Seedsware

EM Series Gesture Model

Windows® Embedded Compact 7

Panel Computer

EMG7

7" Wide: EMG7-W207A8-0024-x11-01

Software Manual

Introduction

Thank you for choosing Seedsware product, the "EMG7".

The EMG7 is a panel computer with touch screen features, equipped with Windows Embedded Compact 7 (also referred to as Windows CE), an operating system for embedded system device by Microsoft, and "iMX535" CPU, by NXP, built-in for hardware.

Because this device has very different specifications from other device such as general Pocket PCs and PDAs embedded with the same OS, it is important that you understand well the features of the EMG7 and use it correctly.

Trademarks

All company names and product names noted herein are tradenames and trademarks (including registered trademarks) of each respected company. Individual displays of their rights are omitted in the descriptions of our product.

Tradenames	Owner
Microsoft, Windows, Visual C++, Visual Studio, Visual C#, Visual Basic, Windows Mobile Device Center, MSDN, Win32	Microsoft U.S.A
Adobe	Adobe Systems Software Ltd.
NXP	NXP Semiconductors

Note: Below names will appear differently than the formal trade names and trademarks as those listed above.

In this manual	Formal Names
Windows Vista	Microsoft® Windows Vista®
Windows7	Microsoft® Windows® 7
Windows8	Microsoft® Windows® 8
Windows8.1	Microsoft® Windows® 8.1
Visual Studio 2008	Microsoft® Visual Studio® 2008
MSDN	MSDN®
Win32	Win32®
Adobe Reader	Adobe® Reader®

Table of Contents

12A4A5-00051E-1

1. Software Specifications

1) Specification Summary

Features of the EMG7

(1) All data stored in the memory and RAM Disc and not inside the SD card will be destroyed after power is turned off.

(2) Applications that write constantly into the SD card will not only interfere with the functions of the program, but also shorten the life of the SD cards with write limitations or damage the SD cards if the power is suddenly shut down while writing. Please use the RAM Disc for writing files temporary.

(3) Multi-touch compliant with projective capacitive touch screen.

2) Windows Embedded Compact 7 Built-in Module

Below is the module of Windows Embedded Compact 7 (by Microsoft) built into the EMG7.

Applications- End User	ActiveSync®
	CAB File Installer/Uninstaller
	Help
Applications and Service Developments	SQL Compact
	C Libraries and Runtimes
	Lightweight Directory Access Protocol (LDAP) Client
	SOAP Toolkit
	XML
	Active Template Library (ATL)
	Object Exchange Protocol (OBEX)
	Message Queuing (MSMQ)
	String Safe Utility Functions
	Component Services (COM and DCOM)
Core OS Services	Battery Driver
	Debugging Tools
	Device Manager
	Device Support
	Internet Appliance(IABASE) Support
	Kernel Functionality
	Notification LED Support
	Power Management
	Serial Port Support
	Time Zone and DST Service
	UI Proxy for Kernel-Mode Drivers
	Windows Embedded Compact Driver Development Kit Support Library
Device Drivers	Display
	Input Devices
	Networking
	SD
	Storage Devices
	USB

Fonts	Scripts
	Symbol
	UI
	Web Fonts
	Wingdings
International	Input Method Manager (IMM)
	Locale Services
	National Language Support (NLS)
Internet Client Services	Internet Explorer 7.0 for Windows® Embedded Compact7
	Internet Options Control Panel
	Internet Options Control Panel Silverlight for Windows Embedded
	Scripting (JScript, VBScript)
	Browser Application
Graphics and Multimedia Technologies	Graphics (BMP, GIF, JPG, PNG)
	Audio (MP3, WMA)
	Media (MPEG1, MPEG2, MPEG4, VC-1, WMV7~9, ASF, AVI) *Screen image may flicker when the bit rate of a video exceeds 6Mbps
Shell and User Interface	Graphics, Windowing and Events
	Shell Graphical Shell (Standard Shell) AYGShell API Set Command Shell Screen configuration
	User Interface Control Panel Applets Software Input Panel Network User Interface Mouse Common Dialog Support
	Silverlight for Windows Embedded
Security	Authentication Services (SSPI)
	Local Authentication Sub-System
	Cryptography API: Next Generation (CNG) Primitives
	Cryptography Services (CryptoAPI 1.0) with High Encryption Provider
	Microsoft® Certificate Enrollment Tool Sample Credential Manager
File Systems and Data Store	Compression
	Database Support
	File and Database Replication (Bit-based)
	File System – Internal (ROM-only File System)
	Registry Storage (Hive-based Registry)
	Storage Manager
	System Password

	Simple Network Time Protocol (SNTP)
	Servers
	Core Server Support
	Networking - Local Area Network (LAN)
	Wired Local Area Network (802.3, 802.5)
	Networking - Wide Area Network (WAN)
	Telephony API (TAPI 2.0)
	Point-to-Point Protocol over Ethernet (PPPoE)
Communications Service	Dial Up Networking (RAS/PPP)
and Networking	Virtual Private Networking (PPTP)
	Networking General
	NDIS User-mode I/O Driver
	TCP/IP
	Windows Networking API/Redirector (SMB/CIFS)
	Winsock Support
	Network Driver Architecture (NDIS)
	Network Utilities (IpConfig, Ping, Route)
	Extensible Authentication Protocol
Application and Service Development	.NET Compact Framework 3.5

*Above chart is in accordance with the contents of Windows® Embedded Compact7 Run-time license.

12A4A5-00051E-1

3) EMG7 Original Functions and Driver Specifications

The built-in driver can be called out by the below 5 modules.

Please refer to Microsoft MSDN for the specification details of each module.

Some arguments will have a use of its own.

Be sure to include a header file, ProIS_SDK.h.

#include <ProIS_SDK.h>

For instructions concerning its unique use, set the argument as instructed in this specification.

Please refer to examples for basic setting of the arguments.

*By changing the driver name of CreateFile and handle name of CloseHandle, it can be applied to all drivers.

3-1. Module

Module Name: CreateFile, CloseHandle

Call Format:
 	
        HANDLE CreateFile(
                LPCTSTR *lpFileName*,
                DWORD *dwDesiredAccess*,
                DWORD *dwShareMode*,
                LPSECURITY_ATTRIBUTES *lpSecurityAttributes*,
                DWORD *dwCreationDispostion*,
                DWORD *dwFlagsAndAttributes*,
                HANDLE *hTemplateFile*
        );

Example 1:
        CreateFile(L"DIO1:", GENERIC_READ | GENERIC_WRITE, 0, NULL,
                OPEN_EXISTING, 0, NULL) ;
 	
        BOOL CloseHandle(
                HANDLE *hObject*
        );
 	
Example 2:
        CloseHandle(hDio) ;

Module: DeviceIoControl

Call Format:
 	
 BOOL DeviceIoControl(
 	
         HANDLE *hDevice*,
 	
         DWORD *dwIoControlCode*,
 	
         LPVOID *lpInBuffer*,
 	
         DWORD *nInBufferSize*,
 	
         LPVOID *lpOutBuffer*,
 	
         DWORD *nOutBufferSize*,
 	
         LPDWORD *lpBytesReturned*,
 	
         LPOVERLAPPED *lpOverlapped*
 	
       );

Example:
 	
       DeviceIoControl(hDio, dwControlCodeReadDin,
 	
                     (LPVOID)(&bInBuf), sizeof(bInBuf),
 	
                     (LPVOID)(&bOutBuf), sizeof(bOutBuf),
 	
                     &dwReturnSize, NULL) ;

12A4A5-00051E-1

3-2. WatchDogTimer Driver

WatchDogTimer Driver is a driver that controls the WatchDog Timer.

(1) Driver Open

Handle can be obtained by opening the driver by using the CreateFile function with device name "WDT1:"

(2) Driver Close

Driver can be closed by using CloseHandle function with the handle obtained in (1).

(3) DeviceIoControl

WatchDogTimer Driver supports the listed below.

NO	Process	Description
1	IOCTL_WDT_START	Start WatchDog Timer
2	IOCTL_WDT_STOP	Stop WatchDog Timer
3	IOCTL_WDT_CLR	Clear WatchDog Timer counter with WatchDog Timer Clear Register.
4	IOCTL_WDT_CNT_WT	Set count value to Watch Dog Timer counter register. Set count value to third argument. Count value: 1~127 (seconds)
5	IOCTL_WDT_CNT_RD	Read out count value of WatchDog Timer counter register. Count value is returned to fifth argument.
6	IOCTL_WDT_STS_RD	Read WatchDog Timer counter status. Count status is returned to fifth argument. 0: counting 1: count up
7	IOCTL_WDT_COLDRST	Cold boot the CPU. When this process is called out, it will be cold reset hardware wise and will not go back to the status before the cold reset.

3-3. BacklightBuzzer Driver

BacklightBuzzer Driver is a driver to control the I/O of front area status LED, backlight, and buzzer.

(1) Driver Open

Handle can be obtained by opening the driver by using the CreateFile function with device name "BKL1:"

(2) Device Close

Driver can be closed by using CloseHandle function with the handle obtained in (1).

(3) DeviceIoControl

BacklightBuzzer Driver supports the listed below.

NO	Process	Description
1	IOCTL_BACKLIGHT_STATUS	Get backlight status Backlight status is returned to fifth argument. 　0: Backlight OFF 　1: Backlight ON
2	IOCTL_BACKLIGHT_ON	Turn backlight ON
3	IOCTL_BACKLIGHT_OFF	Turn backlight OFF
4	IOCTL_BUZZER_STATUS	Get buzzer status. Buzzer status is returned to fifth argument. 　0: Buzzer OFF 　1: Buzzer ON
5	IOCTL_BUZZER_ON	Turn buzzer ON
6	IOCTL_BUZZER_OFF	Turn buzzer OFF
7	IOCTL_BUZZER_HIGHLOW	Set buzzer sound. 　0: Low pitch 　1: High pitch
8	IOCTL_LED_STATUS	Obtain front area status LED. Front status LED status is returned to fifth argument 　0: Off 　1: Green 　2: Red 　3: Orange

NO	Process	Description
9	IOCTL_LED_GREEN	Light front area status LED Green.
10	IOCTL_LED_RED	Light front area status LED Red.
11	IOCTL_LED_ORANGE	Light front area status LED Orange.
12	IOCTL_LED_OFF	Turn off front area status LED.
13	IOCTL_WR_OFFTIME	Set time for backlight to turn off automatically. Set time for light to turn off automatically to third argument (in seconds). Automatic turn off is activated when time is set. Automatic turn off is deactivated when time is set to "0" Count value: 1~65535 (seconds)
14	IOCTL_RD_OFFTIME	Obtain setting time for automatic backlight turn off. Setting time is returned to fifth argument. (in seconds)

12A4A5-00051E-1

3-4. Bright Driver

Bright Driver is a driver that controls the backlight brightness of the LCD.

(1) Driver Open

Handle can be obtained by opening the driver by using the CreateFile function with device name "BAC1"

(2) Driver Close

Driver can be closed by using CloseHandle function with the handle obtained in (1).

(3) DeviceIoControl

Bright Driver supports the two processes listed below.

NO	Process	Description
1	IOCTL_BAC_RD_BRIGHT	Obtain LCD backlight brightness setting value (brightness value). Brightness setting value is returned to fifth argument.
2	IOCTL_BAC_WT_BRIGHT	Set LCD backlight brightness. Brightness setting value is returned to third argument. Brightness setting value: 1(dark)~16(bright)

3-5. RTC Driver

RTC Driver is a driver to control RTC outside of the CPU.

Due to restrictions of external RTC, possible setting range of 'year' is 2000 to 2099.

Window CE will read time data from RTC only at boot up and counts time with the timer inside of the CPU. It is necessary to obtain time from a RTC outside the CPU by using this driver in order to know the correct time.

(1) Driver Open

Handle can be obtained by opening the driver by using the CreateFile function with device name "RTC1:"

(2) Driver Close

Driver can be closed by using CloseHandle function with the handle obtained in (1).

(3) DeviceIoControl

RTC Driver supports the listed below.

NO	Process	Description
1	IOCTL_RTC_RD	Read out time from external RTC Time is returned to fifth argument by SYSTEMTIME format.

3-6. SRAM Driver

SRAM Driver is a driver to read/write the SRAM.

(1) Driver Open

Handle can be obtained by opening the driver by using the CreateFile function with device name "RAM1:"

(2) Driver Close

Driver can be closed by using CloseHandle function with the handle obtained in (1)

(3) DeviceIoControl

SRAM Driver supports the listed below.

NO	Process	Description
1	IOCTL_RAM_RD	Read data from SRAM Set SRAM readout offset and readout byte to third argument Readout data of specified byte from specified offset will be returned to fifth argument 　Offset　　　　: 0x00000000~0x0007FFFF (Double word) 　Readout byte : 0x00000000~0x00080000 (Double word)
2	IOCTL_RAM_WT	Write data to SRAM Set SRAM write offset, write byte, and write data to third argument. 　Offset　　　　: 0x00000000~0x0007FFFF (Double word) 　Write byte　 : 0x00000000~0x00080000 (Double word) 　Write data　 : Set according to write byte

2. Startup and Shutdown Procedure

1) Start-up

1-1. Start-up Procedure
Windows CE will automatically start up when power of EMG7 is turned ON.

*When restarting the EMG7, leave power off for a few seconds (about five seconds) before turning the system back on. It may not start up properly.

1-2. Displaying of Startup Screen
A fixed screen (referred to as 'Logo screen' hereinafter) can be displayed after turning on the power until the Windows CE screen is displayed. (It can be also be customized by the customer)
Whether or not to display the Logo screen can be chosen from the set-up tools.
Please refer to the set-up-tool manual for details.
Also, if the Logo is not set, an entire screen will become white and a progress bar showing the startup progress will be displayed.
The progress bar showing the startup progress is black, so colors other than black are recommended for the background color of the Logo screen.

2) Shutdown
Windows CE does not need to be shut down.
Make sure the SD card is not being accessed before turning the power off.

3. Software Development

1) Developing Environment

This chapter will describe the hardware and the software environment necessary for the application development on the EMG7.

1-1. Hardware Environment

Hardware environment diagram necessary to develop applications for the EMG7 is as shown below.

EMG7   PC for application development

USB Cable *1
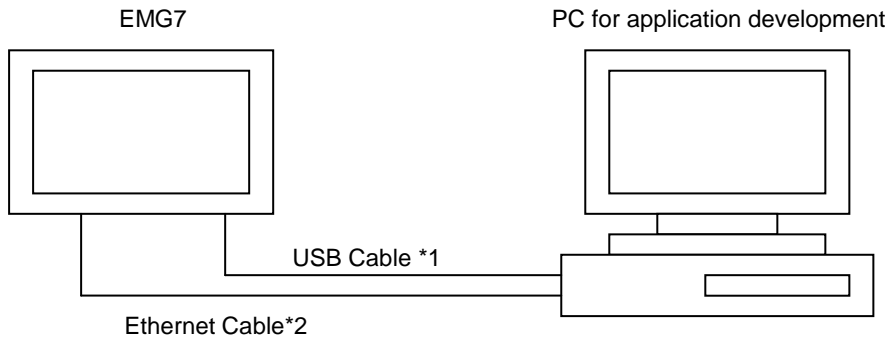
Ethernet Cable*2

> *1 USB cable used is A type-MiniB.
>
> *2 When connecting the EMG7 directly to the PC, a cross cable for Ethernet is necessary.
>
>   When connecting via hub, a normal straight cable can be used.
>
>   Please refer to each part name for connecting location.

・Please refer to the manual of each product for details of each hardware listed.

・Also, each item of the PC used for application development can be combined or used concurrently, but make sure there is enough memory size and disk space necessary when doing so.

1-2. PC Environment for Application Development

Basic Software	Microsoft Windows Vista / Windows7 / Windows8 / Windows8.1
	Recommended: Windows7
PC	Minimum: 2.0GHz CPU, 512MB RAM, 8GB Hard Drive
	Recommended : 2.6GHz CPU, 1GB RAM, 20GB Hard Drive
Communication Port	USB Host Port (USB1.1 or more)
	LAN Port (10Base-T/100Base-TX)
	USB Port development alone is possible, but concurrent use of with LAN Port is recommended.

*Disc device depends on the software memory medium for application development.

12A4A5-00051E-1

1-3. Software Environment

In order to develop applications that operate on the EMG7, below software environment is necessary. Choose the environment depending on the application development language.

Visual Studio 2008	Application can be created by C++, using Visual Studio 2008 which will allow high speed, compact application creation.* Also, by installing .Net Compact Framework, a subset of the .Net Framework, C#.Net・Basic.NET can be used to create an application. Windows application can be made more efficiently by using Net Compact Framework.
EMG7 SDK	A package of standard library for creating applications that operates on the EMG7 and a library header file for the original interface of EMG7. It is an essential package for application development of the EMG7.

*Some in Win32 API are supported by Windows CE but may have different interface specifications.
Please refer to the MSDN from Microsoft for details.

1-4. Files In SDK

SDK is the abbreviation for Software Development Kit, a kit to create and debug applications that operates on the EMG7.

SDK includes a header file library for standard interface provided by Microsoft necessary for application development of the EMG7. It also includes a header file library for the interface originally developed by Seedsware.

A folder structure as below will be installed when the SDK is installed.

C:¥Program Files¥Windows CE Tools¥SDKs¥EMG7_SDK
　¥Include
　¥Lib

*Above is an example of when installed into the C Drive.
Folder for installing will vary according to the installing condition of each customer.

12A4A5-00051E-1

2) Installing of Application Development Tool

This chapter will describe the installation of application development tool.

2-1. Installing of Application Development Tool

USB Port development alone is possible, but concurrent use of with LAN Port is recommended.

Installation procedure of the application development tool is as below:

(1) Installing of Visual Studio 2008

Install accordingly to Visual Studio 2008 procedure.

(2) Installing of SDK

Execute EMG7_SDK.msi

Install accordingly to SDK installer procedure.

(3) Installing the Service Pack

Service Pack1 needs to be installed when using Visual Studio 2008.

Please install from the Microsoft homepage.

(4) Installing the Windows Mobile Device center

Download below tool from Microsoft download page and install to the computer to be used for application development.

There are two types of OS for Windows Vista and Windows 7: 32bit, and 64bit.

Be sure to confirm the OS type and install the Windows Mobile Device center complying with the OS. (The OS bit can be confirmed from computer property.)

WindowsVista / Windows7	Windows Mobile Device Center

*We have confirmed that Windows Mobile Device center can be used properly.

It is necessary to connect the computer used for application development and the EMG7 by Windows Mobile Device Center in order to download programs or debug applications of the EMG7 from the PC used for application development.

The PC for application development and the EMG7 can only be connected by USB, but when downloading data, downloading time can be greatly reduced by using the Ethernet cable together with the USB cable. As long as the network is connected when connecting to Windows Mobile Device Center, data will download automatically via the Ethernet.

## 2-2. Connection Procedure to Windows Mobile Device Center

The connecting procedures for Windows Vista, Windows7 is as described below, assuming Windows Mobile Device center is installed accordingly to each OS environment.
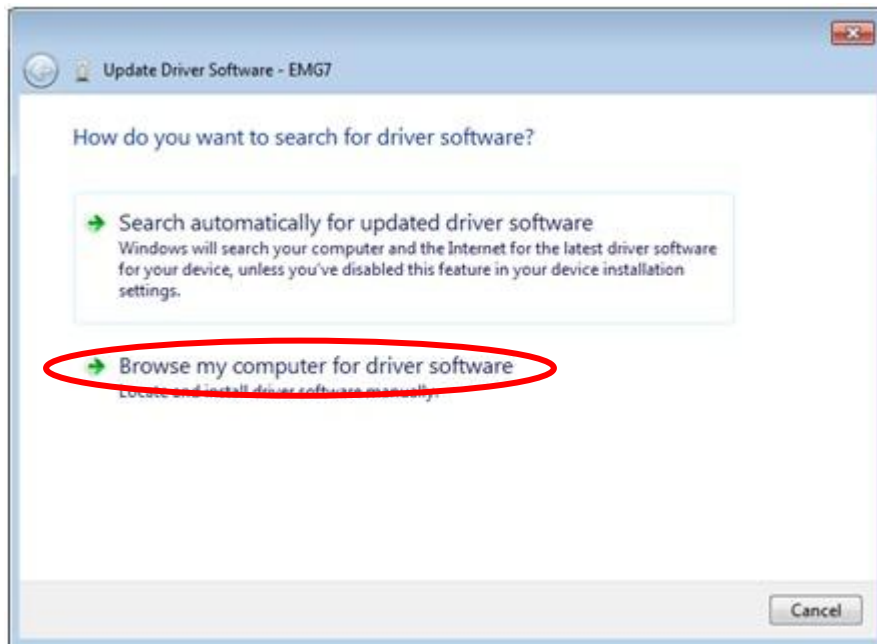
2-2-1. For Windows Vista, Windows7

1. Start the device manager from control panel.

2. Connect the PC and the equipment with an USB.
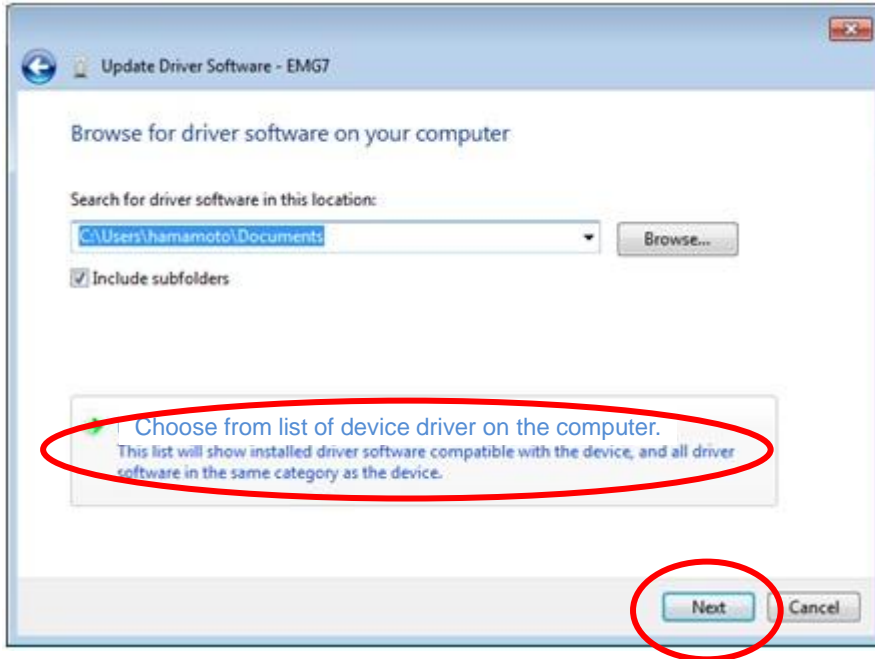   When connected, click *Other Device->EMG7*.

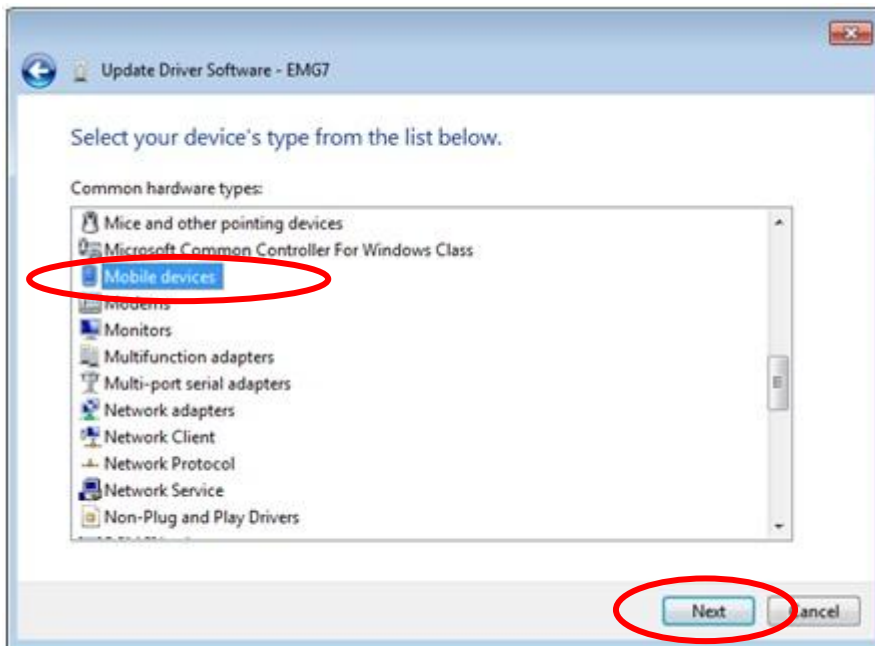12A4A5-00051E-1
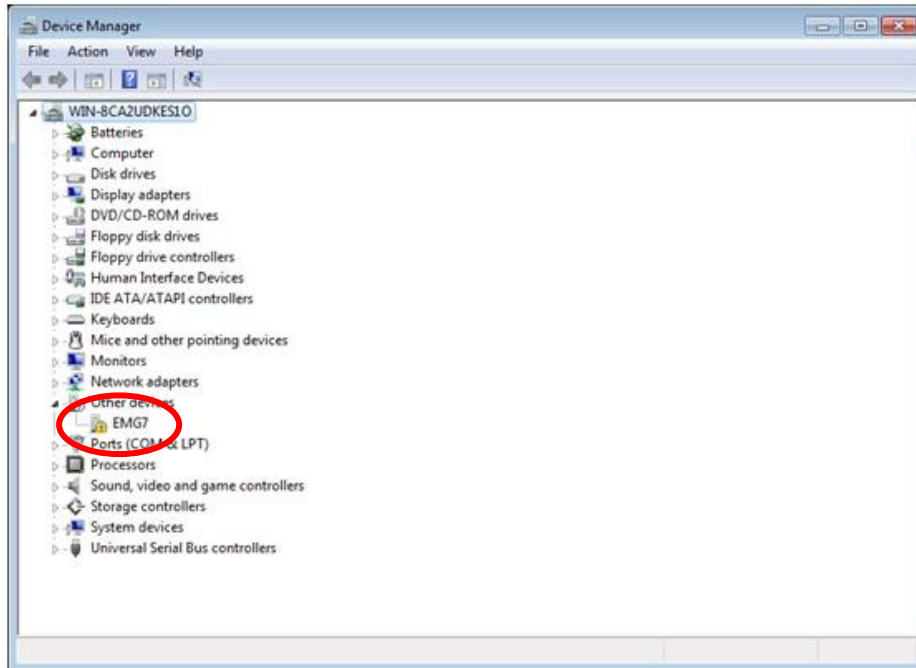
3. Select and execute *Update Driver Software Update* of *EMG7.*
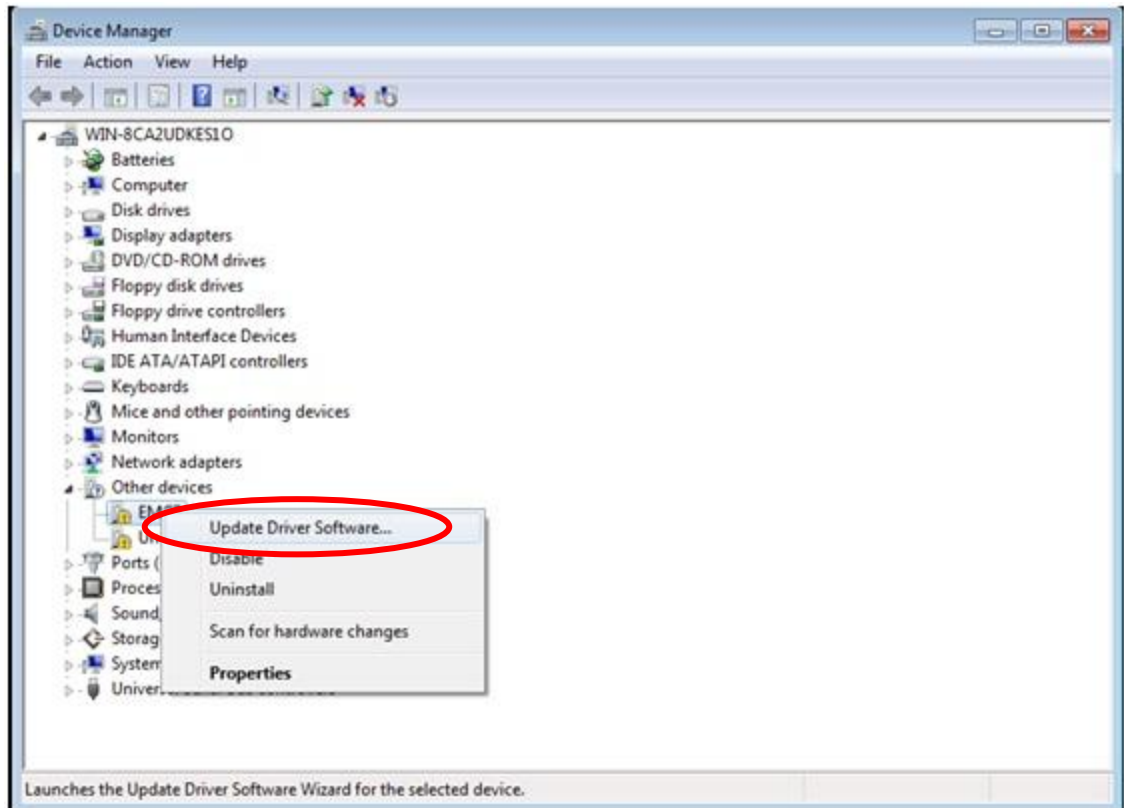
4. Below window will appear. Select and execute *Browse my computer for driver software.*

12A4A5-00051E-1

5. Below window will appear. Select *Choose from list of device driver on the computer.* and click *Next.*

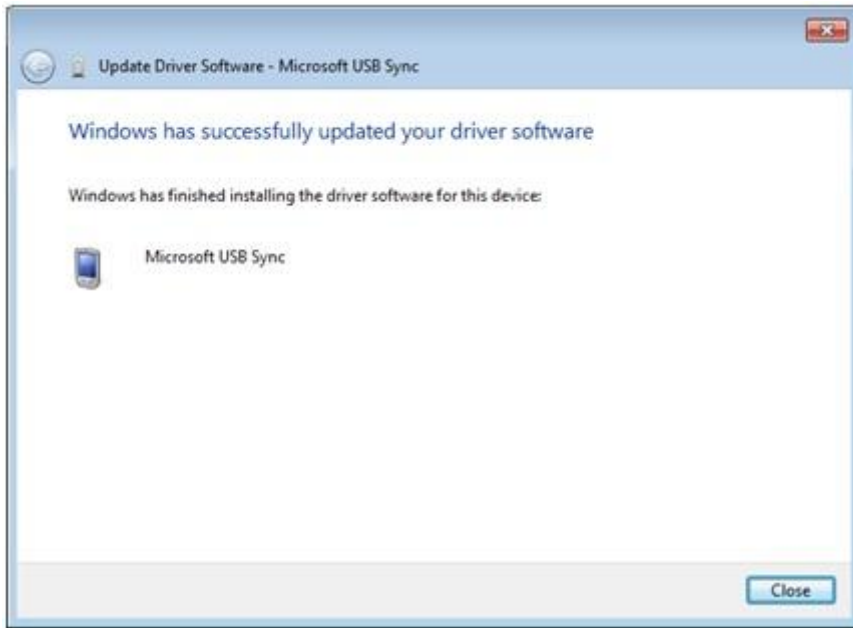6. Below window will appear. Select *Mobile devices* and click *Next.*

12A4A5-00051E-1

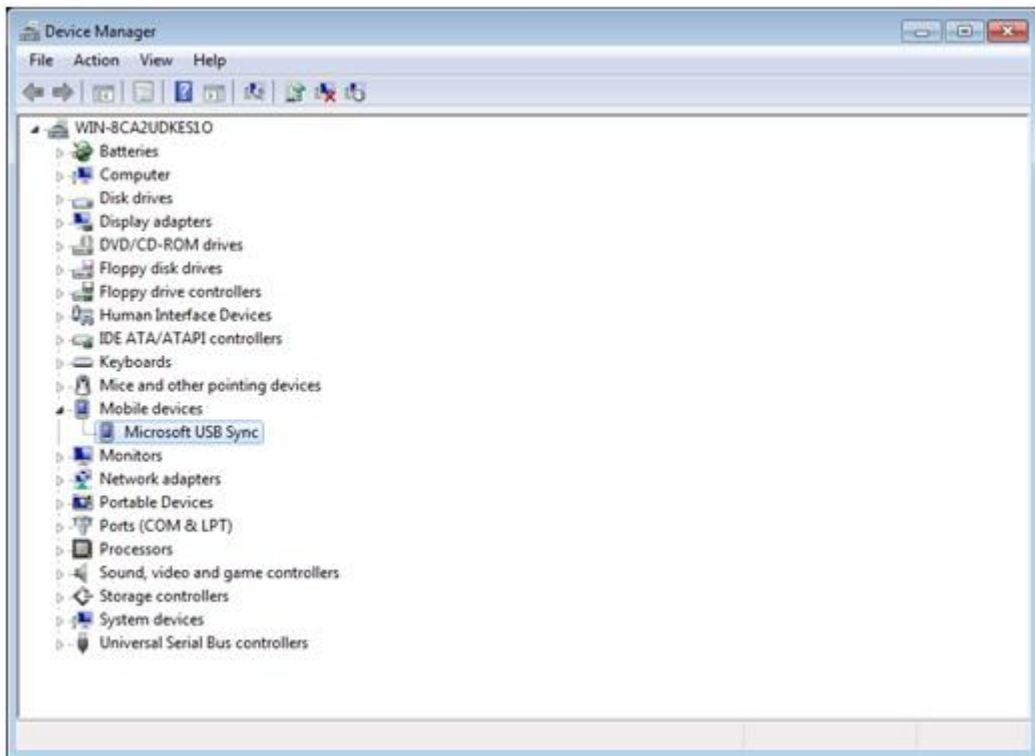7. Below window will appear. Choose *Microsoft* from the manufacturer column and *Microsoft USB Sync* from Model and click *Next*.
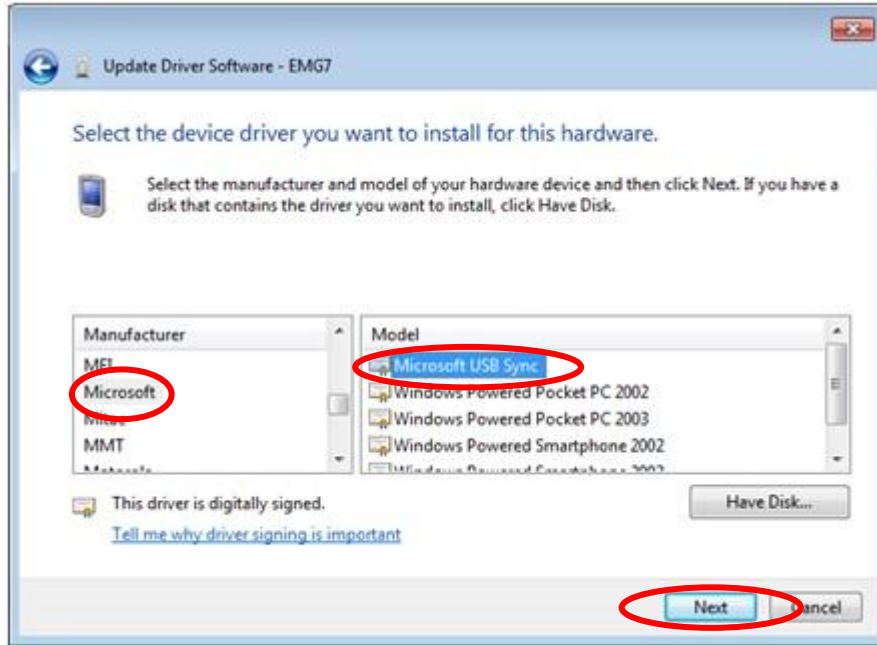
8. Below window will appear. Choose *Yes* and proceed.
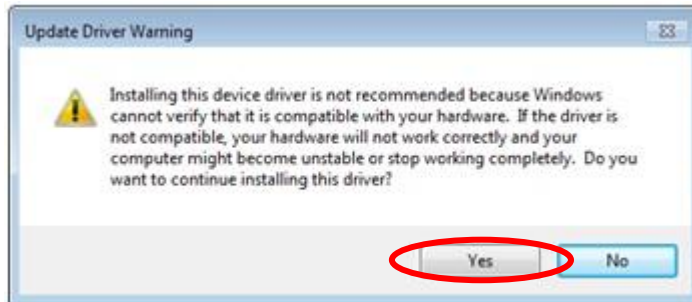
9. Driver installation will begin. Below window will appear when installation is complete.

10. Device Manager will appear as below after installation is complete.

12A4A5-00051E-1

11. Windows Mobile Device Center will connect automatically and below screen will appear.

This procedure does not have to be repeated after successfully connecting the first time.
The Driver will automatically be recognized and connect to the Windows Mobile Device Center.

12A4A5-00051E-1