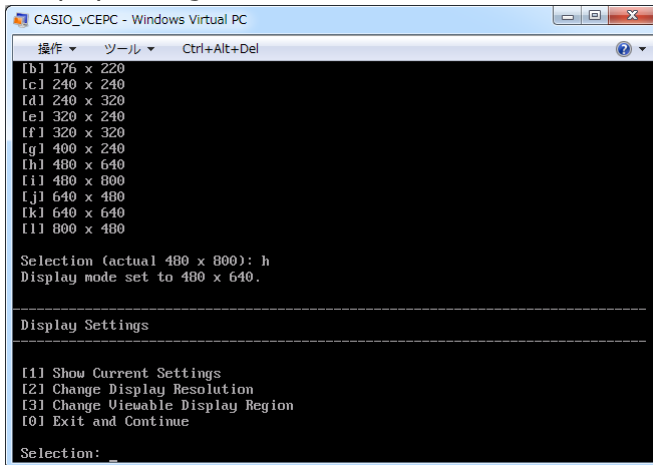
3. 「Display Settings」が表示されるので、キーボードの「3」キーを押下します。



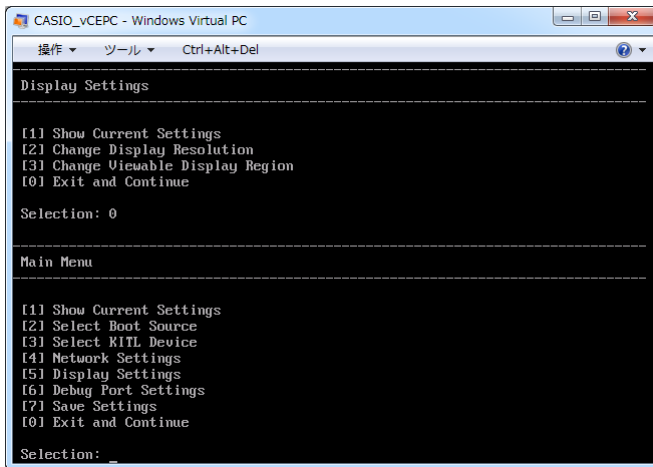
4. 「Select Display Mode」が表示されるので、キーボードの「h」キーを押下します。



5. 「Display Settings」に戻るので、キーボードの「0」キーを押下します。



6. デバイスエミュレータを今後も 480x640 で起動したい場合は、キーボードの「7」キーを押下し設定を保存した後、キーボードの「0」キーを押下してVirtualPCを起動します。デバイスエミュレータを今回だけ 480x640 で起動したい場合は、キーボードの「0」キーを押下してデバイスエミュレータを起動します。



7. デバイスエミュレータが 480x640 で起動します。



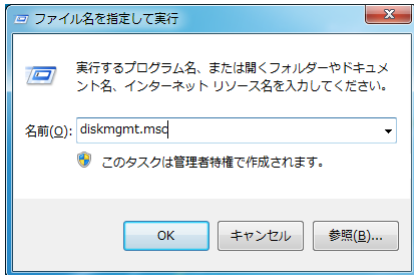
6.1.7 デバイスエミュレータのディスクサイズの拡張

デバイスエミュレータの FlashDisk のサイズはデフォルトでは 512MB です。

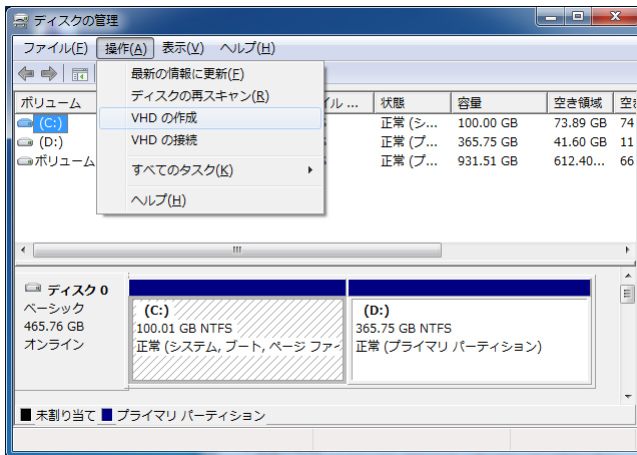
下記の手順によりデバイスエミュレータの FlashDisk のサイズを拡張することができます。

画面の表示内容は環境により異なります。ご注意ください。

1. 「ファイル名を指定して実行」を起動して、「diskmgmt.msc」と入力し、「OK」を押下します。

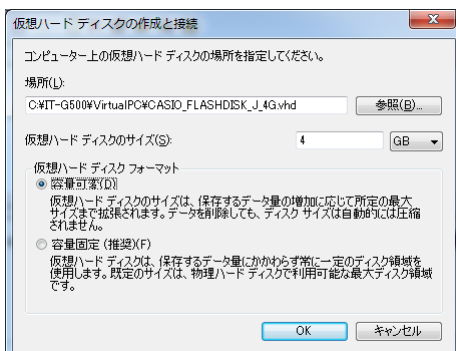


2. 「ディスクの管理」が起動したら、「操作」>「VHD の作成」を選択します。

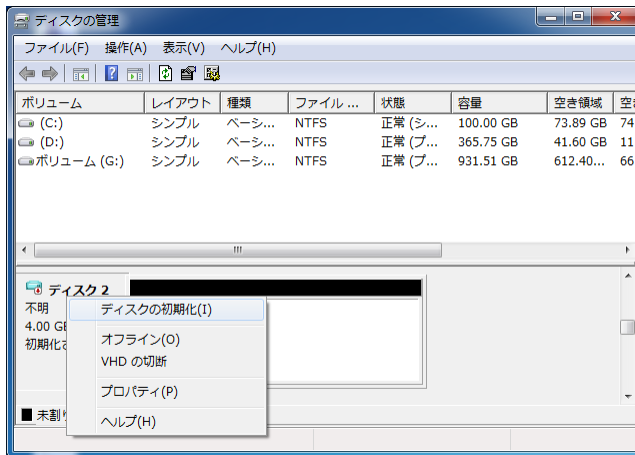


3. 「場所」に新たに作成する FlashDisk の保存場所と名称を、「仮想ハードディスクのサイズ」に新たに作成する FlashDisk のサイズをそれぞれ指定します。「仮想ハードディスクフォーマット」に「容量可変」を選択し、「OK」を押下します。

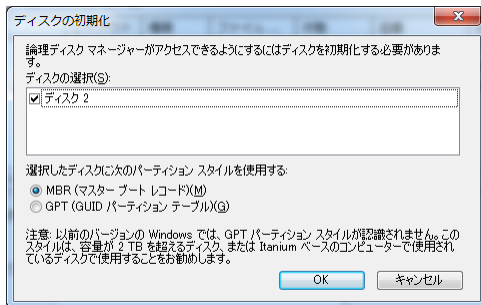
本ガイドでは、C:¥IT-G500¥VirtualPC¥CASIO_FLASHDISK_J_4G.vhd に 4GB の FlashDisk を作成します。



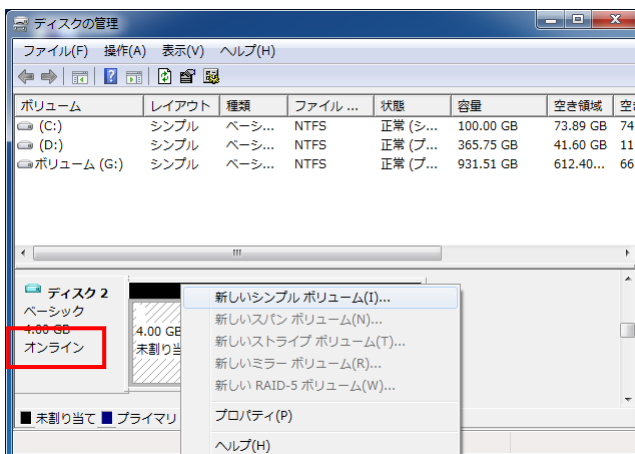
4. 手順 3.で作成したディスク上で右クリックし、「ディスクの初期化」を選択します。ディスク番号は環境により異なります。



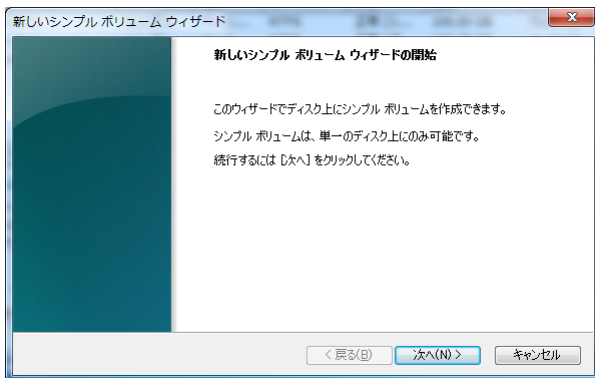
5. 「ディスクの選択」に手順 3.で作成したディスクのみを選択し、「選択したディスクに次のパーティションスタイルを使用する」に「MBR」を選択し、「OK」を押下します。



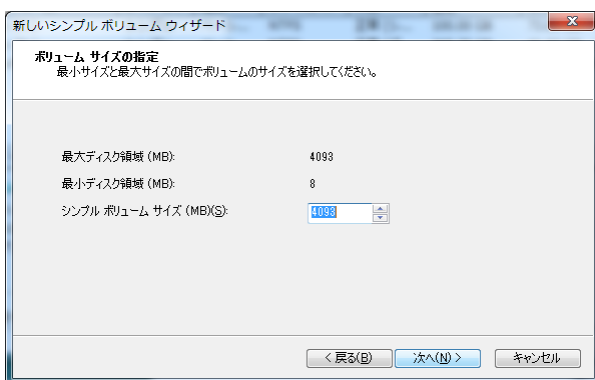
6. 下記の赤枠内が「オンライン」になっていることを確認します。「初期化されていません」と表記されている場合は、手順 4.からやり直してください。
 「未割り当て」と表記されている箇所を右クリックし、「新しいシンプルボリューム」を選択します。



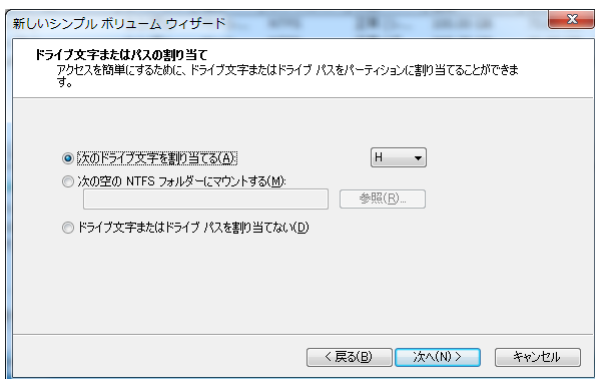
7. 「新しいシンプルボリュームウィザード」が起動するので、「次へ」を押下します。



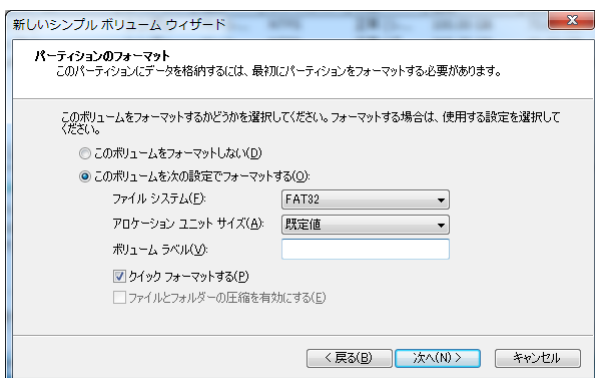
8. 「最大ディスク領域」に記載されているディスクサイズと「シンプルボリュームサイズ」が同じにして、「次へ」を押下します。



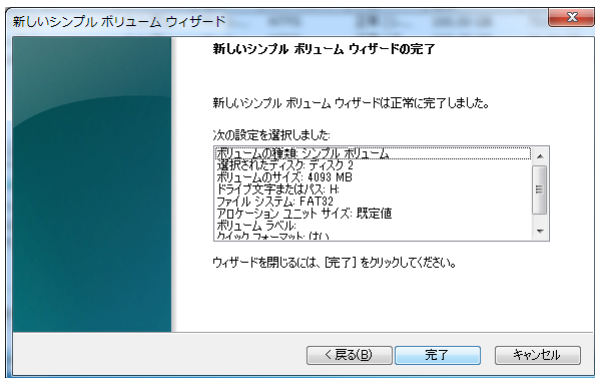
9. 「次のドライブ文字を割り当てる」に未使用のドライブ文字を選択し、「次へ」を押下します。



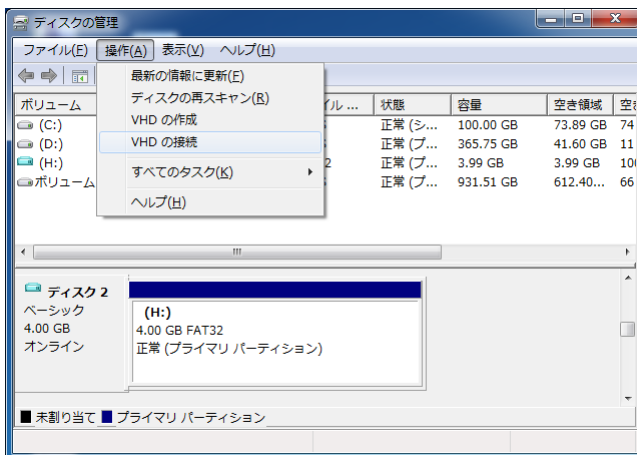
10. 「ファイルシステム」に「FAT32」、「ボリュームラベル」を空にし、「次へ」を押下します。



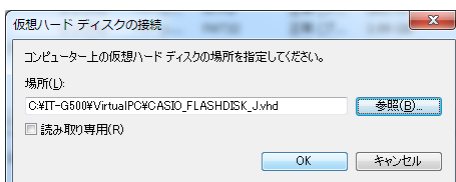
11. 表示された設定に問題がないことを確認して、「完了」を押下します。



12. 「ディスクの管理」に戻り、「操作」>「VHD の接続」を選択します。

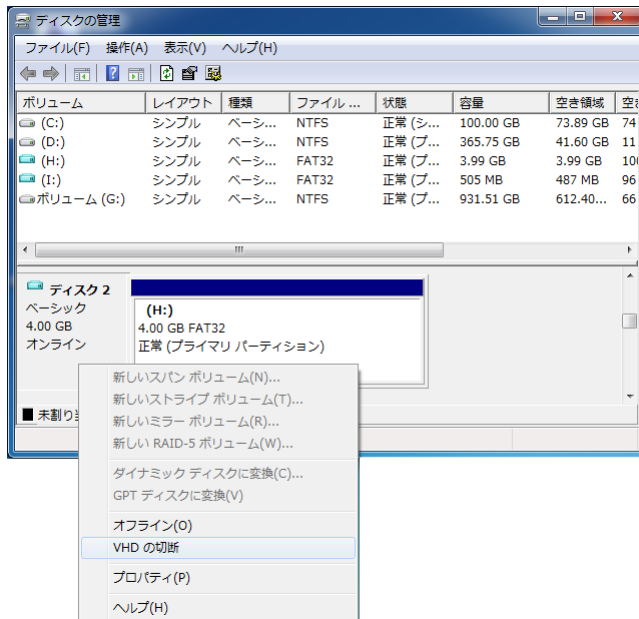


13. 「場所」に既存の FlashDisk を選択し、「OK」を押下します。
本ガイドでは、C:¥IT-G500¥VirtualPC¥CASIO_FLASHDISK_J.vhd を選択します。

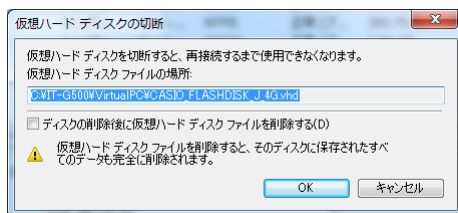


14. 手順 13. で接続したディスク内にある全フォルダ/全ファイルをエクスプローラでコピーし、手順 3. で作成したディスクにペーストします。

15. 「ディスクの管理」に戻り、手順 3. で作成したディスク上で右クリックし、「VHD の切断」を選択します。

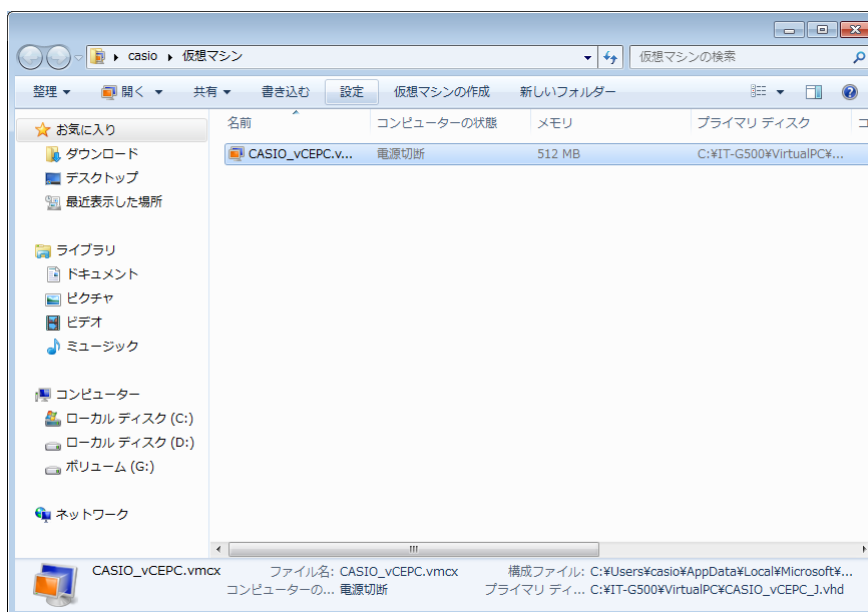


16. 「仮想ハードディスクファイルの場所」に手順 3. で作成したディスクが表示されていることを確認し、「OK」を押下します。

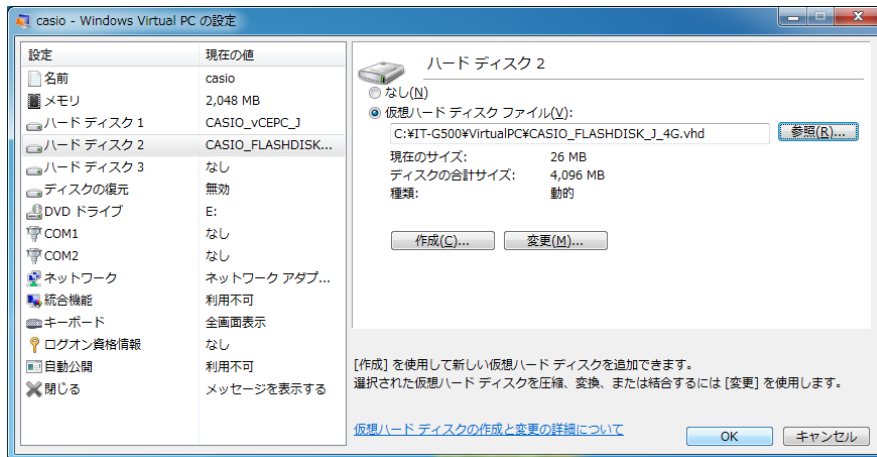


17. 同様の手順で手順 13. で接続したディスクについても、切断します。

18. 「デバイスエミュレータのインストール」で作成したデバイスエミュレータのファイル「CASIO_vCEPC.vmcx」を選択した状態で「設定」を押下します。



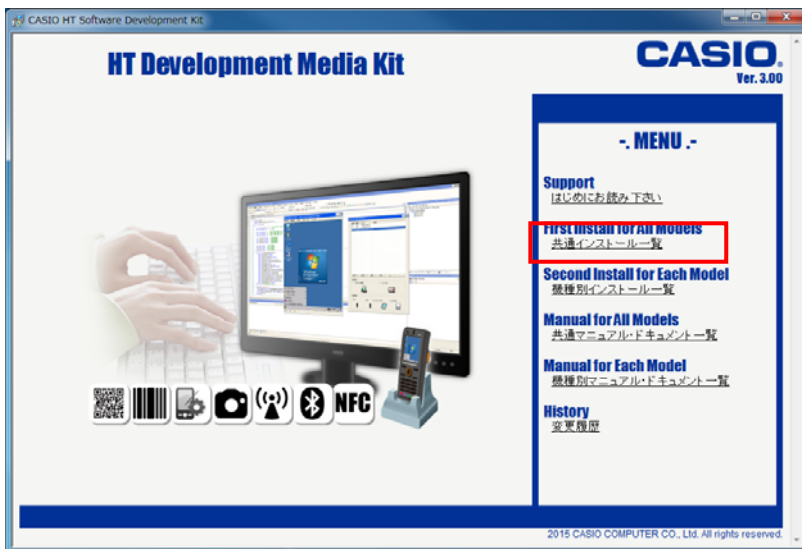
19. 「ハードディスク 2」に手順 3. で作成したディスクを選択し、「OK」を押下します。



6.2 WEH6.5 デバイスエミュレータ

6.2.1 デバイスエミュレータのインストール

1. 「共通インストーラー一覧」をクリックします。



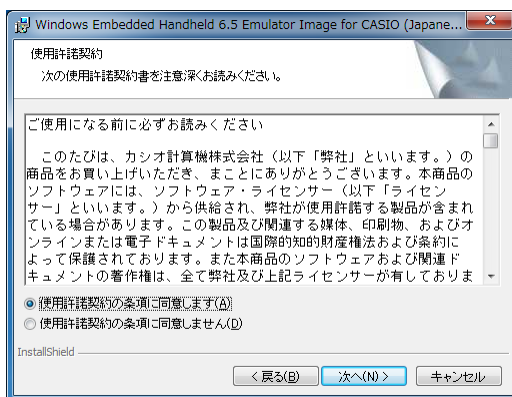
2. 「CASIO デバイスエミュレータ for Windows Embedded Handheld 6.5」をクリックします。



3. 「次へ」を押下します。



4. 使用許諾書に同意し、「次へ」を押下します。



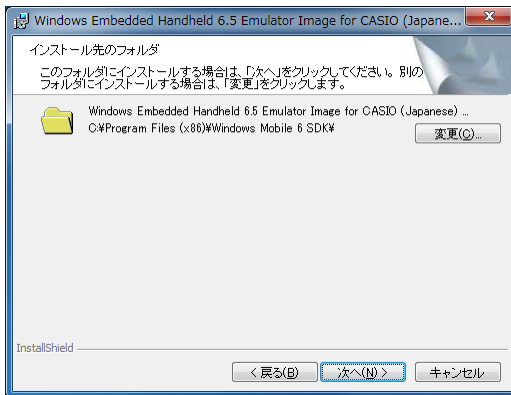
5. 「ユーザ名」、「所属」を入力し、「次へ」を押下します。



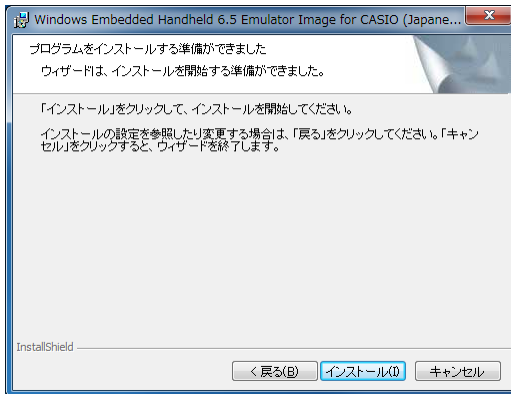
6. インストール先を変更する場合は、必要に応じて変更し、「次へ」を押下します。デフォルトでは以下にインストールします。

32bit: C:\Program Files\Windows Mobile 6 SDK

64bit: C:\Program Files (x86)\Windows Mobile 6 SDK



7. 「インストール」を押下します。



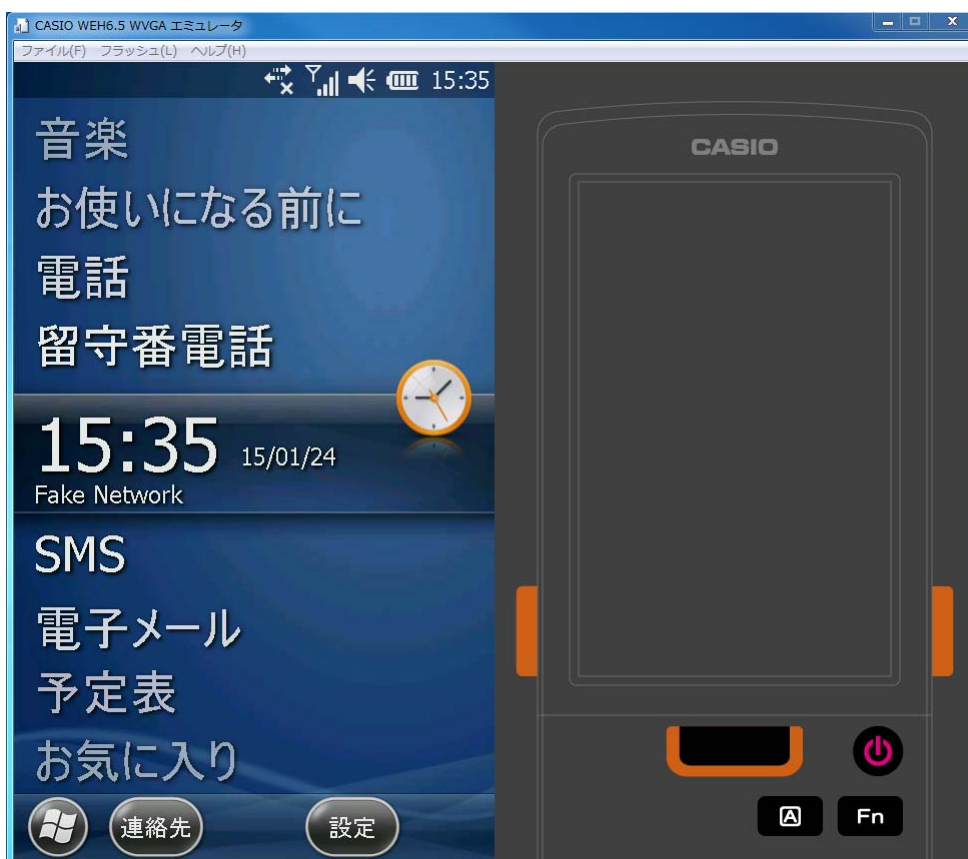
8. インストールが完了したら、「終了」を押下します。



6.2.2 デバイスエミュレータの起動

WEH6.5 デバイスエミュレータは、WVGA (480x800)、VGA (480x640) があります。ここでは、WVGA サイズの起動方法を記載します。

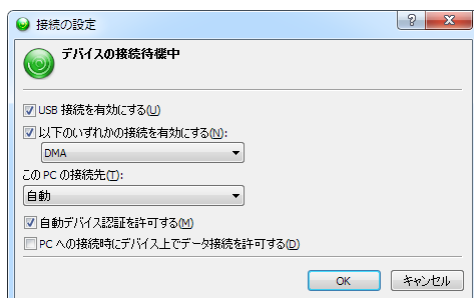
1. スタートメニューの「CASIO Device Emulator」>「WEH6.5 WVGA」を選択します。VGA の場合は「WEH6.5 VGA」を選択してください。
2. デバイスエミュレータが起動します。



6.2.3 Windows Mobile デバイスセンターとの接続

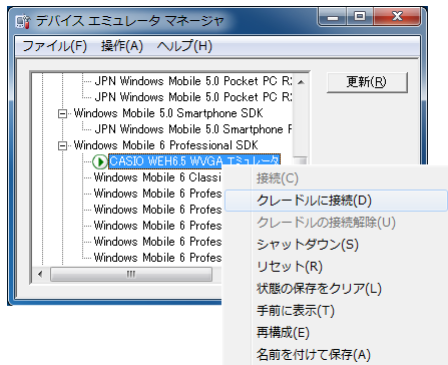
Windows Mobile デバイスセンターの設定


1. Windows Mobile デバイスセンターを起動し、「モバイルデバイスの設定」>「接続の設定」を選択します。
2. 「以下のいずれかの接続を有効にする(N):」にチェックを入れ、「DMA」を選択します。



Windows Mobile デバイスセンターとの接続

1. Visual Studio 2008 を起動し、「ツール(T)」>「デバイス エミュレータ マネージャ(L)...」を選択します。
2. 「利用可能なエミュレータ:」内にある「CASIO WEH6.5 WVGA エミュレータ」上で右クリックし、「クレードルに接続」を選択します。



3. Windows Mobile デバイスセンターが起動し、エミュレータのステータスバー上に  のアイコンが表示されていることを確認してください。

6.3 IO シミュレータ

引き続き I/O シミュレータを使用するために必要なソフトウェアのインストール方法を説明します。

6.3.1 IO シミュレータのインストール

旧バージョン等の IO シミュレータを既にインストール済みの場合は、アンインストールした後、下記の手順にしたがってインストールしてください。コントロールパネルの「プログラムと機能」から「CASIO デバイスエミュレータ」を選択してアンインストールしてください。

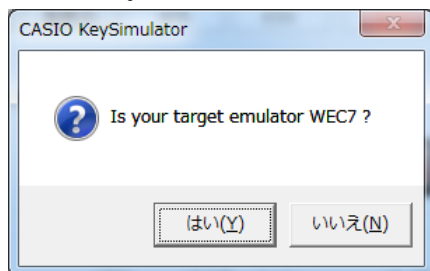
1. 本商品の CD-ROM から、「共通インストーラー一覧」の中の「CASIO IO シミュレータ」をインストールします。



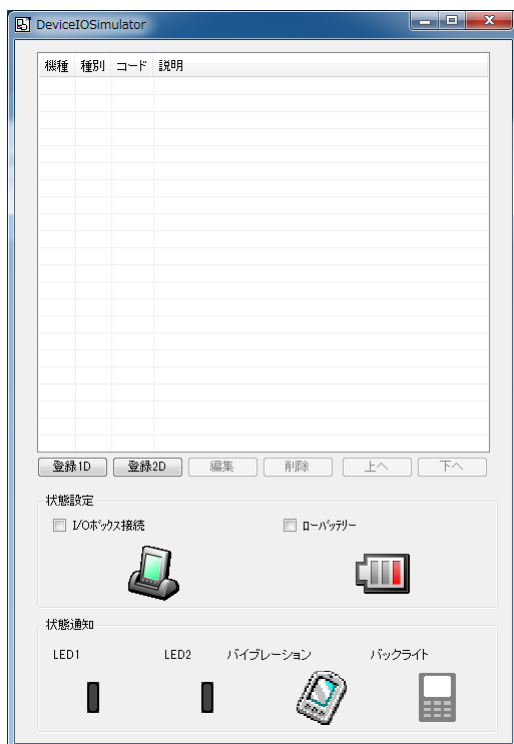
2. WEC7 デバイスエミュレータを使用する場合は、以下のフォルダの共有を有効に設定します。アクセス権限について、読み取りと書き込みを有効に設定します。
C:¥ProgramData¥CASIO

6.3.2 IO シミュレータの起動

1. スタートメニューの「CASIO Device IO Simulator」を選択します。
2. 下記のメッセージが表示されるので、WEC7 デバイスエミュレータを使用する場合は、「はい」を選択してください。



3. IO シミュレータおよびキーシミュレータが起動します。



6.3.3 IO シミュレータの操作

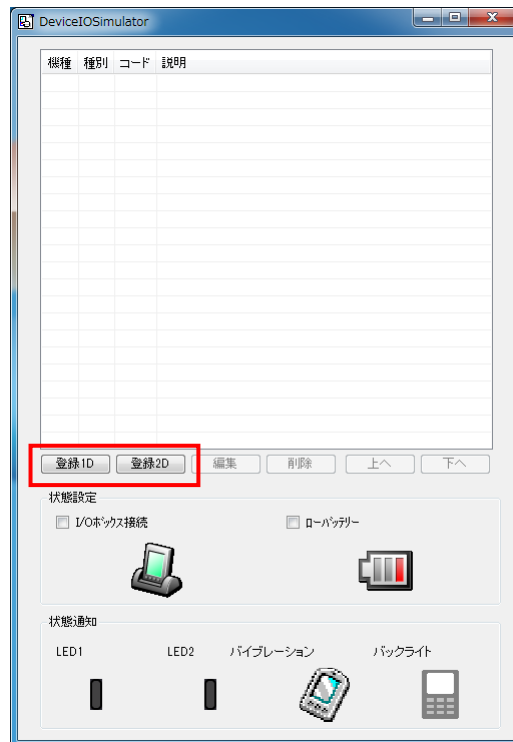
入力バーコードの登録、状態設定、状態通知を擬似的に行うものです。

入力バーコードの登録

(1) 登録選択

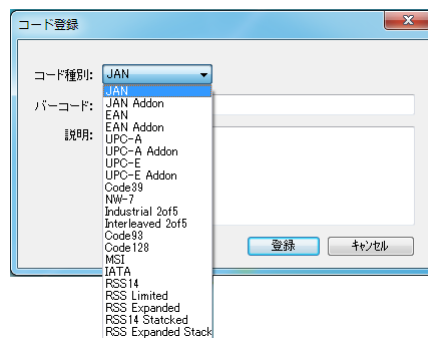
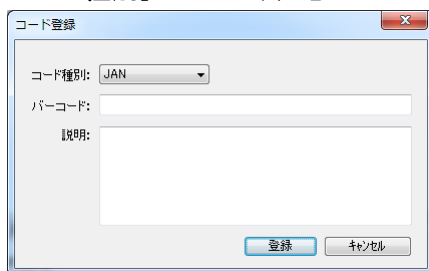
I/O シミュレータの[登録 1D]ボタンまたは[登録 2D]ボタンをクリックすると、バーコード登録画面に移ります。

1D コードの登録には[登録 1D]ボタン、
2D コードの登録には[登録 2D]ボタンを使用します。



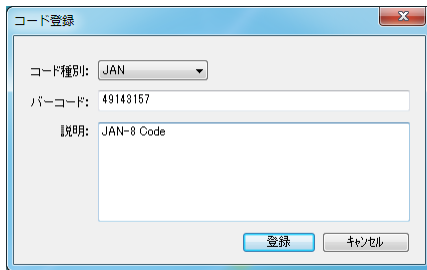
(2) バーコード登録

「コード種別」を選択し、任意のバーコード種別を選択します。



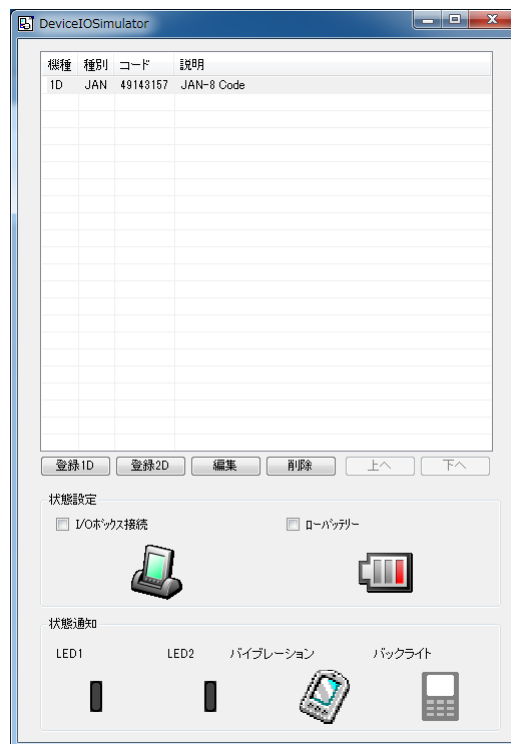
(3) バーコードのデータと説明を登録

任意のバーコードを入力し、「登録」を押下します(必要に応じて「説明」も入力してください)。



(4) 登録完了

登録が完了すると、登録したバーコードを一覧表に表示します。この登録操作を繰り返して、デバッグに必要なバーコードをあらかじめ登録しておいてください。

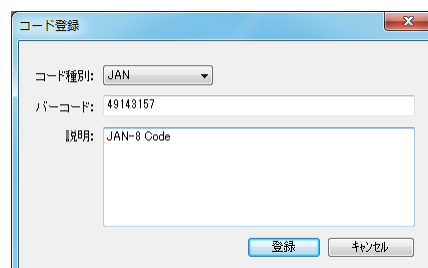


※ バーコード登録時の注意事項

登録可能なバーコードは、デバイスエミュレータ上で読取可能となっているコードに限ります。デバイスエミュレータ起動直後はすべてのコードが読取可能となっていますが、アプリケーションを起動した場合は、そのアプリケーションで設定した読取可能コードのみが、登録可能となります。

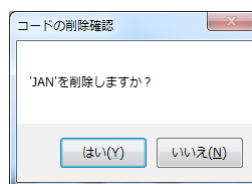
(5) 登録内容の編集

編集したいバーコードを一覧表から選択し、「編集」ボタンをクリックすると、右記の画面を表示しますので、編集してください。



(6) 登録内容の削除

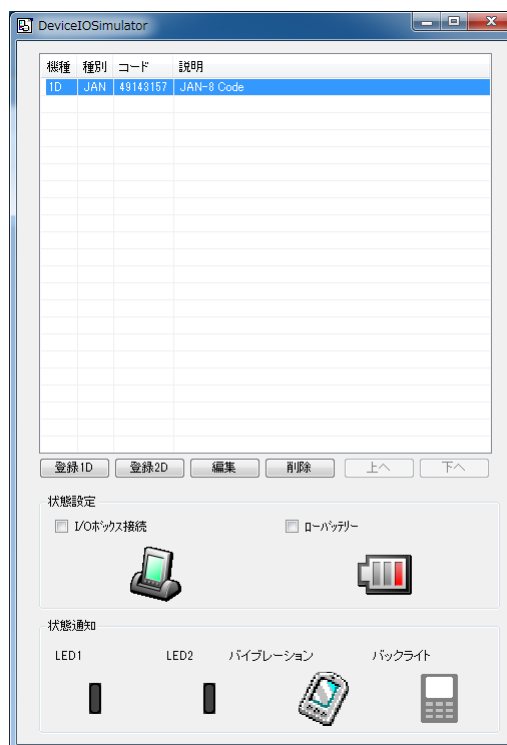
削除したいバーコードを一覧表から選択し、「削除」ボタンをクリックすると、右記の画面を表示しますので、確認後、削除してください。



(7) バーコードの読み込み

登録したバーコードを選択状態にすると、デバイスエミュレータに読み込ませることができます。

任意のバーコードを選択してから、デバイスエミュレータのトリガーキーを押してください。



状態設定機能

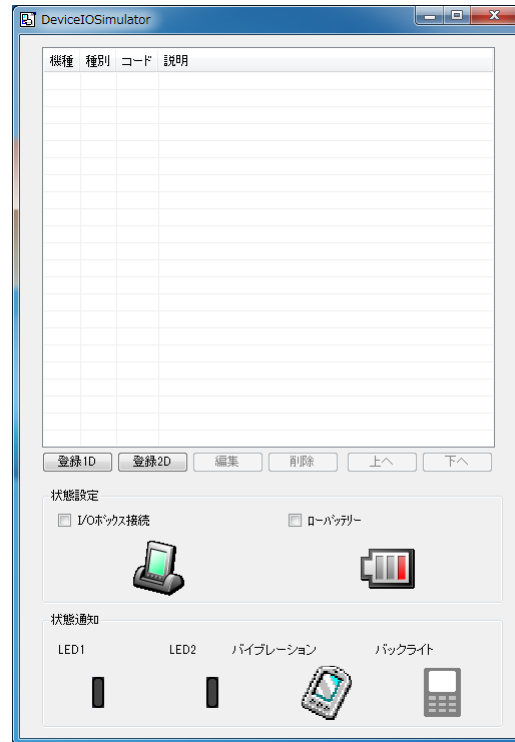
I/O シミュレータの状態設定にある「 I/O ボックス接続」/「 ローバッテリー」にチェックを入れると、それぞれの場合の動きを擬似的に再現します。

- I/O ボックス接続

デバイスエミュレータが I/O ボックスに接続したことを通知します。
アプリケーション側で I/O ボックス接続のイベントを確認することができます。

- ローバッテリー

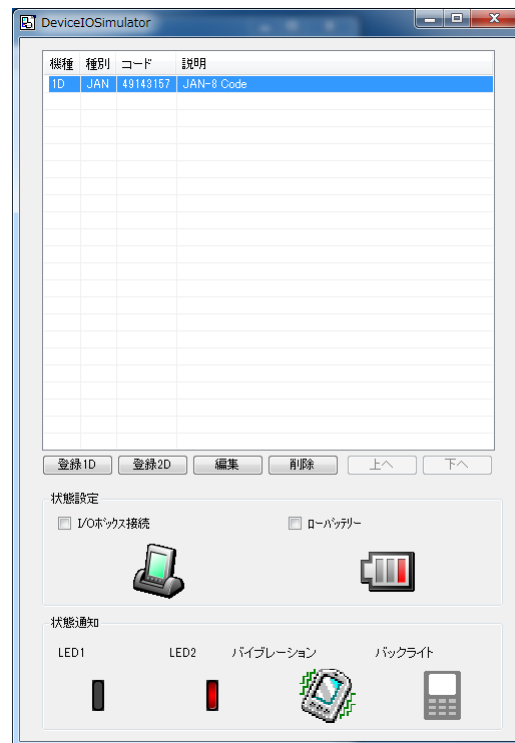
デバイスエミュレータに対して、ローバッテリーが発生したことを通知します。
アプリケーション側で、ローバッテリー発生イベントを確認することができます。



状態通知機能

デバイスエミュレータで発生した状態の変化を、擬似的に表現します。

- LED 表示
デバイスエミュレータが LED を点灯すると、I/O シミュレータの LED アイコンが変化します。
- バイブレーション
デバイスエミュレータがバイブレーション動作を行うと、I/O シミュレータのバイブレーションアイコンが変化します。

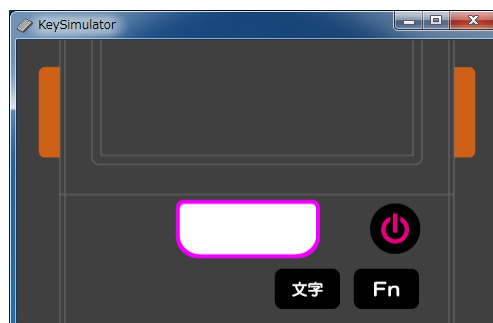


キー通知機能

WEC7 のデバイスエミュレータではキーシミュレータからキーを入力することにより、キーイベントをデバイスエミュレータに通知することができます。

キーシミュレータに存在しないキーについては、PC のキーボードから入力してください。

キーを押下すると、デバイスエミュレータに通知することができます。



7. Visual Studio によるアプリケーションの開発

Visual Studio 2008 を使用した、アプリケーションの開発手順を説明します。

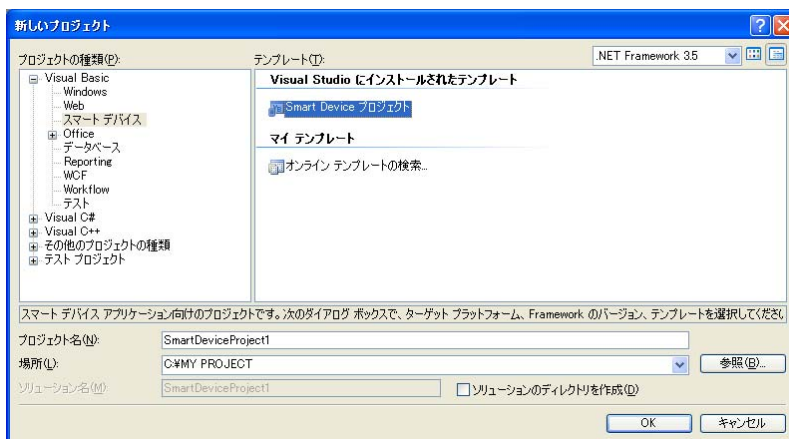
例題は、システムライブラリを利用した、LED を 5 秒間だけ赤く点滅させるアプリケーションです。
このアプリケーションを、3 種類のプログラミング言語 (Visual Basic.NET / C# / C++) で作成します。

この章の例題プログラムは、デバイスエミュレータでも動作します。

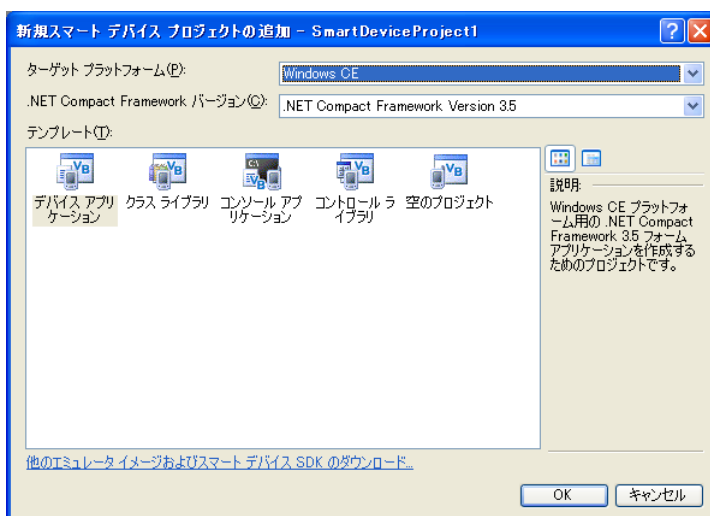
7.1 システムライブラリを使用したプログラミング (Visual Basic)

ここでは、LED を 5 秒間だけ赤く点滅させる GUI アプリケーションを、VB.NET 言語で作成します。

1. Visual Studio2008 で、新しい VB.NET の「デバイスアプリケーション」を作成します。

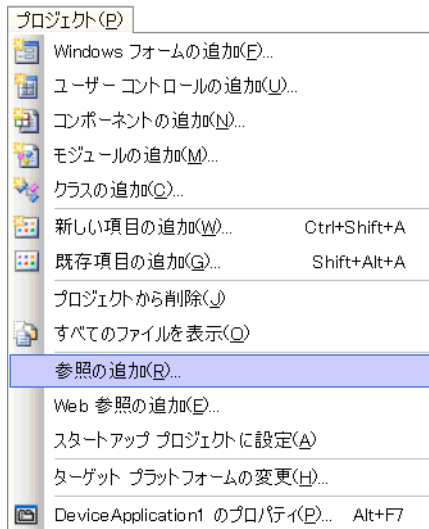


2. ターゲットに「WindowsCE」を指定します。

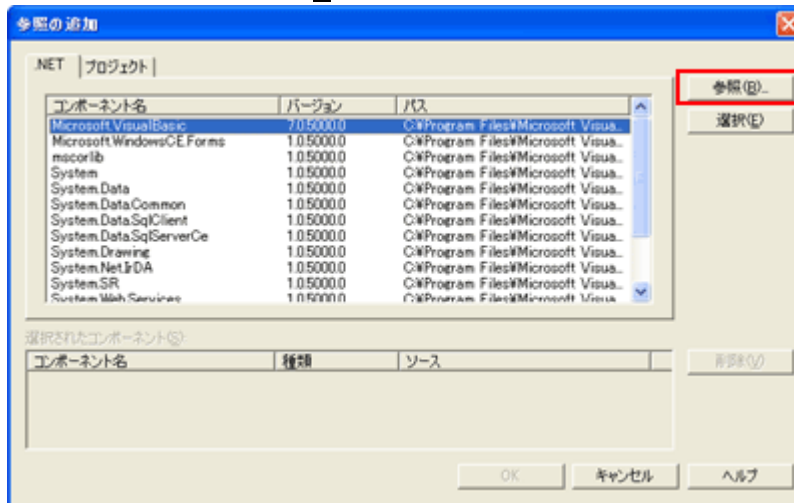


テンプレートから「デバイスアプリケーション」を選択します。

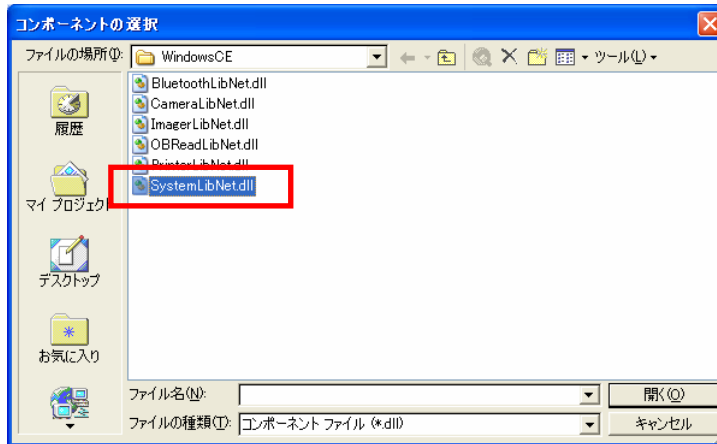
3. [プロジェクト]メニューから、[参照の追加(R)...]をクリックします。



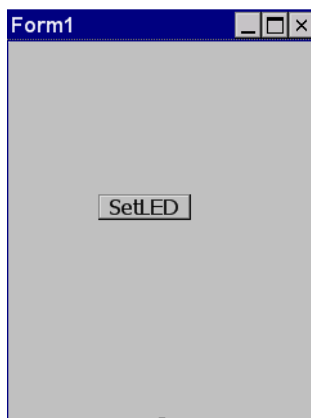
4. 「参照」タブまたは、[参照(B)]ボタンをクリックします。



5. クラスライブラリのインストール先フォルダを参照し、SystemLibNet.dll を選択します。
32bit: C:\Program Files\CASIO\MBSYS\WindowsCE
64bit: C:\Program Files (x86)\CASIO\MBSYS\WindowsCE



6. [OK] ボタンをクリックして、「参照の追加」ダイアログを閉じます。
7. フォームのプロパティで、「Size」を「240,320」に変更します。
8. フォームにボタンを追加し、ボタンの表示文字列を「SetLED」に変更します。



9. 「SetLED」ボタンをダブルクリックすると、フォームのコード画面が表示されます。

10. ボタン押下時のイベント関数に、以下のコードを追加してください。

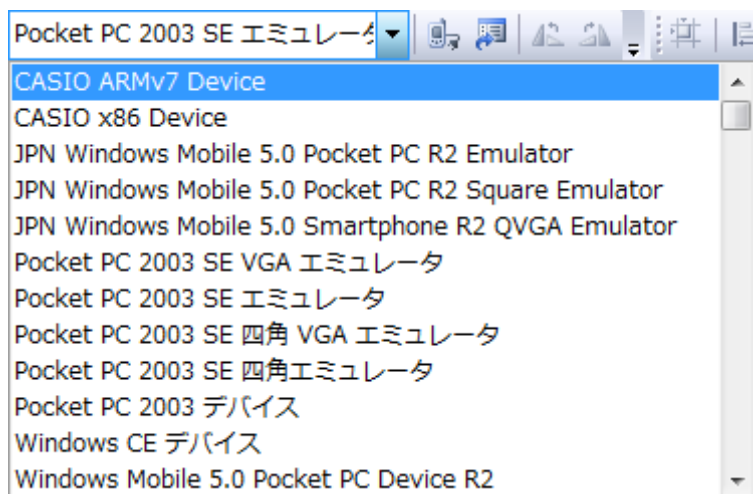
```
Dim result As Int32
Dim msg As String

' 赤LEDを0.5秒点灯 0.5秒消灯を5回繰り返す
result = Calib.SystemLibNet.Api.SysSetLED(Calib.SystemLibNet.Def.LED_RED, 5, 8, 8)

' .NETの「true」値は、「-1」、「false」値は、「0」
If result = -1 Then
    ' 点灯成功 点灯色を取得
    result = Calib.SystemLibNet.Api.SysGetLED()
    Select Case (result And &HF)
        Case Calib.SystemLibNet.Def.LED_OFF
            msg = "LED_OFF"
        Case Calib.SystemLibNet.Def.LED_RED
            msg = "LED_RED"
        Case Calib.SystemLibNet.Def.LED_GREEN
            msg = "LED_GREEN"
        Case Calib.SystemLibNet.Def.LED_ORANGE
            msg = "LED_ORANGE"
        Case Else
            msg = "LED_UNKNOWN"
    End Select
    MessageBox.Show(msg, "LED")
End If
```

11. [ビルド]→[ソリューションのビルド]を選択し、正常にビルドできることを確認します。

12. 端末と PC 間にて Windows Mobile デバイスセンターで接続を確立します。
13. ご利用になる端末にあわせてターゲットデバイスを選択します。



ターゲットデバイス	端末
CASIO ARMv7 Device	IT-G500 (WEC7 モデル)
CASIO ARMv5 Device	DT-X100 / DT-X200
CASIO x86 Device	WEC7 のデバイスエミュレータ
Windows Mobile 6 Professional Device	IT-G500 (WEH6.5 モデル)
CASIO WEH6.5 WVGA エミュレータ	WEH6.5 の WVGA デバイスエミュレータ
CASIO WEH6.5 VGA エミュレータ	WEH6.5 の VGA デバイスエミュレータ

14. 「デバッグ(D)」>「デバッグ開始(S)」または「デバッグなしで開始(H)」を選択します。

ソースコードがコンパイルされて、端末の ¥Program Files¥<プロジェクト名> フォルダにアプリケーションがコピーされます。

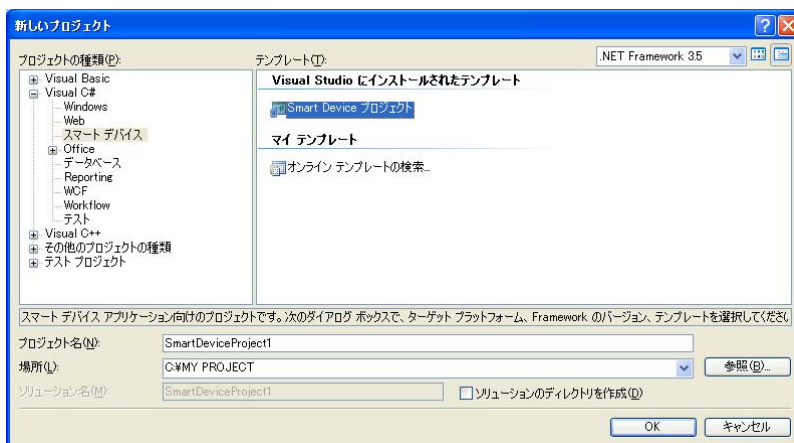
また、SystemLibNet.dll も同フォルダにコピーされます。

端末上で、プログラムが正常に動作することを確認してください。

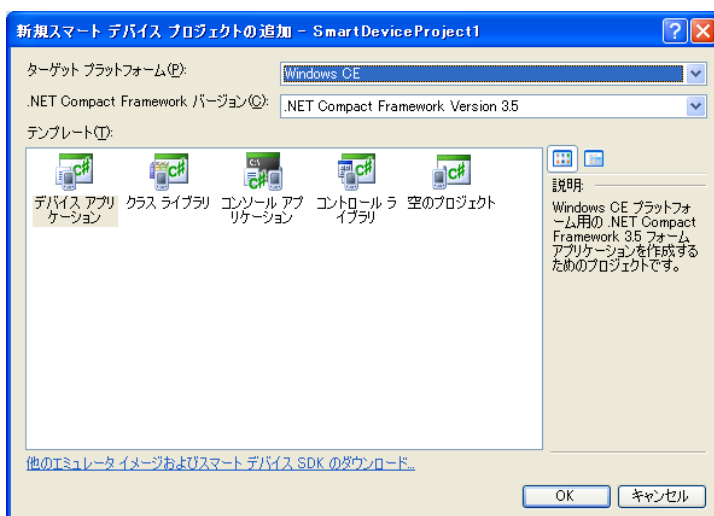
7.2 システムライブラリを使用したプログラミング (Visual C#)

ここでは、LED を 5 秒間だけ赤く点滅させる GUI アプリケーションを、C# 言語で作成します。

1. Visual Studio2008 で、新しい C# の「デバイスアプリケーション」を作成します。

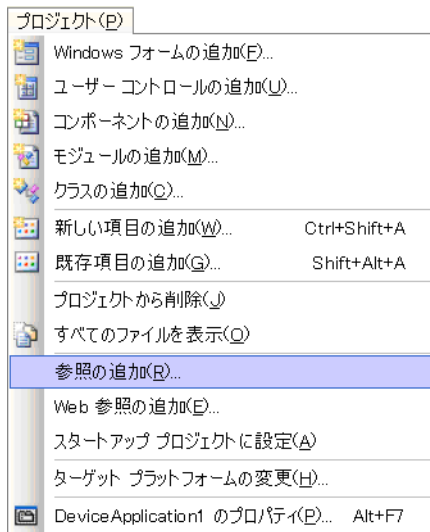


2. ターゲットに「WindowsCE」を指定します。

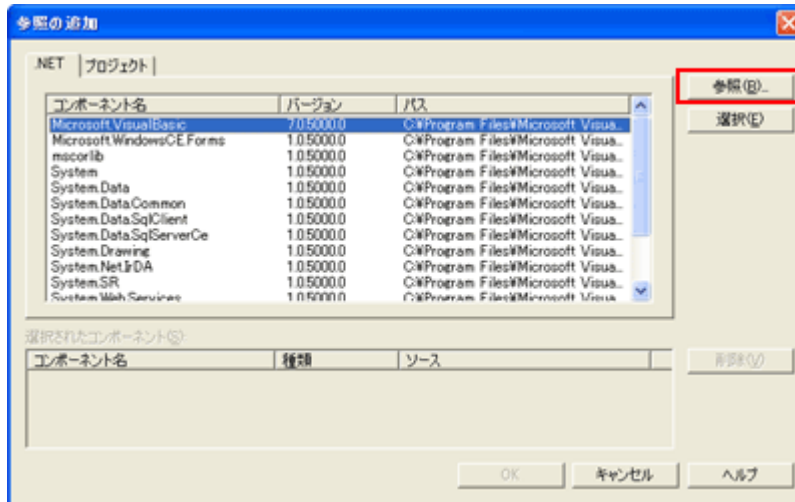


テンプレートから「デバイスアプリケーション」を選択します。

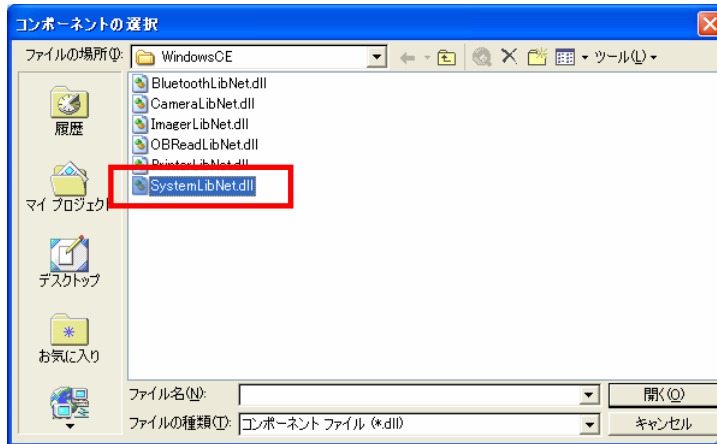
3. [プロジェクト]メニューから、[参照の追加(R)...]をクリックします。



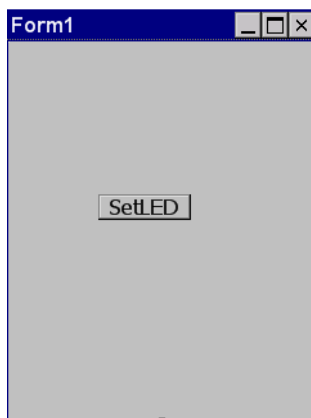
4. 「参照」タブまたは、[参照(B)]ボタンをクリックします。



5. クラスライブラリのインストール先フォルダを参照し、SystemLibNet.dll を選択します。
32bit: C:\Program Files\CASIO\MBSYS\WindowsCE
64bit: C:\Program Files (x86)\CASIO\MBSYS\WindowsCE



6. [OK] ボタンをクリックして、「参照の追加」ダイアログを閉じます。
7. フォームのプロパティで、「Size」を「240,320」に変更します。
8. フォームにボタンを追加し、ボタンの表示文字列を「SetLED」に変更します。



9. 「SetLED」ボタンをダブルクリックすると、フォームのコード画面が表示されます。

10. ソースファイルの先頭に、以下のコードを追加します。

```
using Calib;
```

11. ボタン押下時のイベント関数に、以下のコードを追加します。

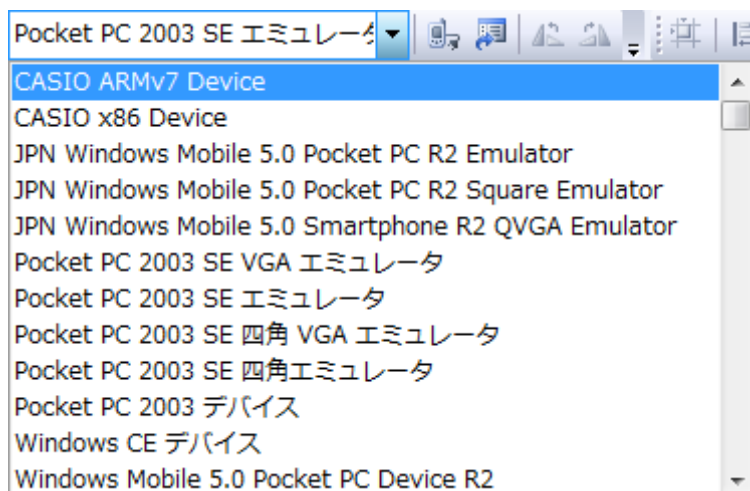
```
Int32 result = new Int32();
string msg;

// 赤 LED を 0.5 秒点灯 0.5 秒消灯を 5 回繰り返す
result = SystemLibNet.Api.SysSetLED(SystemLibNet.Def.LED_RED, 5, 8, 8);

// .NET の「true」値は、「-1」、「false」値は、「0」
if(result == -1)
{ // 点灯成功 点灯色を取得
  result = SystemLibNet.Api.SysGetLED();
  switch(result & 0x0000000F)
  {
    case SystemLibNet.Def.LED_OFF:
      msg = "LED_OFF";
      break;
    case SystemLibNet.Def.LED_RED:
      msg = "LED_RED";
      break;
    case SystemLibNet.Def.LED_GREEN:
      msg = "LED_GREEN";
      break;
    case SystemLibNet.Def.LED_ORANGE:
      msg = "LED_ORANGE";
      break;
    default:
      msg = "LED_UNKNOWN";
      break;
  }
  MessageBox.Show(msg, "LED");
}
```

12. [ビルド]→[ソリューションのビルド]を選択し、正常にビルドできることを確認します。

13. 端末と PC 間にて Windows Mobile デバイスセンターで接続を確立します。
14. ご利用になる端末にあわせてターゲットデバイスを選択します。



ターゲットデバイス	端末
CASIO ARMv7 Device	IT-G500 (WEC7 モデル)
CASIO ARMv5 Device	DT-X100 / DT-X200
CASIO x86 Device	WEC7 のデバイスエミュレータ
Windows Mobile 6 Professional Device	IT-G500 (WEH6.5 モデル)
CASIO WEH6.5 WVGA エミュレータ	WEH6.5 の WVGA デバイスエミュレータ
CASIO WEH6.5 VGA エミュレータ	WEH6.5 の VGA デバイスエミュレータ

15. 「デバッグ(D)」>「デバッグ開始(S)」または「デバッグなしで開始(H)」を選択します。

ソースコードがコンパイルされて、端末の ¥Program Files¥<プロジェクト名> フォルダにアプリケーションがコピーされます。

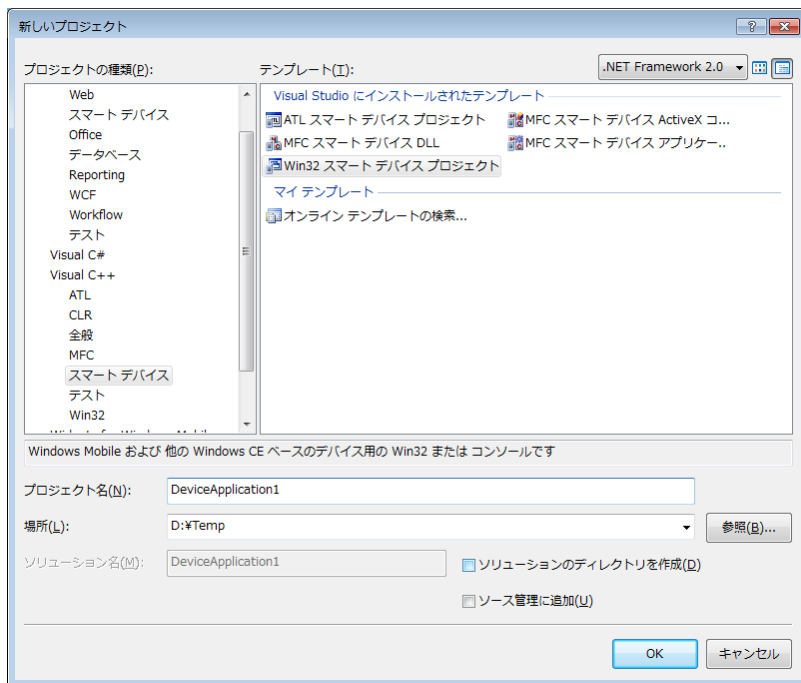
また、SystemLibNet.dll も同フォルダにコピーされます。

端末上で、プログラムが正常に動作することを確認してください。

7.3 システムライブラリを使用したプログラミング (Visual C++)

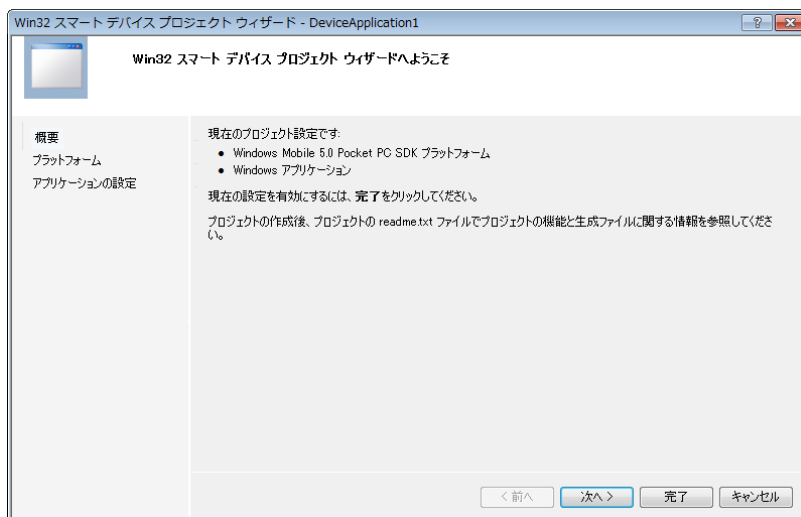
ここでは、LED を 5 秒間だけ赤く点滅させるコンソールアプリケーションを、C++ 言語で作成します。プロジェクトの作成手順は以下のとおりです。

1. Visual Studio2008 で、Visual C++ の「Win32 スマートデバイスプロジェクト」を作成します。

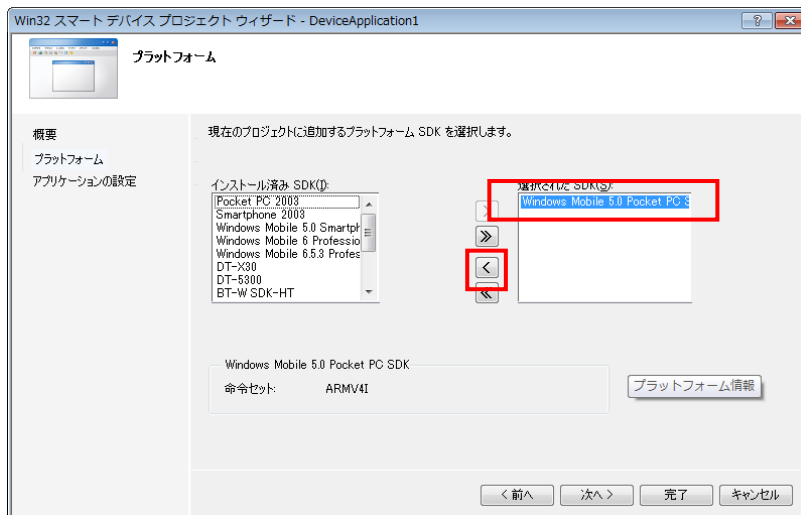


プロジェクト名は、「DeviceApplication1」としています。

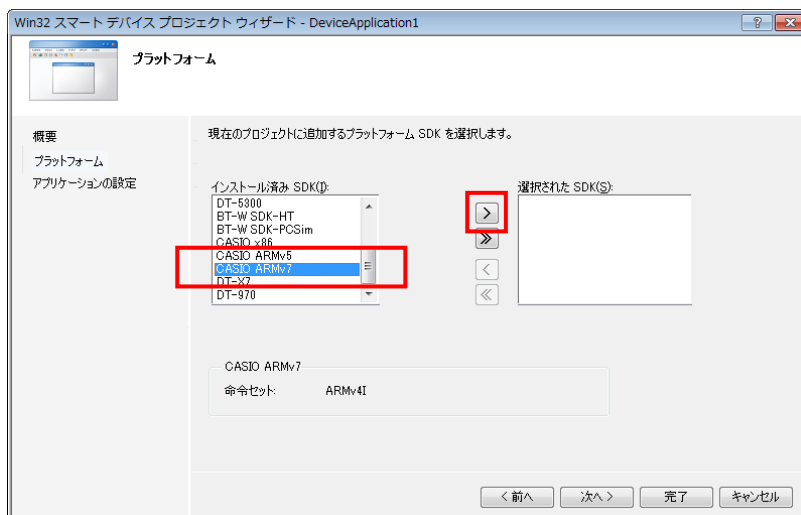
2. 「次へ >」を選択します。



3. 「**選択された SDK(S)**」内にある、デフォルトのプラットフォームを選択して、「<」をクリックします。

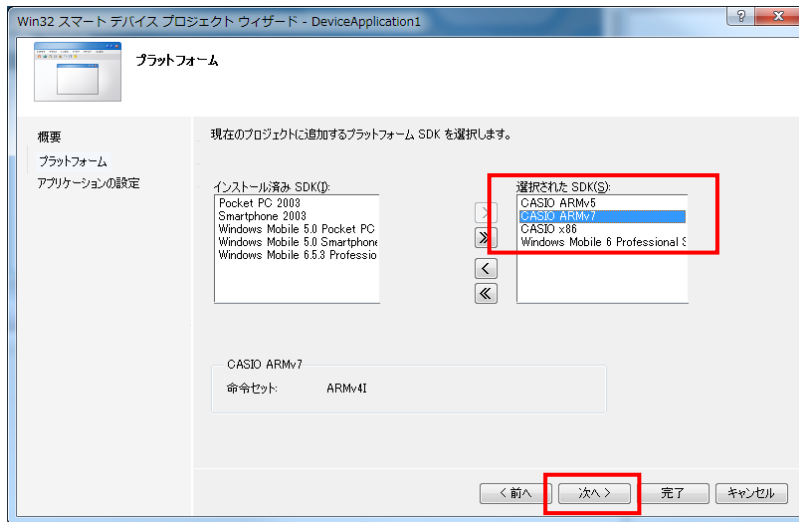


4. 「**インストール済み SDK(I)**」の中からご利用の端末の SDK を選択した状態で「>」をクリックします。お使いになる端末により、選択する SDK が異なります。ここでは、4 つの SDK すべてを選択します。

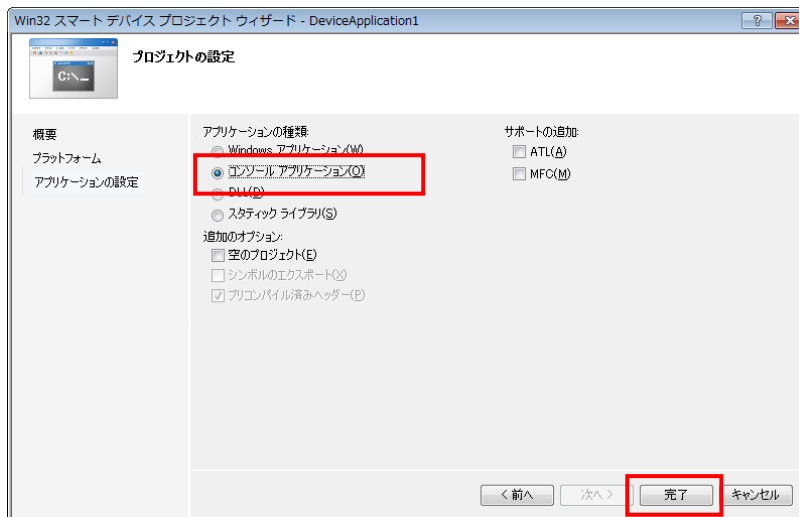


SDK	端末
CASIO ARMv7	IT-G500 (WEC7 モデル)
CASIO ARMv5	DT-X100 / DT-X200
CASIO x86	WEC7 のデバイスエミュレータ
Windows Mobile 6 Professional SDK	IT-G500 (WEH6.5 モデル)/WEH6.5 のデバイスエミュレータ

5. 「**選択された SDK(S)**」に「**CASIO ARMv7**」、「**CASIO ARMv5**」、「**CASIO x86**」、「**Windows Mobile 6 Professional SDK**」があることを確認し、「**次へ >**」をクリックします。



6. 「**アプリケーションの種類**」に「**コンソールアプリケーション(O)**」を選択し、「**完了**」をクリックします。



7. DeviceApplication1.cpp を開き、「#include <comctrl.h>」の次に以下のコードを追加します。

```
#include <SystemLib.h> // システムライブラリを使用
#if !defined(_countof)
#define _countof(_Array) (sizeof(_Array) / sizeof(_Array[0]))
#endif
```

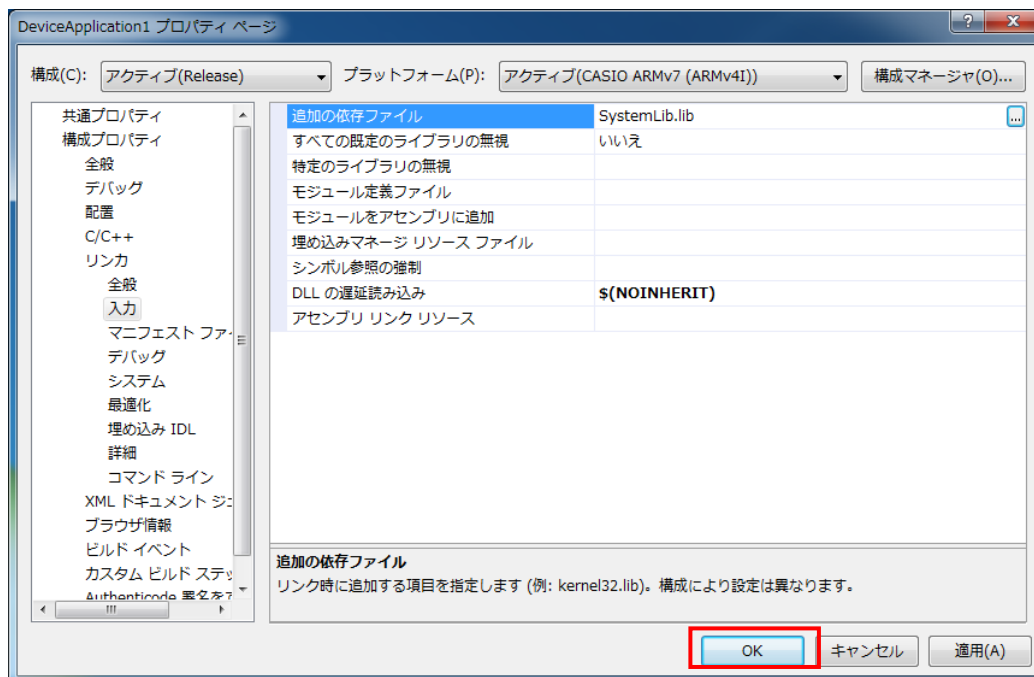
8. DeviceApplication1.cpp 内にある Main 関数に以下のコードを追加します。

```
DWORD result;
TCHAR msg[16];
// 赤 LED を 0.5 秒点灯 0.5 秒消灯を 5 回繰り返す
result = SysSetLED(LED_RED, 5, 8, 8);

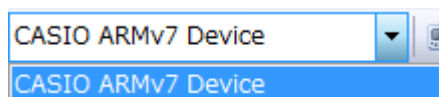
if(result == TRUE)
{ // 点灯成功 点灯色を取得
  result = SysGetLED();
  switch(result & 0x0000000F) {
  case LED_OFF:
    wcscpy_s(msg, _countof(msg), TEXT("LED_OFF"));
    break;
  case LED_RED:
    wcscpy_s(msg, _countof(msg), TEXT("LED_RED"));
    break;
  case LED_GREEN:
    wcscpy_s(msg, _countof(msg), TEXT("LED_GREEN"));
    break;
  case LED_ORANGE:
    wcscpy_s(msg, _countof(msg), TEXT("LED_ORANGE"));
    break;
  default:
    wcscpy_s(msg, _countof(msg), TEXT("LED_UNKNOWN"));
    break;
  }
  MessageBox(NULL, msg, TEXT("LED"), MB_OK);
}
```

9. [ビルド]→[コンパイル]を選択し、正常にコンパイルできることを確認します。
10. [プロジェクト(P)]→[DeviceApplication1 のプロパティ(P)...]を選択します。

11. [構成プロパティ]→[リンカ]→[入力]を選択し、[追加の依存ファイル]に、「SystemLib.lib」と入力して、[OK]をクリックします。プラットフォームはご利用の端末にあわせて選択してください。ここでは、「CASIO ARMv7 (ARMv4I)」を選択します。



12. ターゲットデバイスをご利用の端末にあわせて選択してください。ここでは、「CASIO ARMv7 Device」を選択します。



ターゲットデバイス	端末
CASIO ARMv7 Device	IT-G500 (WEC7 モデル)
CASIO ARMv5 Device	DT-X100 / DT-X200
CASIO x86 Device	WEC7 のデバイスエミュレータ
Windows Mobile 6 Professional Device	IT-G500 (WEH6.5 モデル)
CASIO WEH6.5 WVGA エミュレータ	WEH6.5 の WVGA デバイスエミュレータ
CASIO WEH6.5 VGA エミュレータ	WEH6.5 の VGA デバイスエミュレータ

13. 端末と PC 間の接続を確立し、「デバッグ(D)」>「デバッグ開始(S)」または「デバッグなしで開始(H)」を選択します。

ソースコードがコンパイルされて、端末の ¥Program Files¥<プロジェクト名> フォルダにアプリケーションがコピーされます。

端末上で、プログラムが正常に動作することを確認してください。

注意事項

VC++でMFCを利用したプロジェクトをコンパイルしたときに、下記のエラーが出る場合があります。

```
C:\Program Files\Microsoft Visual Studio 9.0\VC\ce\atl\mf\include\atlconv.h(695) :  
error C2039: 'lstrlenW' : 'ATL' のメンバではありません。
```

本エラーが発生した場合は、Microsoft 社製の `atlconv.h` を下記のように編集してください。

```
695     inline int ocslen(_In_z_ LPCOLESTR x) throw() { return ATL::lstrlenW(x); }  
  
// "ATL::"を削除  
695     inline int ocslen(_In_z_ LPCOLESTR x) throw() { return lstrlenW(x); }
```

`atlconv.h` はデフォルトでは下記にあります。

32bit: C:\Program Files\Microsoft Visual Studio 9.0\VC\ce\atl\mf\include

64bit: C:\Program Files (x86)\Microsoft Visual Studio 9.0\VC\ce\atl\mf\include

8. 自動復旧

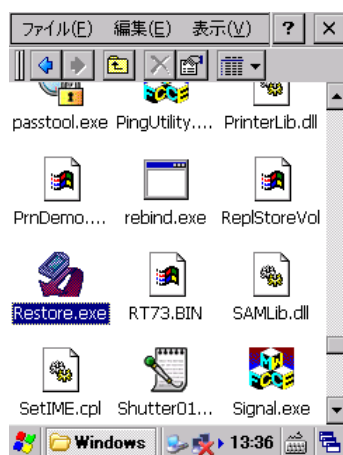
自動復旧ツール(Restore.exe)は、電池消耗等によりRAM上のファイルやレジストリ等が消失してしまった場合に、元の状態に戻すためのツールです。

フルリセットを伴う操作ですので、作成したアプリケーションをインストールしたあとに実行することをお勧めします。

[設定方法]

アプリケーションインストール完了後、Windowsフォルダ内にある「Restore.exe」を実行します。

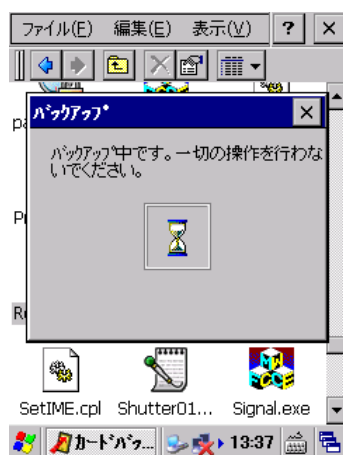
1. Windowsフォルダ内の「Restore.exe」を実行



2. 「はい(Y)」を選択



3. 自動的にバックアップを開始



4. バックアップが終了すると、以下のメッセージを表示



[設定の確認]

自動復旧設定ができていないか確認する方法を記述します。

電源ボタン+CLRボタン+リセットボタンの同時押しでRAM内容がフルクリアされます。

フルクリア後、自動的にバックアップデータがリストアされれば、正常に設定ができています。

詳細は「ソフトウェアマニュアル」を参照してください。

9. 端末の安定化

9.1 リセット運用

Windows Embedded Compact 7 は、長い間レジューム運用しているとメモリーークを起こしてしまう可能性があります。

システムを安定してご使用いただくために、ユーザアプリケーションに一日一回ユーザリセットの処理を入れていただくことを推奨します。

ユーザアプリケーションにユーザリセット処理を組み込むための **API**(システムライブラリ)を提供しています。

[使用例]

ユーザアプリケーションに「業務完了」といったメニューを用意し、そこから本 **API** を呼び出してください。

9.2 サービスパックとパッチファイル

機能アップや不具合対応を行うサービスパックやパッチファイルをリリースしています。

システムを安定運用するために、これらの適用をお勧めしています。

9.3 メモリ

メモリエリアに十分な余裕がないとシステムが動作しなくなる可能性があります。少なくとも、データ記憶用エリアの残容量に **1Mbyte** 以上を確保してください。

メモリの残容量を、十分配慮したアプリケーションを作成されることをお勧め致します。

[事例]

現象: 動作が鈍くなった。

原因: アプリケーションのログを制限なく貯め続け、データ記憶容量に十分な空きがなくなっていた。

10.テクニカルインフォメーション・オンラインサポートサービス

本商品をご購入のお客様は、テクニカルインフォメーション・オンラインサポートサービスを3ヶ月間ご利用いただけます。本サービスはお客様の開発における疑問や問題点をWEB上でお問い合わせいただくことにより、迅速に解決することを目的としたサービスです。

ご利用には、ユーザー登録が必要です。お手数ですが、以下の手順でご登録ください。

1. 下記 URL にアクセスします。

<https://techinfo.casio.jp/support/>

次の画面が表示されます。

Technical Information **Online Support**

カシオでは開発/運用担当者の皆様が円滑にシステム構築・運用を行うことができるようオンラインでのサポートを行っています。

ログイン
ログインID:
パスワード:
ログイン

会員登録
新規登録
カシオ電機株式会社

What's New

- 2014/12/27 **最新情報** 企業向け販売開始
- 2014/12/24 **最新情報** セキュリティ対策のお知らせ
海運運送のパスワード印刷し忘れ防止
- 2014/12/24 **最新情報** 実験教材と標準ツール販売開始のお知らせ
- 2014/12/18 **最新情報** 最新の標準品のソフトウェアダウンロード
お知らせを掲載しました。
- 2014/12/17 **最新情報** 最新の標準品のソフトウェアダウンロード
お知らせを掲載しました。
- 2014/12/17 **最新情報** 最新の標準品のソフトウェアダウンロード
お知らせを掲載しました。

製品リリース情報

オンラインサポートの対象となる製品は下記の通りです。

モバイル
DT-300 / DT-230 / DT-YALHG330 / DT-SGK330 / DT-3100 / DT-370 / DT-3700 / DT-320 / DT-10 / DT-3200 / DT-17 / DT-2800 / DT-2300 / DT-18 / DT-300

PRIME
FORMS RICE / FAX STAGE

EA
YX-100

パートナー
PRIME STAGE for School

カメラ
EX-S0KFI

ソフトウェアダウンロード

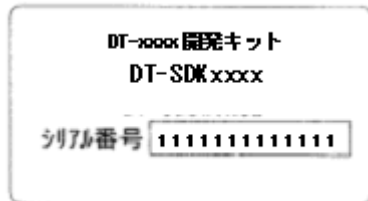
Easy-GX / DT-17ソフトウェア開発キット / DT-2300ソフトウェア開発キット(SD-E501KK0-2E) / LMPRM DT-UG98883E(SD-F200LMD) / DT-2800ソフトウェア開発キット / DT-2300ソフトウェア開発キット / DT-300ソフトウェア開発キット / DT-18ソフトウェア開発キット

techinfo.casio.jp
cybertrust
2014-02-03 16:55 JST

セキュリティについて
お客様が本サイト上で弊社へ送信される情報は、全てSSL(Secure Socket Layer)プロトコールによる暗号化を施しております。

2. [新規登録]を押します。
3. 会員規約／個人情報保護方針に同意します。
4. お客様情報を登録します。

新規登録を行うには、シリアル番号(13桁)が必要です。



シリアル番号ラベル例(xxxx は機種名、ライセンスカード / 製品 CD に貼付)

5. 「**会員規約／個人情報保護方針**」の同意ボタンを押します。

お客様情報入力画面に移りますので、以下のサンプルのように入力してください。

新規登録

必要事項をご記入の上、「内容の確認へ」ボタンをクリックしてください。
登録完了後にお客様のユーザーIDを発行いたします。

※印は必須項目です。

■登録製品 SD-E540MA0

■シリアル番号※ (半角)
※ご購入された製品に同梱されているサポート登録用のシリアルキー(13桁)をご記入ください。

お申込者情報

■パスワード※ (半角英数字4～8文字以内)
↓ 確認のためもう一度ご記入ください

■会社名※

■会社名(フリガナ)※ (全角カタカナ)

■郵便番号 〒 - (半角数字)

■住所※

■部署名※

■ご担当者※

■ご担当者(フリガナ)※ (全角カタカナ)

■役職名

6. 必要事項を入力したら、[**内容の確認へ**]ボタンを押します。
7. 入力した内容に間違いがなければ、[**登録する**]ボタンを押します。

折返し、お客様の ID を明記したメールをお届けします。
その ID と上記で指定したパスワードでログインし、サポートをお受けください。

11. 注意事項

11.1 HWND_BROADCAST について

Windows Embedded Compact 7 から HWND_BROADCAST を使用する場合、WM_USER + n が使用できなくなりました。

そのため、従来の WM_USER を使用していた通知をアプリケーションから取得する場合は、RegisterWindowMessage() を使用してシステムが割り当てる UINT を使用する必要があります。従来から変更となった通知は下記の 5 件です。

Message	Message 名
文字キー	TEXT("SysKBDChrKey")
Fn キー	TEXT("SysKBDFnKey")
クレードル着脱	TEXT("SysCradleDetect")
WakeOn	TEXT("SysWakeOn")
SD カード挿抜	TEXT("SysCardDetect")

実際の使用方法については、下記のサンプルコードを参照してください。

```
#define CHRKEY_MSG_NAME TEXT( "SysKBDChrKey" )
#define FNKEY_MSG_NAME TEXT( "SysKBDFnKey" )
#define IOBOX_MSG_NAME TEXT( "SysCradleDetect" )
#define WAKEON_MSG_NAME TEXT( "SysWakeOn" )
#define CARD_MSG_NAME TEXT( "SysCardDetect" )
UINT uMsgKbdChrKey = 0;
UINT uMsgKbdFnKey = 0;
UINT uMsgIoBoxDetect = 0;
UINT uMsgWakeOn = 0;
UINT uMsgCardDetect = 0;

BOOL CALLBACK DialogProc(HWND hDlg, UINT message, WPARAM wParam, LPARAM lParam)
{
    switch (message)
    {
        case WM_INITDIALOG:
            uMsgKbdChrKey = RegisterWindowMessage(CHRKEY_MSG_NAME);
            uMsgKbdFnKey = RegisterWindowMessage(FNKEY_MSG_NAME);
            uMsgIoBoxDetect = RegisterWindowMessage(IOBOX_MSG_NAME);
            uMsgWakeOn = RegisterWindowMessage(WAKEON_MSG_NAME);
            uMsgCardDetect = RegisterWindowMessage(CARD_MSG_NAME);
            (省略)
            return TRUE;

        default:
            // Fn キー
            if(message == uMsgKbdFnKey) {
                (省略)
            }
    }
}
```

```

// 文字キー
if(message == uMsgKbdChrKey) {
    (省略)
}

// クレードル着脱
if(message == uMsgIoBoxDetect) {
    (省略)
}

// Wake On
if(message == uMsgWakeOn) {
    (省略)
}

// SD カード挿抜
if(message == uMsgCardDetect) {
    (省略)
}
break;
}
return FALSE;
}

```

11.2 システムフォントについて

Windows の仕様により、ServicePack 等をインストールした後にシステムフォントが変更となる場合があります。アプリケーションを開発時に、システムフォントを使用せず、フォントを指定することにより、システムフォント変更による影響を受けなくなります。

カシオ計算機お問い合わせ窓口

製品に関する最新情報

- 製品サポートサイト（カシオペア・ハンディターミナル）

<http://casio.jp/support/ht/>

カシオ計算機株式会社

〒151-8543 東京都渋谷区本町 1-6-2

TEL 03-5334-4638(代)