User's Manual

EB-850/JG2+TFT

TESSERA TECHNOLOGY INC.

Date published: June 2008

V 1.01

Windows and Windows XP are registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.

- The information is subject to change without notice.
- No part of this document may be copied or reproduced in any form or by any means without prior written consent of TESSERA TECHNOLOGY INC.
- TESSERA TECHNOLOGY INC. does not license assurance or enforcement of intellectual property rights and other rights of TESSERA TECHNOLOGY INC. and third parties by the use of the products and/or the information in this document. TESSERA TECHNOLOGY INC. does not assume any liability for infringement of rights of third parties by the use of the products and/or the information in this document.
- Descriptions of circuits, software and other related information in this document are provided for illustrative purposes in semiconductor product operation and application examples. The use of these circuits, software and information in the design of a customer's equipment shall be done under the full responsibility of the customer. TESSERA TECHNOLOGY INC. assumes no responsibility for any losses incurred by customers or third parties arising from the use of these circuits, software and information.

CAUTION:

- Do not give any physical damage to this equipment such as dropping.
- Do not superimpose voltage to this equipment.
- Do not use this equipment with the temperature below 0°C or over 40°C.
- Make sure the USB cables are properly connected.
- Do not bend or stretch the USB cables.
- Keep this equipment away from water.
- Take extra care to electric shock.
- This equipment should be handled like a CMOS semiconductor device. The user must take all precautions to avoid build-up of static electricity while working with this equipment.
- Measurement tools including the workbench must be grounded when testing.
- The user/operator must be grounded using the wrist strap.
- The connectors and/or device pins should not be touched with bare hands.

TABLE OF CONTENTS

	5
CHAPTER 1 PREPARATION	7
1.1 Development Tools / Software	8
1.2 Installation of Development Tools	9
1.2.1 Installation Package	
1.2.2 Installation of Development Tools	9
1.3 Sample Programs	13
1.3.1 Preparation of Sample Programs	
1.3.2 Overview of Sample Programs	16
CHAPTER 2 EXPERIENCES	17
2.1 Start PM+	
2.2 What is PM Plus	
2.3 Load Workspace (Project)	
2.4 Check Compiler Option Settings	
2.4.1 Compiler Common Options2.4.2 Preprocessor Tab	
2.5 Create Load Module Files	
2.6 Check Debugger Settings	
2.7 Check Kit Connection	31
2.8 Start Debugger (ID850QB)	33
2.9 Run Programs	36
2.10 Stop Programs	38
2.11 Close Debugger (ID850QB)	
2.12 Quit PM+	40
CHAPTER 3 HARDWARE SPECIFICATIONS	
3.1 Layout of hardware functions	
3.2 Hardware Functions	
3.2.1 SW1	
3.2.2 SW2	-
3.2.3 SW3 (INTP0)	
3.2.4 SW4 (INTP1)	43
3.2.5 D3	
3.2.6 D4	
3.2.7 D5	
3.2.8 J1	
3.2.9 J2	43

	3.2.10	J4	44
	3.2.11	J8	44
СНАРТ	ER 4	ROUBLESHOOTING	45
4.1	If you c	annot find USB driver when you connect MINICUBE2 and PC	45
4.2	Error w	nen you start the debugger	
	4.2.1	"Can not communicate with ICE" (F0100 or A0109)	45
	4.2.2	"Incorrect ID Code." (Ff603)	46
	4.2.3	"The on-chip debug function had been disabled in the device." (F0c79)	46
	4.2.4	"Disabling the on-chip debug function is prohibited." (F0c33)	46
СНАРТ	ER 5 (OTHER INFORMATION	47
5.1	Create	a New Workspace	48
5.2	Registe	r Additional Source File	56
5.3	Debugg	er Useful Tips	58
	5.3.1	Change Display of Buttons	58
	5.3.2	Display Source List and Function List	58
	5.3.3	Set/Remove Breakpoints	59
	5.3.4	Display Global Variables	60
	5.3.5	Display Global Variables While Programs Are Running	61
	5.3.6	Display Local Variables	62
	5.3.7	Display Memory and SFR Contents	62
5.4	QB-Pro	grammer	63
5.5	Circuit I	Diagram	67

Introduction

EB-850/JG2+TFT is the evaluation kit for developing an application system using "V850ES/JG2", NEC Electronics 32bit All Flash microcontroller.

The user only needs to install the development tools and USB driver, and connect MINICUBE2 to this kit with USB, to experience the development processes such as coding, building, debugging, and monitoring the output.



Configuration for Debugging

As the sample programs are preinstalled, the user can check the TFT displays and the touch panel operations by supplying power with bundled USB cable.

Overview

This manual consists of the following contents.

Read chapter 1 and 2 first for installing the development tools and using the sample programs. Read chapter 3-6 for customizing the sample programs and the hardware.

Chapter 1: Preparations

Install the tools

Chapter 2: Experiences

Experience the basic operations of integrated development environment (PM+) and integrated debugger (ID850QB) with using sample programs.

Chapter 3: Hardware Specifications

Explain the hardware specifications of EB-850/JG2+TFT

Chapter 4: Troubleshooting

Describe how to solve troubles you may face, such as errors when starting the integrated debugger (ID850-QB)

Chapter 5: Other Information

Introduce other information, such as how to create a new workspace (project) on integrated development environment (PM+), how to register additional source files, and some useful tips of the integrated debugger.

The circuit diagrams of demonstration kit are included in this chapter.

Reader

This manual is intended for development engineers who are new to those development tools for development using V850.

It is assumed that the readers have been familiar with basics of microcontrollers, C and assembler languages, and the Windows operating system.

Purpose

This manual is intended to give users an understanding of the features, hardware configurations, development tools for V850.

Users can understand more by reading this manual and using the development tools together.

Chapter 1 Preparation

This chapter describes following topics:

- Overview of development tools
- Installation of development tools
- Preparation and overview of sample programs

Using the development tools bundled with EB-850/JG2+TFT, the user can experience the development processes such as coding, building, debugging, and monitoring the output.

1.1 Development Tools / Software

- Device file DF703724 V1.00
 A device file contains device specific information. So, users need a device file to use the development tools.
- Integrated Development Environment (IDE) PM+ V6.30
 The IDE works on Windows operation system.
 Users can develop a system efficiently by using the editor with idea processor function, compiler, and debugger.
- C Compiler Package CA850 W3.10 (code size limited version) C compiler for the V850 series. The object code size is limited to 128 Kbyte. This compiles C source program and assembler source program into executable code for V850 series.
- Integrated Debugger ID850QB V3.41

This is the tool for debugging the object program generated from C compiler and assembler. The debugger enables to do C source level debugging. With the debugger, you can debug the code easily and efficiently by referring and changing variables, using step-in debugging function, and so on.

- Built-in Flash Memory Writing Program QB-Programmer This is the Windows software to write programs on microcontroller built-in flash memory.
- Sample Programs, "Graphic display program" and "Washing machine menu program" Sample programs using graphic libraries.

1.2 Installation of Development Tools

1.2.1 Installation Package

The installation package includes the development tools and documentations. Users can use the installer to install those development tools and documentations.

1.2.2 Installation of Development Tools

 Please insert the CD-ROM in the drive. The installer will show up automatically. If it does not start automatically, please start it by double clicking the SETUP.EXE from Windows Explorer.

🙀 NEC Electronics Microcomputer Development 1	Fools Setup 🛛 🔀
Welcome to the NEC Electronics Microcomputer devel setup program.	opment tools
Before you execute this installer, please exit all window	s programs.
EB-850/JG2+TFT T1.00 (Februa Copyright (C) NEC Electronics Corp	
Beadme First	
Install	<u>,</u>
<u>D</u> ocuments	Welcome
<u>S</u> ample Program	Welcome
Link to NEC Electronics <u>M</u> icrocomputer	
E <u>x</u> it	

2) Click "Install..."

3) "Installer" dialog box shown below is opened. Select tools that you need to install. (As default, all the tools that you need to use the demonstration kit are selected.) "Explain" area displays an explanation of the selected tool. To change the installation destination, click "Browse...".

When all the settings are completed, click "Install...".

* In this document, it is assumed that users install the tools under "NEC Electronics Tools" directory (default installation directory). Users can find the tools by selecting "Start Menu" -> "Programs" -> "NEC Electronics Tools".

🛃 Tool Installer		x
DeviceFile Install		~
DeviceFile: DF703724 V850ES/Jx2 Ver1.00	Search	
Tool Install:		
Tools:		
Product	Size 🔺 🛛 Exit	
PM+ V6.30	17,460KB	
PM+ V6.30 Documents	12,636KB	
CA850 W3.10	11,444KB	
CA850 V3.10 Documents	17,612KB	
🗹 LDG V1.20	6,193KB	
LDG V1.20 Documents	1,671KB	
STK850 V2.11	4,576KB	
D850QB V850 Integrated Debugger V3.41	37,952KB 🗾	
Explain:		
	Drive: C:	
	Available Space:	
	9 ,748,336	КВ
Destination:	Required Space:	
C:\Program Files\NEC Electronics Tools	Browse 133,668	КВ

4) Click "OK", when "Install" confirmation dialog box is displayed.



5) Agree with the license agreement and click "Yes" for continuing the installation. To stop the installation, click "No".

InstallShield Wizard
License Agreement Please read the following license agreement carefully.
Press the PAGE DOWN key to see the rest of the agreement.
USER LICENSE AGREEMENT IMPORTANT-READ CAREFULLY: This User License Agreement ("ULA") is a legal agreement between you (either a natural person or an entity) and NEC Electronics Corporation ("NEC") for the SOFTWARE PRODUCT. As used herein, "SOFTWARE PRODUCT" means the NEC's computer software products provided with this ULA, which includes computer software and may include associated media, printed
Do you accept all the terms of the preceding License Agreement? If you choose No, the setup will close. To install CA850 W3.10, you must accept this agreement. InstallShield K Back Yes No
< Back Yes No

6) Enter the product ID, and click "Next".

* The product ID is available on the attached sheet and the document "README_E.HTML".

InstallShield Wizard	×
Please enter the product ID of the product.	
Enter Product ID.	
Product ID is written on the case or in the accompanying documents.	
Product ID	
InstallShield	
< <u>B</u> ack <u>N</u> ext >	Cancel

7) It starts copying the files.

8) When the installation is completed, the following dialog opens. Click "OK". Installation of the development tools is completed.



[Notes on the installation]

- Support Windows versions are Windows XP and Windows 2000
- Administrator authority is required for this installation.
- Installation directory name must use only ASCII characters (1-byte alphanumeric characters). Do not use 2-byte and /*:<>?|"¥;, characters. It may cause errors.

[Limitation]

- CA850 limits the object size to 128 Kbyte.

1.3 Sample Programs

This section explains the preparation and overview of sample programs.

1.3.1 Preparation of Sample Programs

 Please insert the CD-ROM in the drive. The installer will show up automatically. If it does not start automatically, please start it by double clicking the SETUP.EXE from Windows Explorer.



2) Click "Sample program"

🖛 Back 🔻 🔿 🛪 🔯 🚰 🛱 🥘 Search 👔 Favorites 🧐 Media 🍏 🔂 🖬 🎒		
Address 🖉 E:\DOC\SAMPLE_E.HTML	• (⇒Go
		_
Sample Programs		
Welcome to EB-850/JG2+TFT world.		
The purpose of sample programs is to experience EB-850/JG2+TFT.		
The pulpose of sample programs is to experience ED-050/052 (IF I.		
Sample programs consist of the following items.		
• TK850		
• 1K850 • V850TFT		
 graphic sample : Graphics display sample program. 		
 washing_machine : Washing machine menu sample program. 		
Refer to the user's manual about usage.		
EB-850/JG2+TFT sample program		
<u>EB 85005G2 (TTT user's manual</u>		
EZ-850/JG2+TFT Graphics library user's manual		
<u>B-850/JG2+TFT Graphics library application note</u>		
B) Done	🔠 Local intra	net
al Anue	Cocal Intra	net

3) Click "EB-850/JG2+TFT sample program"

		File Dow	nload	<u><</u>
		?	Some files can harm your computer. If the file information below looks suspicious, or you do not fully trust the source, do not open or save this file.	
			File name: TK850.exe	
			File type: Application	
			From: E:\SAMPLE	
			This type of file could harm your computer if it contains malicious code.	
			Would you like to open the file or save it to your computer?	
			Open Save Cancel More Info	
			Always as Prote opening this type of file	
				_
4)	Click "Save"			

						<u>? ×</u>
Savei	in: 🥃 Local Disk	(C:)	•	🗢 🗈 💣		
History History Desktop My Documents My Computer		-				
My Network P	File name: Save as type:	TK850 Application		•	Save	

6) The self-extraction sample program set (TK850.exe) is copied to the specified directory. When this file is executed, "TK850" directory is created, and then another directory is created under it for storing the sample programs.

1.3.2 Overview of Sample Programs

The sample programs consist of following directories.

Directory	Content
TK850¥V850TFT¥include	Sample program common header files
TK850¥V850TFT¥src	Sample program common source files
TK850¥V850TFT¥graphic_sample	Graphic display sample project
TK850¥V850TFT¥washing_machine	Washing machine menu sample project
TK850¥V850TFT¥bin	Bitmap conversion tool
TK850¥V850TFT¥lib	Graphic library

Chapter 2 Experiences

In this chapter, you will experience following development tools with using sample programs.

- Build with Integrated Development Environment (IDE), PM+
- Run the program with Integrated Debugger, ID850QB

You will use the programs that you prepared in "1.3 Sample Programs", as the sample programs for EB-850/JG2+TFT.

By building the sample programs and running the program with ID850-QB, you will be able to understand how to use the development tools (PM+, ID850QB) and the concept of project files which you need to be aware for producing application programs.

The overall steps are as follows:



2.1 Start PM+

Let's start using the development tools.

First, start the PM+.

Select "Windows Start Menu" -> "Programs" -> "NEC Electronics Tools " -> " PM+ V6.30".



2.2 What is PM Plus

In PM+, application programs and environment settings are handled as a single project. It manages series of actions such as programming with the editor, source management, build, and starting debugger.

Also, one or more project files are managed as a workspace.



Project window:

A window in which the project names, source files, and include files are displayed with a tree structure.

Output window:

A window in which the build execution status is displayed.

➡ For details about the menu bar and tool bar, refer to "Help" in PM+. "Help" can be found by selecting "Help" on menu bar, then "PM+ Help".

What is a project?

A project is the unit that PM+ manages. A project refers to application systems developed on PM+ and environment on PM+.

PM+ saves and loads project information in a "project file".

What is a project file?

A project file contains project information that includes the source files, device name, tool options for compiling, editor, and debugger information.

The file name format is "xxxxx.prj".

Project files are created in the directory you specifies when you create a new workspace.

What is a project group?

A project group is a group related projects in an application system.

The target device of each project must be the same within a project group.

What is a workspace?

A workspace is the unit used to manage all the projects and project groups required for one application system.

A workspace file contains one or more project files.

The file name format is "xxxxx.prw".

2.3 Load Workspace (Project)

In this section, you will be using the workspace that you created in "1.3 Sample Programs".

⇒ For creating a new workspace, refer to "Chapter 5 Other Information".

The workspace has information about the build environment for the sample programs.

Select "File" on menu bar and "Open Workspace...". Then, select "graphic_sample.prw" under the directory "TK850¥V850TFT¥graphic_sample".

 PM+-No Workspace [OutPut] <u>_ | ×</u> Build Tool Window Help File Edit Find Layer View Project h 🖪 阳 rece New Ctrl+N 💽 + + + - 😫 🔜 💡 Open... Ctrl+O nsert file 🖸 🔺 差 差 差 🛣 Close - 🗆 🗶 🛄 OutPut <u>- 🗆 ×</u> vew Workspace.. - [EOF] Open Workspace ۸ Save As... Print Preview Ctrl+P Print... Exit PM+ L: X: Y: Open an existing workspace



	Open Worksp	ace				<u>?×</u>		
	Look in: 🔀	graphic_sample		• 🗢 🖻	। 💣 🎟 -			
	include src graphic_sa	mple.prw						
					sar	en the dire nple progr red.	ectory that the ams are	
	File name:	graphic_sample.prw			Opr			
	Files of type:	Workspace File(*.prw	ı]	Ť	Can He			
	S	elect "graphic_sa	imple.prw" a	nd click "(Open".			
Workspace file	name: graphi	c_sample .prw						
	PM+-graphic_sample.p a Edit Find Layer View	rw [ProjectWindow] Project Build Tool Window H	telp		_oad the 'graphic_			
1	🖦 🖌 🗋 🖨 📕 🎒	💽 🛛 X 🖻 💼 🖓 rece	V	▼ + + 4 ★ + 4 ★ ★ ▲ 4				
	ProjectWindow Files Memo Image: State of the stat	_ DutPu	ut					
	ti - ☐ Include Files		Projec	rt				
Eor	Help, press F1							

The workspace file "graphic_sample.prw" contains one project called "graphic_sample". You will use this project "graphic_sample".

CAUTION:

Please ignore when you get a prompt saying that files could not be found. This may be occurred when the installation directory is not the default.

2.4 Check Compiler Option Settings

The options have been set by project file. However, because some options are important and useful, following two settings are covered in this section.

- To set security ID
- To enable C++ comments

2.4.1 Compiler Common Options

Select "Tool" on menu bar, then "Compiler Common Options...". Select "Device" tab on "Compiler Common Options" window, and check following settings.

0	Compiler Common Options	x
	File Startup Link Directive ROM Flash Device	_,
	C 256M Byte Mode	
	BPC Register:	
	Security ID:	
<		
	By checking this check box, 256MB mode is available.	
	When it is checked, -X256M option of the compiler, the assembler and the linker is set.	
	OK Cancel Apply Help	

If you forgot the security ID code stored in the address 70H-79H, the debugger (ID850-QB) will not be able to start.

2.4.2 Preprocessor Tab

Select "Tool" on menu bar in PM+, then "Compiler Options...". Select "Preprocessor" tab, and check "Use C++ Style Comment". This setting allow you to use the C++ comment using "//". It is useful feature when editing programs.

Compiler Options
Detail of Optimization External Register Output File Output Code Message Assembler Others General Input File Preprocessor C Language and Kanji Optimization and Debug Information
Include Search Path[-1]: Edit
\include;include
Define Macro[-D]: Edit
Undefine Macro[-U]: Edit
Limit of Number of Macro[-Xm]:
Command Line Option:
-cpu F3718 -I\include -Iinclude -Xcxxcom -Xpro_epi_runtime=off -Xr -Od -g
OK Cancel Apply Help

2.5 Create Load Module Files

Now, you are going to compile, assemble, and link the program you developed to create load module files. This process is called build.

Click the build button in PM+, or select "Build" on menu bar, then "Build".





PM+-graphic_sample.prw[OutPut] File Edit Find Layer View Project Build Tool Window Help	
1 🖦 - 🗅 😂 🖬 🖨 🖧 X 🗈 🖻 🎇 rece 🔽 🔹 🔹 + 🔸 + - 😤 🔜 💡	
graphic_sample - graphic_sample 🔽 Debug Build 🔄 🖆 🏯 🏥 💠 👗 🧱 📰	
ProjectWindow Files Memo C:\Program Files\NEC Electronics Tools\CA850\W3.10\bin\ca Tools\CA850\W3.10\bin\ca C:\Program Files\NEC Electronics Tools\CA850\W3.10\bin\ca Tools\CA850\W3.10\bin\ca Tools\CA850\W3.10\bin\ca C:\Program Files\NEC Electronics Tools\CA850\W3.10\bin\ca Too	at at at at at at at at at at at at at a

Build has been completed successfully.

What is build?

Build is a function that creates an executable file from source files in a project. PM+ automatically performs compiling, assembling, linking, and other processing actions. To reduce the time for the build, PM+ detects and compiles/assembles only the files that have been updated from the previous build process.

What is rebuild?

Build compiles and assembles only the source files that have been updated from the previous time, whereas rebuild compiles and assembles all the source files.

When the settings, such as compiler options, have been changed, you must rebuild instead of build.

2.6 Check Debugger Settings

As the build process is completed, now you are going configure the debugger settings for debugging. The debugger settings have been set by the project file as well. However, because those settings are important for debugging, some settings are covered in this section.

Select "Tool" on menu bar in PM+, then "Debugger Settings...".



Check if "ID850QB V3.41 V850 Integrated Debugger" is selected on "Debugger".

Debugger Setti	ngs	×
Select Debugge		
Debugger:	ID850QB V3.41 V850 Integrated Debugger	>
File Name:	C:\Program Files\NEC Electronics Tools\ID850QB\V3.41\bin	
Option:		
Debug Target		
Debug Targe	t File:	
C:\TK850\\	/850TFT\graphic_sample\romp.out	
Execute	Symbol Reset after Download	
🔽 Execute	CPU Reset after Download	
Debug Options		
🗖 Downloa	d the Debug Target Files in the same Project Group	
Debug Targe	et File List:	
	OK Cancel Help	

If you cannot select "ID850QB V3.41 V850 Integrated Debugger", use menu bar to select "Project", "Project settings" -> "Tool version settings" -> "Detail settings" -> then select "ID850QB" and "V3.41".

2.7 Check Kit Connection

To debug, connect the kit, MINICUBE2, and PC.

1) Set the switches on MINICUBE2 to "M2" and "T" as shown below.



2) Connect the MINICUBE2 cable to J4 on EB-850/JG2+TFT as shown below.



3) Connect MINICUBE2 to PC with USB cable.



4) Connect EB-850/JG2+TFT to PC with USB cable.



- * For USB connections, MINICUBE2 must be connected first. If you connect the others first, it may break the system.
- * When you disconnect the USB connections, the order must be (1) EB-850/JG2+TFT (2) MINICUBE2.
- * Make sure to turn off the systems before the J4 connection.



Click the debug button III in PM+, or select "Build" on menu bar, then "Debug".

If you do not see the debug button, refer to "2.6 Check Debugger Settings" to configure the settings.

The steps to start the debugger will be explained below.





"Configuration" dialog is opened. Follow the settings below and click "OK".

- Select "4" in "Multiply rate"
- Select "UARTA0" in "Port" at "Target Device Connection" area

Configuration
Chip Name: uPD 70F3718 S.000 Clock Main OSC(MHz) Cancel
Internal Memory Multiply rate Restore
ROM: 512* KBytes Project
RAM: 40960* Bytes Sub OSC(KHz) About
Data Flash: O* KBytes Help
Use Data Flash
Programmable I/O Area
Peripheral Break Monitor Clock Target Device Connection C Break System Port: UARTAD Image: Non Break Image: User Image: User
Mask
Memory Mapping Add Access Size: • 8Bit □ 16Bit ○ 32Bit Add Memory Attribute: Mapping Address & Chip Delete Target · ✓



Click "Yes" when the confirmation dialog for downloading load module file is opened.





NOTE:

It just completes the download of programs (writing the programs on built-in flash memory). It does not mean that the program is running. Therefore, the LCD on the kit does not display anything. To run the demonstration, move on to the next section "2.9 Run Programs".

2.9 Run Programs

Now, you are ready to run the program.

Click the restart button in ID850QB, or select "Run" on menu bar, then "Restart". The sample program runs.

📑 ID850Q8 : graphic_sample.prj	
Fire View Option Run Event Browse Jump Window Help	
Search <	
Search << >> Watch Quick Refresh Close 31 graphic_context g_context;	
33 int	
34 main(int argc, char **argv) 36 36 36	
37	
* 38 _EI<;	
<pre>41 lcd_init(); 42 ret = graphic_init(&config);</pre>	
<pre>43 if (ret != GRAPHIC_ERROR_NORMAL) { goto end;</pre>	
<pre>45</pre>	
417 if (ret != GRAPHIC_ERRÖR_NORMAL) (goto end; Run sam	ple program
1 47 5 50 50 51 /* Sample */	pio piogram
* 48 49 50 50 51 * goto end; * Run sam * 52 53 graphic_sample(); *	
* 55 return 0;	
56 57 *	
main.c#35 main 00000004 0.0 msec	///
ID850QB : graphic_sample.prj	
File Edit View Option Run Event Browse Jump Window Help	
Source (main.c)	
Search << >> Watch Quick Refresh Close	
31 graphic_context g_context; 32 /*	
34 main(int argc, char **argu)	



When programs are running, the status bar will be red.


Check if graphics are displayed on the LCD panel of EB-850/JG2+TFT.

You could confirm the sample program is working.

- Menu can be opened by touching the title bar.
- For more information about the sample programs, refer to "Application Note" document.
- The programs downloaded by ID850QB cannot be used without MINICUBE2 connection. For stand-alone operation, write the HEX file with QB-Programmer.

2.10Stop Programs

Now, you are going to stop the program.

Click the stop button III in ID850QB, or select "Run" on menu bar, then "Stop".

File Edit View Option	sample.prj	
	↗涎≐ ◙≋≋ ◙ቈ≜◙ ≥≼≋ ฃ≀⊽ೇ⊗	
i∎ Source (main.c)		
	> Watch Quick Refresh Close paphic_context g_context;	
32 33 i	*	
* > 34	ain(int argc, char **argu) int ret;	
* > 35 36 37 * 38	_EI();	
39		
* 41 * 42	<pre>/* Initialize */ lcd_init(); ret = graphic_init(&config); if (ret != GRAPHIC_ERROR_NORMAL) (</pre>	
* 43 * 44 45	goto ena,	
* 46 * 47 * 48	<pre>ret = g_context_init(&g_context); if <ret !="GRAPHIC_ERROR_NORMAL)" <="" end;<="" goto="" pre=""></ret></pre>	
49	>	
* 51 52 53	/* Sample */ graphic_sample();	
54 e		
* 55 * 56) 57	×	
main.c#35	main 00000004 RUN	
말# ID850QB : graphic_		
File Edit View Option	Run Event Browse Jump Window Help	
File Edit View Option	IRUN Event Browse Jump Window Help IIIII ▲ 『『『『『『『『『』』 ▲ 由歌 星 美国家 聞 ? ♥ ?? @	
File Edit View Option	i Run Event Browse Junp Window Help → M ▲ 図書■ 型素校 Q 書曲配 書写電 面 ? ♥ ?? ⑧ river.c)	
File Edit View Option	IRUN Event Browse Jump Window Help → M ▲ 図書画 圓元 校 Q 書 由 医 書 译 ▼ ジ ③ river.c) > Watch Quick Refresh Close	
File Edit View Option III IP IP IN IN IN IIII IP IP IN IN IN IN IIIII IP IP IN IN IN IN IN IIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIII	IRUN Event Browse Jump Window Help · 에스 전종画 四點之 오슈슈 또 물 종종 집 옷 맛 ? (20) river.c) > Watch Quick Refresh Close peturn (unsigned int)=GET_PIXEL_HEMORY(x, y);	
File Edit View Option III IP IP IP IP IP III IP IP IP IP IP IP IIII IP	I Run Event Browse Jump Window Help I M ▲ 歐國國 圓龍電 Q ▲ 由 取 基 通信 图 Y F 20 river.c) > Watch Quick Refresh Close return (unsigned int)=GET_PIXEL_HEHORY(x, y);	
File Edit View Option III III IIII IIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIII	IRUN Event Browse Jump Window Help · 에스 전종画 四點之 오슈슈 또 물 종종 집 옷 맛 ? (20) river.c) > Watch Quick Refresh Close peturn (unsigned int)=GET_PIXEL_HEMORY(x, y);	
File Edit Wew Option III III IIII IIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIII	IRUN Event Browse Jump Window Help IMA E IMA E IMA E Image: State	
File Edit View Option III IP	INU Event Browse Jump Window Help INI INI	
File Edit Wew Option III IP IP IP IP Source (graphic.d) IP IP IP IP IP IP IP IP IP IP IP IP IP IP IP IP IP IP	<pre>i Run Event Browse Jump Window Help i M @ @ @ @ @ @ @ @ @ @ @ @ @ @ @ @ @ @</pre>	
File Edit Wew Option III IP IP IP IP Source (graphic.d) IP IP IP IP IP IP IP IP IP IP IP IP IP IP IP IP IP IP	<pre>i Run Event Browse Jump Window Help i M</pre>	
File Edit Wew Option III Image: heat state stat	<pre>a Run Event Browse Jump Window Help I M L Define The second second</pre>	
File Edit Wew Option III III IIII IIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIII	<pre>a Run Event Browse Jump Window Help i Min Event Browse Jump Window Help i Min Event Browse Jump Window Help i Min Event Refresh Close return (unsigned int>*GET_PIXEL_MEMORY(x, y); * * * * * * * * * * * * * * * * * *</pre>	
File Edit Wew Option III ▶ ▶ ▶ ▲ ▲ III ▶ ▶ ▲ ▲ ▲ ▲ IIII ▶ ▶ ▲	<pre>a Run Event Browse Jump Window Help I M L Definition The second se</pre>	
File Edit Wew Option III III IIII IIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIII	<pre>i Run Event Browse Jump Window Help i M i Riferent Dose return (unsigned int)*GET_PIXEL_MEMORY(x, y); wdatch Quick. Refrest Dose return (unsigned int)*GET_PIXEL_MEMORY(x, y); wdatch Quick. Refrest Dose return (unsigned short x, unsigned short y, unsigned in et_pixel(unsigned short y, unsigned short)color; // Connect x*/</pre>	
File Edit Wew Option III III IIII IIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIII	<pre>i Run Event Browse Jump Window Help i M i Rife in the interval of the in</pre>	
File Edit Wew Option III III IIII IIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIII	<pre>i Run Event Browse Jump Window Help i M i Riferent Dose return (unsigned int)*GET_PIXEL_MEMORY(x, y); wdatch Quick. Refrest Dose return (unsigned int)*GET_PIXEL_MEMORY(x, y); wdatch Quick. Refrest Dose return (unsigned short x, unsigned short y, unsigned in et_pixel(unsigned short y, unsigned short)color; // Connect x*/</pre>	
File Edit Wew Option III III IIII IIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIII	<pre>i Run Event Browse Jump Window Help i M i Riferent Dose return (unsigned int)*GET_PIXEL_MEMORY(x, y); wdatch Quick. Refrest Dose return (unsigned int)*GET_PIXEL_MEMORY(x, y); wdatch Quick. Refrest Dose return (unsigned short x, unsigned short y, unsigned in et_pixel(unsigned short y, unsigned short)color; // Connect x*/</pre>	
File Edit Wew Option III III IIII IIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIII	<pre>i Run Event Browse Jump Window Help i M i Riferent Dose return (unsigned int)*GET_PIXEL_MEMORY(x, y); wdatch Quick. Refrest Dose return (unsigned int)*GET_PIXEL_MEMORY(x, y); wdatch Quick. Refrest Dose return (unsigned short x, unsigned short y, unsigned in et_pixel(unsigned short y, unsigned short)color; // Connect x*/</pre>	
File Edit View Option III IIII IIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIII	<pre>i Run Event Browse Jump Window Help i M i Riferent Dose return (unsigned int)*GET_PIXEL_MEMORY(x, y); wdatch Quick. Refrest Dose return (unsigned int)*GET_PIXEL_MEMORY(x, y); wdatch Quick. Refrest Dose return (unsigned short x, unsigned short y, unsigned in et_pixel(unsigned short y, unsigned short)color; // Connect x*/</pre>	

When the program stops, the status bar changes back to the original color.

2.11 Close Debugger (ID850QB)

Select "File" on ID850QB menu bar, then "Exit".

ID850QB:graphic_sample ile Edit View Option Run Open Ctrl+O Save As Close	Event Browse Jump Window Help	_ <u> </u>
Download Load Module Upload	rch Quick Refresh Close	
Debugger Peret Exit	<pre>> el(unsigned short x, unsigned short y, unsigned in *GET_PIXEL_MEMORY(x, y) = (unsigned short)color; yook</pre>	
332 332 H 338 H 339 H 339 H 348 H 342 H 342 H 342 H 342 H 342 H 342 H 345 H 345 H 348 H 349 H 349 H 351 H 351	<pre>int</pre>	
graphic_driver.c#326	get_pixel 00052D9A 9m 51s 510ms	
ID850		×
	This will end your Debugger session.	
?	Do you want to save the settings in the project file	?

If you click "Yes", it saves the current settings in the project file, and then closes the ID850QB. It is recommended to save the settings as it saves the window you used, window size, layout, etc. If you click "No", it does not save the current settings and closes the ID850QB.

2.12Quit PM+

Select "File" on PM+ menu bar, then "Exit PM+".



PM+ is closed.

The experiences section ends here.

You can find more information about basic operations and other useful features in "Chapter 5 Other Information".

Chapter 3 Hardware Specifications

In this chapter, the hardware of EB-850/JG2+TFTwill be explained.

Microcontroller	µPD70F3718 *V850ES/JG2
Clock	Main system clock: 20MHz (5MHz x 4 multiplying)
	Sub system clock: 32.768KHz
LCD controller	S1D13A05 (EPSON)
	External memory bus connection
	256Kbyte display built-in memory
LCD panel	NL2432HC22-41K (NEC LC technology)
	3.5 inches TFT
	Touch panel
	240x320 dots, 250,000 color
Interface	USB (J8)
	For power supply (no driver bundled)
	Connector for MINICUBE2 connection (J4)
Power supply voltage	5V (J8)

* The name with bracket is the name printed on the board.



3.1 Layout of hardware functions

3.2 Hardware Functions

3.2.1 SW1

SW1 is the reset switch. You can reset the micro controller by pressing this switch.

3.2.2 SW2

SW2 is the DIP switches connected to "P75/ANI5"-"P711/ANI11" in CPU. By setting this to ON, it can connect to GND. If you do not use as A/D, set this to ON. As pull-up resistor is not connected, it will not be "High" by setting the switch OFF.

Normally, set it ON.

3.2.3 SW3 (INTP0)

SW3 is the Push switch connected to "P03/INTP0" pin in microcontroller. When you push the switch, it becomes "Low", and when you release the switch, it becomes "High" by pull-up resistor.

3.2.4 SW4 (INTP1)

SW4 is the Push switch connected to "P04/INTP1" pin in microcontroller. When you push the switch, it becomes "Low", and when you release the switch, it becomes "High" by pull-up resistor.

3.2.5 D3

D3 is the power LED. When the power is ON, it lights.

3.2.6 D4

D4 is the LED connected to "P50" pin in microcontroller. When you drive the "P50" to "Low", it lights.

3.2.7 D5

D5 is the LED connected to "P51" pin in microcontroller. When you drive the "P51" to "Low", it lights.

3.2.8 J1

J1 is the jumper switch pin to select power supply.

1-2 Short	Use power supply from J2 connector
2-3 Short	Use USB power supply from J8 (USB) connector

3.2.9 J2

J2 is the connector for external power supply. Supply power with 5V.

3.2.10 J4

J4 is the connector for MINICUBE2 connection.

3.2.11 J8

J8 is the connector for USB connection. It is connected to USB pin from LCD driver (S1D13A05), but as this kit does not bundle USB driver, use it for power supply purpose.

Chapter 4 Troubleshooting

This chapter describes how to solve troubles you may face.

4.1 If you cannot find USB driver when you connect MINICUBE2 and PC

Check Point 1

If you use USB hub, do not use it.

Check Point 2

Check if you installed "MINICUBE2 USB Driver" in "1.2 Installation of Development Tools". If not, install the driver.

4.2 Error when you start the debugger

There could be several reasons to make this error happened.

The solving processes differ depending on errors. Please check the error message first. The solving processes for each error are described below.

4.2.1 "Can not communicate with ICE..." (F0100 or A0109)

Check Point 1

If you use USB hub, do not use it. (USB hub is not supported)

Check Point 2

If above check point is not the problem, close the debugger and disconnect the USB cable from PC. Re-connect USB cable properly to both the PC and MINICUBE2, and then re-start the debugger.

4.2.2 "Incorrect ID Code." (Ff603)

This error occurs when the security ID stored in microcontroller built-in flash memory is different from the ID code you entered at the start of debugger.

Security ID entry area at the start of debugger

[D Code -	

Check Point 1

Enter correct security ID and click OK on the configuration window.

Check Point 2

If you forgot the security ID, you need to erase the microcontroller built-in flash memory. Before erasing it, check if you actually set the security ID. Also remember the code you set for the security ID.

After this, erase the flash memory with referring to "5.4 QB-Programmer".

4.2.3 "The on-chip debug function had been disabled in the device." (F0c79)

This error occurs when the value at address C3H (On-chip debug option byte) in microcontroller built-in flash memory is incorrect. You need to erase the flash memory.

Check Point 1 Check if you actually set the correct on-chip debug option byte. If it is not correct, set it correctly.

Check Point 2 Erase the flash memory with referring to "5.4 QB-Programmer".

4.2.4 "Disabling the on-chip debug function is prohibited." (F0c33)

Basically, this error occurs when you start (download) the debugger without doing the settings correctly. Do the same checking processes as "4.2.3 The on-chip debug function had been disabled in the device. (F0c79)".

Chapter 5 Other Information

This chapter explains about some important and useful operation techniques of development tools, and circuit diagram of the kit for developing of user programs.

5.1 Create a New Workspace (Project)

5.2 Register Additional Source File

5.3 Debugger Useful Tips

5.4 QB-Programmer

5.5 Circuit Diagram

5.1 Create a New Workspace

Now, create a new workspace and project.

PM+ allows you to create a new project including all the information needed for build by following the step-by-step dialog.

Select "File" \rightarrow "New Workspace..." in the menu bar of PM+.







🔽 Select only Installed Tools

Please select the Tools from NEC Electronics to be used

The following tools have been excluded from selection because they are not installed. CA850, ID850, ID850NW, SM850, SM+ for V850, TW850

< Back

Click "Detail Settings" button.

9. Confirmation V850 Microcontrollers Only (4-6)

Cancel

Help

Detail Setting...

Next >



Click "Next".



I

New WorkSpace - Step 6/9 [LinkDirective File]		×	
Please specify the Link Directive File.	Workspace Information Select Tools Select Real-Time 0S Startup File(#) Startup File(#) Secup Source Files Select Debugger Select Debu		Select "Copy and the sample"
The link directive file describes information about the allocation of program In case of using the sample file, Please customize it according as your tar ROM to the internal ROM area in ROM less mode, please select "use inte The specified link directive file is registered to the Project Related Files.	n code and data. get. If you map the external		

Click "Next".

se setup Source Files. urce File Name :	1. Workspace Information
	Add 2. Select Tools
	3. Select Real-Time US
	4. Startup File(#)
	Remove All 5. Register Mode(#)
	6. Link Directive File(#)
	>>7. Setup Source Files
	Up 8. Select Debugger
	9. Confirmation
	(#) V950 Microcontrollers Only (4-6)
	specifying the list file or the folder. lies here, and you can setup source files using [Project]->[Project

Click "Next".

use

EB-850/JG2+TFT USER'S MANUAL



Click "Finish".



This completes workspace and project creation.

Additional source files can be registered at any time thereafter.

➡ For details, refer to "5.2 Register Additional Source File".

Next, you are going to set the security ID.

Select "Tool" on menu bar, then "Compiler Common Options...".



↓	
Compiler Common Options	×
File Startup Link Directive ROM Flash Dev	ice
□ 256M Byte Mode	
BPC Register:	"Compiler Common Option
Security ID:	Settings" dialog opens
This edit box can be specified a security ID by hexader	rimal
When it is specified, -Xsid option of the linker is set.	
OK Cancel	Apply Help
Select "Device" tab	
\checkmark	
Compiler Common Options	×
File Startup Link Directive ROM Flash Dev	vice
C 256M Byte Mode	
BPC Register:	Enter "0xFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF
	(F x 20) in "Security ID" if there will not
	be a problem entering it.
This edit box can be specified a security ID by hexade	cimal.
When it is specified, Xsid option of the linker is set.	
OK Cancel	Apply Help

Click "OK".

This completes the setting of the security ID.

5.2 Register Additional Source File

This section explains how to register additional source files to a project.

The following example shows that the additional source files "b.c" and "c.c" are registered to the project which has the source file "a.c".

Place the cursor on the source file in the PM+ project window, and select "Add Source Files..." displayed in the right-click menu.





5.3 Debugger Useful Tips

This section describes some useful techniques for the debugger (ID850QB).

5.3.1 Change Display of Buttons

Execution controls (run, stop, step-in debugging, reset, etc) and opening functional window can be made by below buttons. However, it could be difficult to know which button does what.



In this case, select "Option" on menu bar, then "Debugger Option". Check "Pictures and Text" on "Tool Bar Picture" setting area.

- Tool Bar Pi <u>c</u> tures	
Pictures and Text	
Sector 2010 100 100 100 100 100 100 100 100 10	
O Pictures only	

With this setting, the buttons display the text as well, so that it is easier to know what they are.



5.3.2 Display Source List and Function List

When you wish to see source file list or function list, select "Browse" on menu bar, then "Other" -> "List" to open the list window. The information in the windows is synchronized with the source window. Therefore, it is not just for referring to the list, but it is useful when you wish to update files or functions.



5.3.3 Set/Remove Breakpoints

Breakpoints are set or removed by clicking lines in which " * " is displayed It changes to "B" when you click "*" (the breakpoint is set). It changes to "*" when you click "B" (the breakpoint is canceled).





	🔟 Source (main.c)		
Breakpoint was set	Search B *	<u> </u>	watch game1(void UCHAR UCHAR int	Quick) buff[] = ↓ c, t; ▲ r; ↓

5.3.4 Display Global Variables

With using Watch window, you can display global variables. There are several ways to register global variables to Watch window. In this section, how to register global variables from source window is described.

1) Right-click the variable on Source window, then select "Add Watch..."

Source (78K0R_Voice.c)	<u> </u>
Search << >> Watch Quick Refresh Close	
88 static U16 Adpcm_Work[16];	▲
89 static U8 ucPlaySts;	Move
90 static U8 stop_Led 91 static U8 output_count;	Mix
92 static U16 output_data[2];	
93 static US PlauMode:	Add Watch
U32 uiIntCoun	Symbol
94 95 95 96 97 98 97 98 98 97 98 98 97 98 98 98 98 98 98 98 98 98 98 98 98 98	
96 U16 usKeyStsC 97 U8 ucKeyStsL	
98	Break when Write to
	Break when Read fr

2) Add Watch dialog opens. Click "OK".

Add Watch	×
<u>N</u> ame:	s_count
Radix:	● Proper O Hex O Dec O Oct O Bin O String
Size:	€ Adaptive C Byte C Word C Double Word
N <u>u</u> mber:	
	OK Cancel <u>R</u> estore <u>H</u> elp

3) Adding a variable to watch window is completed

🖻 Watch		
Add	Delete	Up
c_count	0×	00
•	▶ ◀	

5.3.5 Display Global Variables While Programs Are Running

Global variable can be referred even when the programs are running.

1) Select "Option" menu -> "Extended Option...". Follow below settings in the "RAM Monitor" area.

☑ Use MINICUBE Extended Function	
RAM Monitor	
Break When Readout: © RR <u>M</u>	Setting Select
Redraw Interval: 500	msec
	Specify the redraw interval time (default: 500ms). The time should be between 100 and 65500 with 100ms units.

 Right-click the variable in Watch window while programs are running, then select "RRM Setting...".

🖻 Watch	
Add Delete	Up
+Adpcm_Work[16]	Break when Access to this Variable Break when Write to this Variable Break when Read from this Variable Clear
	RRM Setting

 RRM Setting dialog opens. Click "Set" button to complete the setting, then click "Close" button to close the dialog.

RRM Setting			×	
Address 1 Dxfcfba 2	s Size	Symbol Bytes	Set <u>C</u> lose <u>R</u> estore <u>Cl</u> ear <u>Delete</u> <u>Jump</u> <u>H</u> elp	to complete the setting. Then, click "Close" to close the dialog.

This completes the settings. This is a useful function, but there are some notes for this function.

- The maximum size of variable area is 16byte in total when programs are running.
 - The maximum number of variable areas is 8 areas when programs are running.
- Program momentarily breaks upon a read of variables.

5.3.6 **Display Local Variables**

Local variable window is used to display local variables.

By clicking the button below, you can open the local variable window.

Unlike global variables, local variables cannot be displayed when programs are running.



5.3.7 **Display Memory and SFR Contents**

Similar to the local variable window, you can display memory and SFR contents.

By clicking the button below, you can open the memory window.

8	🛙 Men	iory															. 🗆	×
	Sear	ch		<<		\gg	Re	efres	า	DM	M		Clos	e				
2	Addr	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	
	CF00		00	00	00	00	00	00	00	10	10	10	10	02	10	10	02	
~	FCF10	1 × × .	05	11	11	A9	ЗB	09	09	13	00	00	01	04	\mathbf{FF}	00	00	
F	CF20	00	00	00	00	03	00	00	00	00	00	\mathbf{FF}	\mathbf{FF}	03	00	00	00	
F	FCF30	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
F	FCF40	00	00	00	00	00	00	00	00	FA	FC	00	00	00	00	94	0D	
F	CF50	00	00	00	00	FF	01	\mathbf{FF}	00	36	36	\mathbf{FF}	03	00	00	00	00	_
F	CF60	00	00	00	00	FF	FF	FF	FF	94	0D	ΕE	0B	52	0C	94	0D	Ŧ
F	CF70	00	00	00	00	00	00	00	00	00	00	02	04	AB	72	F2	00	1
Ę	CF80	00	00	01	00	00	00	03	00	02	04	00	58	03	E9	02	02	Ŧ

٠

By clicking the button below, you can open the SFR window.

SFR



5.4 QB-Programmer

Also, you can use QB-Programmer when you wish to run programs without the MINICUBE2 connection (stand-alone) after completing debugging.

As ID850QB downloads monitoring files for debugging, the programs cannot be run with stand-alone. QB-Programmer does not write monitoring files, so that the programs can be run with stand-alone.

- 🎇 QB-Programmer <u>File D</u>evice <u>H</u>elp Programmer o-Programmer startup Command standby QB-Programmer : V2.21 Firmware : V4.04 Device Name Firmware: Parameter file Name Version Load file Name Date : Chksum : Area File checksum Type : Chksum : Area Connection to device Port Pulse Speed Range Freq. Multiply ľ NUM Ready
- 1) Start QB-Programmer.

2) Click "Setup" button.

Device Setup
Standard Advanced
Parameter File PRM File Read
Target Device Connection Supply Oscillator
🔽 On Target
Port Frequency MHz
Speed Multiply rate
Operation Mode
C Chip Start
C Block End
🗖 Show Address
OK Cancel

3) Click "PRM File Read" button.

Open		? 🛛
Look in: 🗀 Pf	RM70F3724_V100	▼ 🗢 🗈 💣 🖩 -
70F3715_CSI	3.prm 3.prm 70F3719_CSI0.prm 0.prm 70F3719_CSI3.prm 3.prm 70F3720_CSI0.prm 0.prm 70F3720_CSI3.prm 3.prm 70F3720_CSI3.prm 3.prm 70F3720_CSI3.prm	70F3722_CSI0.prm 70F3722_CSI3.prm 70F3723_CSI0.prm 70F3723_CSI3.prm 70F3724_CSI0.prm 70F3724_CSI3.prm 70F3724_CSI3.prm
File name: 7	0F3718_CSI0.prm	Open
Files of type: F	'RM Files(*.PRM)	▼ Cancel

4) Select the file "PRM¥PRM70F3724_V100¥70F3718_CSI0.prm" from CD-ROM.

Device Setup
Standard Advanced
Parameter File 70F3718_CSI0.prm PRM File Read
Target Device Connection Supply Oscillator
Port UART-ch0 Frequency 5.00 MHz Speed 153600bps Multiply rate 4.00
Operation Mode
Chip Start 000
C Block End 013 💌
Show Address
OK Cancel

5) Check all the settings are shown above.



6) Click "Erase" button, when you need to erase the flash memory.



7) Click "Load" button when you need to write programs. Select a HEX file.



8) Click "Auto Procedure" button to start the writing process.

5.5 Circuit Diagram

From following page, it shows the circuit diagram of the kit.