GOT1840T Windows CE.NET4.2

User’s Manual

Disclaimers

The information in this manual has been carefully checked and is believed to be accurate. AXIOMTEK Co., Ltd. assumes no responsibility for any infringements of patents or other rights of third parties which may result from its use.

AXIOMTEK assumes no responsibility for any inaccuracies that may be contained in this document. AXIOMTEK makes no commitment to update or to keep current the information contained in this manual.

AXIOMTEK reserves the right to make improvements to this document and/or product at any time and without notice.

No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording, or otherwise, without the prior written permission of AXIOMTEK Co., Ltd.

March 2005, Version A1
Printed in Taiwan

ESD Precautions

Integrated circuits on computer boards are sensitive to static electricity. To avoid damaging chips from electrostatic discharge, observe the following precautions:

- Do not remove boards or integrated circuits from their anti-static packaging until you are ready to install them.
- Before handling a board or integrated circuit, touch an unpainted portion of the system unit chassis for a few seconds. This helps to discharge any static electricity on your body.
- Wear a wrist-grounding strap, available from most electronic component stores, when handling boards and components.

Trademarks Acknowledgments

AXIOMTEK is a trademark of AXIOMTEK Co., Ltd.

IBM is a registered trademark of International Business Machines Corporation.

MS-DOS, and Windows 95/98/NT/2000 are trademarks of Microsoft Corporation.

Award is a trademark of Award Software, Inc.

IBM, PC/AT, PS/2, VGA are trademarks of International Business Machines Corporation.

Intel and Celeron, Pentium III are trademarks of Intel Corporation.

C&T is a trademark of Chips and Technologies, Inc.

Realtek is a registered trademark of Realtek Semiconductor Corporation.

Other brand names and trademarks are the properties and registered brands of their respective owners.

This page does not contain any information.

Table of Contents

This page does not contain any information.

Chapter 1
Introduction

AXIOMTEK has accomplished an evaluation platform for Windows CE.NET solutions. The various Windows CE.NET ported SBC products now can fully support Windows CE embedded systems. By using Platform Builder to create the Windows CE image, AXIOMTEK has well developed the on-board device drivers, which makes the product to be a convenient platform for customers.

1.1 Why Windows CE

With a compact size, Windows CE is both a stable and a cost effective OS with following benefits:

- **Richest OS support for dynamic application**

  The applications include: GUI, Active X, COM/DCOM, Telephony API, Databases and .NET Compact Framework.

- **Best programming model and development tools**

- **Highest performance of built-in communication capabilities**

  Such as Network, TCP/IP, Winsock IP Multicast, IrDA, RS232/485, USB, RAS, SSL, PCT, PPP, SLIP, CSLIP, and TAPI.

- **Fastest time to market**

1.2 What improved in the platform

- **Persistent registry solution to CompactFlash disk**
- **Generic Digital Input/Output stream device driver**
- **Launching application on startup**
- **Up to 256M DRAM**

1.3 System requirement

- **Persist Storage:** 32 Mb(Minimum)
- **Memory (DRAM):** 64 Mb(Minimum)

Chapter 2
Installation

2.1 BIOS Configuration

- **Display:**
  - BIOS setting\PNP\PCI Configuration

    Assign IRQ for VGA: Disabled. VGA driver does not use IRQ.
  - BIOS setting\Advanced Chipset Features

    Frame Buffer size: 4M

    Note: need 8M frame buffer size for 1024x768x32 resolution
  - Resolution: BIOS setting\Standard CMOS feature

    The resolution is determined on the panel size.

    Panel Type: T9 800x600 TFT

    Select Display Device: CRT+LCD

- **IrDA:**
  - BIOS\Integrated Peripherals\

    UART2 Mode Select: HPSIR

    UART2 Duplex Mode: Half

    TX, RX Inverter enable: Yes, Yes

- **USB:**
  - BIOS\PNP/PCI Configuration\
    Assign IRQ for USB: Enabled
  - BIOS\Advanced Chipset Features\
    OnChip USB:Enabled
    USB Keyboard Support: Disabled

- **Resources:**
  - IRQ configuration:
    - Release IRQ for PCI device: Network, Audio, USB
    - BIOS\Advanced Chipset features\OnChip Modem:Disabled
    - BIOS\Power Management Setup\ACPI: MODEM Use IRQ:NA
    - BIOS\Power Management Setup\ACPI function:Disabled
    - BIOS\PNP/PCI Configuration\
      Resources controlled by: Manual
      IRQ-5 assigned to PCI/ISA PnP
      IRQ-9 assigned to PCI/ISA PnP
      IRQ-10 assigned to PCI/ISA PnP
      IRQ-11 assigned to PCI/ISA PnP
      IRQ-12 assigned to Legacy ISA
  - Assign IRQ for VGA:
    Disabled.
  - Nework:
    IRQ 5
  - Parallel Port:
    IRQ 7, 0x378
  - COM1:
    IRQ 4, 0x3F8
  - COM2:
    IRQ 3, 0x2F8
  - COM3:
    IRQ10, 0x3E8
  - COM4:
    IRQ11,0X2E8

- Audio:
  IRQ 9
- USB:
  IRQ 5

2.2 Setting up Boot Media and Boot up

The steps below outline how to setup the boot media with the MS-DOS using a floppy drive on the target system. The basic steps consist of formatting the boot media with MS DOS 6.22, so that the boot sector and DOS files are placed on the disk. Create an MS DOS 6.22 boot floppy disk.

1. Set CompactFlash Storage card as Primary Master device by jumper setting.
2. Boot the target system from the boot floppy that was just created.
3. Use the FORMAT.COM utility with the /S option to prepare the target boot media (Windows CE CompactFlash Storage). This will format the target boot media and place the MS DOS 6.22 system files on the target boot media.

   A:>format c: /S where c: is your target drive.
4. Copy the files(located in "\UTIL" directory) to target boot media:
5. Copy the files (located in "\platXXXX") to the target boot media.
6. Remove any floppy disk from the target floppy drive and boot the target system.

Note:

Booting up time:

It will take about 15 seconds for first time booting up in that Windows CE builds hive-based registry

2.3 Changing Display resolution

This image supports 640x480(default), 800x600 and 1024x768 display resolution.

Changing display resolution function procedures:

- Re-power on to DOS mode
- Modify autoexec.bat:

  loadcepc /L:640x480x16 or

  loadcepc /L:800x600x16 or

  loadcepc /L:1024x768x16

  Where 640x480 is display resolution, 16 is color depth which could be 8 or 16, and loadcepc.exe is boot loader.
- Restart device to Windows CE by software reset.

Changing display device function procedures:

- Click "Start\Programs\Display" menu item to change registry

  Select CRT, LCD, or TV.
- Save registry by clicking "Start\Programs\Save Registry" menu item
- Re-power on to let previous configuration take effect

Note:

1. *The resolution in combo box is ignored. Instead the resolution parameter is passed in by bootloader(ie. loadcepc /L:640x480x16)*
2. *Supported LCD types:*

   .Sanyo 12.1" SVGA LM-JK63-22NTR

   .Unipac 8" SGVA ub084s01-1

Chapter 3

Persistent Registry Entries

Windows CE supports a persistent, external registry so it can be located on any supported file system. The registry architecture is broken into two parts, a boot registry and a main registry. The boot registry loads at boot time and is responsible for loading the minimum number of drivers and applications to support loading the file-based (hive-based) main registry. The boot registry signals an event when it is completed, which triggers the main registry to load.

Procedures to save registries to CompactFlash disk(or HDD):

- Creating file-based registry

  The file-based registry, a hidden file system.hv and user.hv, is created on first time system booting up

  Thus, it will take a few seconds to boot up first time.

  Note that system.hv is located at directory "root\Settings" while user.hv is located at directory "root\Settings\Default"

- Saving registry to disk

  Click "Start\Programs\Save Registry and Shut Down" menu item, then registrys are saved

- Deleting registry from disk

  If you need to boot up with default registry in Windows CE image nk.bin, delete the file-based registry by execute delreg.bat located at directory "\util" in accompanying CD-ROM

Note that the device boots but occasionally registry data is lost when the device is powered off. This can happen if changes are not flushed out to the persistent file. So it had better save registry before you shut down the device: Click "Start\Programs\Save Registry and Shut Down" menu item, then power off

Chapter 4
Platform Features

4.1 Device Driver Supported

The supported device drivers include following:

- Persistent Registry solution for x86 platform
- Audio device driver
- Digital IO stream device driver
- IDE CompactFlash Storage support
- PS2 keyboard/mouse
- USB port: 2 port supported
- Display device driver: CRT & LCD
- RTL8139 network device driver
- IrDA (IrComm) device driver
- Serial port: COM1, COM2, COM3, COM4
- Penmount touch screen device driver (COM3)

4.2 Windows CE features

This platform is built on partial of WebPad configuration of Windows CE .NET. We exclude features of Windows Media Audio/MP3, Windows Media Video/MPEG-4 video.

The following table shows the configuration's features.

Functional area	Features
Kernel/OAL	Memory, process Compression support
CoreDLL	Full NLS-capable APIs
	Local heap and memory allocation
	Messaging, user input, windowing, and GDI support
	Communications support for serial communications and TAPI
	WAV API support
	Stdio support
	C runtime support
	FileInfo and FileOpen common dialogs
	Shell API support
	FormatMessage API support and Message resources
	Keyboard accelerator support
	TAPI API support
File system	RAM, ROM file system support
	External file system
	Hive-based registry
	FAT file system
Device Manager	Generic device driver support

GWES	Messaging, and user input support including support for standard window controls, such as buttons, edit controls, and scroll bars
	Timer message support
	Clipboard support
	Power management
	Notification LED support
	GDI support, including TrueType, text drawing, palette, and printing support
	Customizable Touch Screen Calibration UI Tchui
	Network UI dialog boxes
	WAV API and PCM manager
	IMM support Immthunk
	Window and dialog box management
	Customizable Startup, Out of Memory,
	Icons and Cursors
	User notifications API support
	Windows CE common controls and common dialog boxes
Security	Cryptography API version 1.0 support components
	Cryptography API version 2.0 support
	Authentication Service(SSPI)
Core OS Service	Debugging tool help: Toolhelp API
	Kernel feature: Fiber API

Communications service and networking:

Local Area Network(LAN):

- Wired LAN.

Personal Area Network(PAN):

- IrDA

Wide Area Network(WAN):

- Dial up networking(RAS/PPP).
- Telephony API(TAPI2.0).
- Virtual Private Networking(PPTP).

Networking Features:

- Extensible Authentication Protocol.
- Network utilities(loConfig,Ping,Route).
- Windows networking API/Redirector(SMB/CIFS).
- WinSock2.0.

Hardware support

Display

Keyboard and mouse

Battery

USB HID driver Usbhid, Usbmouse

CardTest, sample PC Card driver

ATA disk

SRAM disk

Serial,

Parallel

Audio

Platform Manager	Basic engine
	Transport: ActiveSync, KITL, TCP/IP
Internet Client service	
	Internet Explorer (IE5.5) component
	Scripting (Jscript5.5)
Windows CE Shell Components	Command Processor Cmd
	Control Panel applications
	Sample Shell
	Handheld PC style shell
Applications	
	Microsoft® Internet Explorer 5.5 Internet browser
	Microsoft® WordPad
	Inbox
	Save registry utility
	Launch application on startup utility
Application&Service development	
	Active Template Library (ATL)
	ActiveSync
	C libraries and Runtimes.
	Lightweight Directory Access Protocol (LDAP)
	Microsoft Foundation Class (MFC).
	Simple Object Access Protocol (SOAP).
	XML.

Component Service:DCOM

.NET Compact Framwork

.NET Compact Framework 1.0:

.SQL server 2000 .NET data provider

.SQL server CE2.0 .NET data provider.

.OS dependency for .NET Compact Framework 1.0

.Smart device authentication utility.

Multimedia Technology

Waveform audio.

Chapter 5
Application Note

5.1 SDK

1. There is a self-extracted SDK files, located at directory "\SDK\", for developping application VC++ . The embedded visual tools should be installed before this SDK.
2. Set Active Platform as "Wp84600" before you build application program using embedded tools. For example, setting in VC as:

   Build\Set Active Platform

   select Wp84600
3. Application and Service development support:

   .C libraries & runtimes

   .Microsoft Foundation Class(MFC).

   .Active Template Library(ATL).

   .Lightweight Directory Access Protocol(LDAP)Client.

   .Message Queuing(MSMQ).

   .Simple Object Access Protocol(SOAP) tool kit.

   .XML

5.2 Samples

Sample of digital Input/Output reading and writing:

This is a stream device driver. You can call the device via File OPEN, READ, WRITE and DEVICEIOCONTROL functions. See embedded visual tools(VB,VC) for more detailed information about File I/O functions. Sample program located in:

\Samples\TestDio\
\dNetVC
\dNetVB

1. Device Name: "DIO1:"

   Example in VC :

```
hFile = CreateFile( L"DIO1:", GENERIC_WRITE |
                    GENERIC_READ,
                    FILE_SHARE_WRITE |
                    FILE_SHARE_READ, NULL,
                    OPEN_EXISTING,
                    FILE_ATTRIBUTE_NORMAL,
                    NULL );
```

2. Generic Function: DeviceIoControl(....)

   This function offer Digital I/O read and write via the following I/O ranges:

   120 -- 123(hex.)

   200 – 21f(hex)

   250 -- 25f(hex.)

   300 – 33f(hex)

   3f0 -- 3ff(hex.)

- Subfunction definition

```
#define IOCTL_AXDIO_CFG    0    //Reserved
#define IOCTL_AXDIO_READ   1    //Read data from
                                specific IO port
#define IOCTL_AXDIO_WRITE 2     //Write data to
                                specific IO port
```

- Digital Output example in VC:

```
char bData:            // Digital Output data
wIoPort = 0x123;       // Digital Output port
DeviceIoControl(hFile, (DWORD)
                    IOCTL_AXDIO_WRITE, (LPVOID) &
                    wIoPort, sizeof(WORD), (LPVOID) &
                    bData, sizeof(BYTE), 0, 0);
```

- Digital Input example in VC:

```
   char bData:            // Digital Input data
   wIoPort = 0x123;       // Digital Input port
DeviceIoControl(hFile, (DWORD) IOCTL_AXDIO_READ,
                 (LPVOID) & wIoPort, sizeof(WORD),
                 (LPVOID) & bData,
                 sizeof(BYTE), 0, 0);
```

- Digital Input/Output example in Visual Studio .NET:
  For easily accessing digital input/output, we add GenDio.dll which call digital IO stream device driver and exports two functions AxOutp() and AxInp().

  - Digital I/O access in C#:

```
using System.Runtime.InteropServices;

[DllImport("GenDio.dll")]
public static extern int AxOutp (int
              iIoport, int iData);
```

```
[DllImport("GenDio.dll")]
public static extern int AxInp (int
            iIoport);

int iIoBase, iData;

AxOutp(iIoBase, iData);
iData = AxInp(iIoBase);
```

- Digital I/O access in Visual Basic .NET:

```
Imports System.Runtime.InteropServices
Declare Function AxOutp Lib "GenDio.dll"
            (ByVal iIo As Integer,
            ByVal iData As Integer) As
            Integer
Declare Function AxInp Lib "GenDio.dll"
            (ByVal iIo As Integer) As
            Integer

Dim iIoBase As Integer
Dim iData As Integer

AxOutp(iIoBase, iData)
iData = AxInp(iIoBase)
```

5.3 ActiveSync via RS232

ActiveSync operation:

Device site:

1.Start\Programs\Communication\Remote networking

2.Make new connection-->Direct connection-->Serial cable on COM#-->Finish

Where #=1,or 2,or 3,or 4

3.Start\Setting\Control Panel\Communication

4.Choose PC connection tab and click on Change connection

.Select the connect you just created in step 2.

Host site:

1.Open ActiveSync

2. \File\Connection setting then

check "Allow serial cable connection to this COM port" and select COM#:

Where #=1,or 2,or 3,or 4

Execute utility:

3.Device: double click on "\Windows\repllog.exe" icon

4.Host: Continuing from step 7

.click "Get connection"

Wiring: null MODEM cable

The following table shows the pin settings for a 9-pin null modem cable.

Data Terminal Equipment (DTE) Data Communications Equipment (DCE)

DTE	dce	Label
1,6	4	CD, DSR – DTR
2	3	RD – TD
3	2	TD – RD
4	1,6	DTR – CD, DSR
5	5	Gnd – Gnd
7	8	RTS – CTS
8	7	CTS - RTS

Note 9 is NC on both ends.

The following table shows the pin settings for a 25-pin null modem cable.

Note that RTS and CTS are crossed to support hardware flow.

DTE	dce	Label
2	3	TD – RD
3	2	RD – TD
4	5	RTS - CTS
5	4	CTS – RTS
6,8	20	DSR, DCD – DTR
7	7	Gnd – Gnd
20	6,8	DTR – DSR, DCD

5.4 Use embedded VC over Ethernet with Manual Server

Platform Manager supports *application connectivity*. Application connectivity is a communications framework that allows you to establish a connection between an application running on a development workstation and a target device. Remote Tools and eMbedded Visual C++ use the application connectivity support in Platform Manager to connect to a target device..

In the IDE, a *device* represents the information that is required by Platform Manager to establish application connectivity with a platform on a target device. With an application on a development workstation, you can only successfully connect to a target device if you correctly configure a device for the platform. A device includes the selection of a *startup server* and a *transport*. A transport allows the target device and desktop-side part of Platform Manager to communicate over the communications hardware. A startup server initiates the connection by downloading the device-side part of Platform Manager to the target device and starting the device-side part of Platform Manager on the target device.

Platform Builder associates transports and startup servers with a platform when it creates a software development kit (SDK) for that platform. When you install an SDK, the SDK registers information about the platform with Platform Manager. Microsoft eMbedded Visual C++ uses the information from the SDK to determine how to connect to a target device for that platform.

The manual startup server, unlike the ActiveSync Startup Server or theCESH Startup Server, does not copy files to or launch files on a target device. Choose the manual startup server if you typically build the device-side part of Platform Manager into your OS image or if you want to copy filesmanually to a target device outside of Platform Manager.

The manual startup server displays a dialog box that lists the files that you must manually copy to the target device before you can establish a connection to the target device with Platform Manager. The dialog box also displays a command that runs the files on the target device. You must copy the listed files to the target device, unless the files are already in the OS image, and type the displayed command at a command prompt to launch the files. You must launch the files before you can connect to the target device with Platform Manager.

- Manual server operation over Ethernet

1. Device site:
   1. Start Windows CE
   2. setting IP address:
      \Start\Setting\Control Panel\Network
      RTL8139 adapter
   3. Start\Programs\Command Prompt

```
\>cd Windows
\Windows>CEMGRC /S /T:TCPIPC.DLL /Q
      /D:192.168.10.58:5000
Where /D:<IP addressI><Port number>
           Port
```

Note: you should change IP string according to your IP address.

2. Desktop site:
   1. run Device SDK (self-extracted file)

   2.run eVC

      \Tools\Configure Platform Manager
      device dialog
      select MyWeb1
      MyWeb1(Default Device)
      click "Properties" button
      --------->device properties dialog
      select "TCP/IP transport for Windows CE"
      click "Configure" --------->TCP/IP transport Configuration dialog
      x Fix port Port Number:5000
      click "Advance"   ----------->device properties dialog select "Manual Server"
      click "Test" ------------>Manual Server Action dialog

Launch cemgrc with the following cmd line:

```
"/S /T:TCPIPC.DLL /D:192.168.10.58:5000"
------------>Testing Device Connection dialog
                 Connecttion to device established
                 TCP/IP transport for Windows CE
```

3. Now you can download/debug you application via Ethernet.

5.5 Access Drive on host PC over network

This is alternative way to copy your application to target device over network.First you must comfirm the network is ok: Key in your IP address on the network setting, then click "IE explorer" icon on desktop to see whether the network work or not. The procedure is as follows

Host site:

1. Sharing Drive: Right click on share drive and check "share this folder" and key in "sahre name" ie, MyC
2. Assume your computer name: MyHostPC

Device Site:

1. Start Windows CE
2. Setting IP address:

   \Start\Setting\Control Panel\Network

   RTL8139 adapter

3.Double click "My Computer" icon on desktop.

4.Key in Address: \\MyHostPC\Myc

5.Logon to network server dialog appears.

6.Key in host site: User Name: myname, Password: mypassword, Domain: MyDomain

Note that you should replace myname, mypassword and mydomain with yours.

Now you can access shared drive on your host PC

5.6 Launching application on startup

In this image we add launching application on startup utility. Note that it will wait for CompactFlash disk readiness about 3 seconds.

Usage:

1. Make "Startup" directory on Compact Flash disk
2. Copy execution file (.exe) to "Startup" directory just creaded.
3. Start (or restart) Windows CE

5.7 Touch Screen

Penmount touch screen connect with COM3. You can calibrate it as following steps:

Calibration:

.Click “Calibration” on desktop to calibrate touch screen.

.In the start\programs menu, select “save registry”, thus Calilbration data will be saved and effective in next boot.

5.8 Connecting to Visual Studio.NET 2003 over Ethernet

.NET Compact Framework is default built in this image. It is allowed to connect this device with Visual Studio .NET 2003 over Ethernet. The procedure is as follows:

1. Host PC environment :
   .Windows 2000 professional or Windows XP
   .ActiveSync 3.7
   .Visual Studio .NET 2003.
   .Windows CE utilities for Visual Studio .NET 2003.

2. Install Windows CE Utilities for Visaul Studio .NET 2003 which can be downloaded from Microsoft Web site.

3. Before deploying managed application to device. You should connect device with host PC with ActiveSync. See section 5.3 for detailed information.

4. In Visual Sdudio .NET 2003:
   (A)Tools\Options\Device Tools\Device:
      .Show device for PC: Windows CE
      .Device Name: Windows CE device
      .Transport: TCP Transport
         .Configure with "Obtain IP address automatically using ActiveSync.
      .Startup Server: ActiveSync startup provider
   (B)Tools\Select Windows CE device CPU
      .In Select Device CPU architecture, select desired CPU
      .Click on Refresh button.

5. Then you can deploy managed program to Windows CE device over Ethernet.