

User's Manual

V850ES/Jx3-H – Plug it!

**Demonstration Kit for the V850ES/Jx3-H
32-bit RISC microcontroller**

- **The information in this document is current as of June, 2008. The information is subject to change without notice. For actual design-in, refer to the latest publications of NEC Electronics data sheets or data books, etc., for the most up-to-date specifications of NEC Electronics products. Not all products and/or types are available in every country. Please check with an NEC Electronics sales representative for availability and additional information.**
- No part of this document may be copied or reproduced in any form or by any means without the prior written consent of NEC Electronics. NEC Electronics assumes no responsibility for any errors that may appear in this document.
- NEC Electronics does not assume any liability for infringement of patents, copyrights or other intellectual property rights of third parties by or arising from the use of NEC Electronics products listed in this document or any other liability arising from the use of such products. No license, express, implied or otherwise, is granted under any patents, copyrights or other intellectual property rights of NEC Electronics or others.
- Descriptions of circuits, software and other related information in this document are provided for illustrative purposes in semiconductor product operation and application examples. The incorporation of these circuits, software and information in the design of a customer's equipment shall be done under the full responsibility of the customer. NEC Electronics assumes no responsibility for any losses incurred by customers or third parties arising from the use of these circuits, software and information.
- While NEC Electronics endeavors to enhance the quality, reliability and safety of NEC Electronics products, customers agree and acknowledge that the possibility of defects thereof cannot be eliminated entirely. To minimize risks of damage to property or injury (including death) to persons arising from defects in NEC Electronics products, customers must incorporate sufficient safety measures in their design, such as redundancy, fire-containment and anti-failure features.
- NEC Electronics products are classified into the following three quality grades: "Standard", "Special" and "Specific".
The "Specific" quality grade applies only to NEC Electronics products developed based on a customer-designated "quality assurance program" for a specific application. The recommended applications of an NEC Electronics product depend on its quality grade, as indicated below. Customers must check the quality grade of each NEC Electronics product before using it in a particular application.
"Standard": Computers, office equipment, communications equipment, test and measurement equipment, audio and visual equipment, home electronic appliances, machine tools, personal electronic equipment and industrial robots.
"Special": Transportation equipment (automobiles, trains, ships, etc.), traffic control systems, anti-disaster systems, anti-crime systems, safety equipment and medical equipment (not specifically designed for life support).
"Specific": Aircraft, aerospace equipment, submersible repeaters, nuclear reactor control systems, life support systems and medical equipment for life support, etc.

The quality grade of NEC Electronics products is "Standard" unless otherwise expressly specified in NEC Electronics data sheets or data books, etc. If customers wish to use NEC Electronics products in applications not intended by NEC Electronics, they must contact an NEC Electronics sales representative in advance to determine NEC Electronics' willingness to support a given application.

(Note)

(1) "NEC Electronics" as used in this statement means NEC Electronics Corporation and also includes its majority-owned subsidiaries.

(2) "NEC Electronics products" means any product developed or manufactured by or for NEC Electronics (as defined above).

M8E 02. 11-1

CAUTION

This is a Test- and Measurement equipment with possibility to be significantly altered by user through hardware enhancements/modifications and/or test or application software. Thus, with respect to Council Directive 89/336/EEC (Directive on compliance with the EMC protection requirements), this equipment has no autonomous function. Consequently this equipment is not marked by the CE-symbol.

EEDT-ST-005-10

CAUTION

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. All test and measurement tool including the workbench must be grounded. The user/operator must be grounded using the wrist strap. The connectors and/or device pins should not be touched with bare hands.

EEDT-ST-004-10

For customers in the European Union only

Redemption of Waste Electrical and Electronic Equipment (WEEE) in accordance with legal regulations applicable in the European Union only: This equipment (including all accessories) is not intended for household use. After use the equipment cannot be disposed of as household waste. NEC Electronics (Europe) GmbH offers to take back the equipment. All you need to do is register at <http://www.eu.necel.com/weee>

Regional Information

Some information contained in this document may vary from country to country. Before using any NEC product in your application, please contact the NEC office in your country to obtain a list of authorized representatives and distributors. They will verify:

- Device availability
- Ordering information
- Product release schedule
- Availability of related technical literature
- Development environment specifications (for example, specifications for third-party tools and components, host computers, power plugs, AC supply voltages, and so forth)
- Network requirements

In addition, trademarks, registered trademarks, export restrictions, and other legal issues may also vary from country to country.

NEC Electronics Inc. (U.S.)

Santa Clara, California
Tel: 408-588-6000
800-366-9782
Fax: 408-588-6130
800-729-9288

NEC Electronics Hong Kong Ltd.

Hong Kong
Tel: 2886-9318
Fax: 2886-9022/9044

NEC Electronics (Europe) GmbH

Duesseldorf, Germany
Tel: 0211-65 03 0
Fax: 0211-65 03 1327

NEC Electronics Hong Kong Ltd.

Seoul Branch
Seoul, Korea
Tel: 02-528-0303
Fax: 02-528-4411

Sucursal en España

Madrid, Spain
Tel: 091- 504 27 87
Fax: 091- 504 28 60

NEC Electronics Singapore Pte. Ltd.

Singapore
Tel: 65-6253-8311
Fax: 65-6250-3583

Succursale Française

Vélizy-Villacoublay, France
Tel: 01-30-67 58 00
Fax: 01-30-67 58 99

NEC Electronics Taiwan Ltd.

Taipei, Taiwan
Tel: 02-2719-2377
Fax: 02-2719-5951

Filiale Italiana

Milano, Italy
Tel: 02-66 75 41
Fax: 02-66 75 42 99

NEC do Brasil S.A.

Electron Devices Division
Guarulhos, Brasil
Tel: 55-11-6465-6810
Fax: 55-11-6465-6829

Branch The Netherlands

Eindhoven, The Netherlands
Tel: 040-244 58 45
Fax: 040-244 45 80

Branch Sweden

Taeby, Sweden
Tel: 08-63 80 820
Fax: 08-63 80 388

United Kingdom Branch

Milton Keynes, UK
Tel: 01908-691-133
Fax: 01908-670-290

Revision History

Date	Revision	Chapter	Description
17-02-2009	V1.00	---	First release
08-05-2009	V1.10	13	Change of the schematic image format

Table of Contents

1. Introduction	10
1.1 Main features of <i>V850ES/Jx3-H - Plug it!</i>	10
1.2 System requirements	11
1.3 Package contents	11
1.4 Trademarks	11
2. <i>V850ES/Jx3-H - Plug it!</i> system configuration	12
2.1 <i>V850ES/Jx3-H - Plug it!</i>	12
2.2 Host computer	12
2.3 Power supply via USB interface	12
3. <i>V850ES/Jx3-H - Plug it!</i> components	13
3.1 SW1, Configuration Switch (DIP-Switch)	14
3.2 SW2, RESET button	15
3.3 SW3, Switch (INTP12)	15
3.4 SW4, Switch (INTP13)	15
3.5 JP1 and JP2, Power Supply selectors	15
3.6 JP3, Power LED	16
3.7 LED1, power LED	16
3.8 CN4, AC power supply connector	16
3.9 J1, optional power supply connector	16
3.10 NWIRE1, N-Wire connector	16
3.11 U3, Seven-segment-LED	17
3.12 USB1, serial interface connector	18
3.13 USB2, universal serial bus connector	18
3.14 Layout of solder-short pads	19
4. On-Chip debugging	20
4.1 OCD via TK-V850 On-Board debug function	21
4.1.1 Resources used by TK-V850 OCD Interface	21
4.2 OCD via N-Wire emulator	23
5. <i>V850ES/Jx3-H - Plug it!</i> installation and operation	24
5.1 Getting started	24
5.1.1 CD-ROM contents	24
6. Hardware installation	25

7. Software installation.....	25
7.1 IAR Systems Embedded Workbench for V850 installation.....	25
7.2 Sample program installation.....	25
7.3 USB Driver Installation	26
7.3.1 Installation on Windows 2000	26
7.3.2 Installation on Windows XP.....	30
7.4 Confirmation of USB Driver Installation	33
8. IAR sample session	34
9. Troubleshooting.....	39
10. Flash Programmer WriteEZ1.....	40
10.1 Device Setup.....	40
10.2 Using WriteEZ.....	41
11. Sample programs.....	42
11.1 General Introduction.....	42
11.2 “Jx3-H_PLUGIT_DEMO” sample program	43
11.2.1 How to run the sample program.....	43
11.3 “Jx3-H_PLUGIT_COM” Serial Conversion sample program	44
11.3.1 Development Environment.....	45
11.3.2 System Structure.....	45
11.3.3 How to run the sample program.....	46
11.3.4 Communication Device Class (CDC).....	49
11.3.5 Processing Flow.....	50
11.3.6 File Structure	53
11.3.7 Descriptor Information.....	53
11.4 “Jx3-H_PLUGIT_STORAGE” Mass Storage sample program.....	56
11.4.1 Development Environment.....	57
11.4.2 System Structure.....	57
11.4.3 How to run the sample program.....	58
11.4.4 Mass Storage Class (MSC)	59
11.4.5 SCSI Command Process	59
11.4.6 Processing Flow.....	60
11.4.7 File Structure	64
11.4.8 Descriptor Information.....	64
12. Cables	67
12.1 USB interface cable (Mini-B type)	67
13. Schematics	68

List of Figures

Figure 1: V850ES/Jx3-H - Plug it! system configuration	12
Figure 2: V850ES/Jx3-H - Plug it! Components	13
Figure 3: Seven-segment-LED U3	17
Figure 4: USB1, USB Mini-B Type Host Connector Pin Configuration	18
Figure 5: Solder-short pad opened shape	19
Figure 6: Solder-short pad shortened shape	19
Figure 7: On-Chip debugging	20
Figure 8: V850 OCD Interface Resources	22
Figure 9: Device File Installation	25
Figure 10: Found New Hardware Wizard (Windows 2000)	26
Figure 11: Search Method (Windows 2000)	27
Figure 12: Driver File Location (Windows 2000)	27
Figure 13: Address Specification 2 (Windows 2000)	28
Figure 14: Driver File Search (Windows 2000)	28
Figure 15: USB Driver Installation Completion (Windows 2000)	29
Figure 16: Found New Hardware Wizard 1 (Windows XP)	30
Figure 17: Found New Hardware Wizard 2 (Windows XP)	30
Figure 18: Search Location Specification 1 (Windows XP)	31
Figure 19: Search Location Specification 3 (Windows XP)	31
Figure 20: Windows XP Logo Testing (Windows XP)	32
Figure 21: USB Driver Installation Completion (Windows XP)	32
Figure 22: Windows Device Manager	33
Figure 23: IAR Embedded Workbench	34
Figure 24: IAR Embedded Workbench Project Workspace	35
Figure 25 : Device selection	35
Figure 26: Include directory	36
Figure 27: IAR debugger options	36
Figure 28: Embedded Workbench Linker Configuration	37
Figure 29: MINICUBE Hardware Setup Dialogue	38
Figure 30: IAR C-SPY debugger	39
Figure 31: WriteEZ1 User Interface	40
Figure 32: WriteEZ1 Device Setup Dialogue	40
Figure 33: WriteEZ1 Device Menu	41
Figure 34: Project Window Jx3-H_PLUGIT_DEMO	43
Figure 35: Project Window Jx3H_PLUGIT_COM	44
Figure 36: Development Environment	45
Figure 37: System Structure of the sample program	45
Figure 38: Jx3H_PLUGIT_COM-sample Debug Communication Port	46
Figure 39: Found New Hardware Wizard 1 (Windows XP)	46
Figure 40: Found New Hardware Wizard 2 (Windows XP)	47
Figure 41: Search Location Specification 1 (Windows XP)	47
Figure 42: Jx3H_PLUGIT_COM-sample Application Communication Port	48
Figure 43: Flow Chart Jx3H_PLUGIT_COM application	50
Figure 44: Flow Chart Jx3H_PLUGIT_COM Endpoint 0 Monitoring Process	51
Figure 45: Flow Chart Jx3H_PLUGIT_COM Endpoint 1 Monitoring Process	52
Figure 46: Project Window Jx3H_PLUGIT_STORAGE	56
Figure 47: Development Environment	57
Figure 48: System Structure of the sample program	57
Figure 49: Jx3H_PLUGIT_STORAGE sample Mass Storage Device	58
Figure 50: Flow Chart Jx3H_PLUGIT_COM USB Initialization Process	60
Figure 51: Flow Chart Jx3H_PLUGIT_COM Endpoint 0 Monitoring Process	61
Figure 52: Flow Chart Jx3H_PLUGIT_COM Endpoint 1 Monitoring Process	62
Figure 53: Flow Chart Jx3H_PLUGIT_COM SCSI Command Process	63
Figure 54: USB interface cable (Mini-B type)	67
Figure 55: V850ES/Jx3-H - Plug it! schematics 1/3	68
Figure 56: V850ES/Jx3-H - Plug it! schematics 2/3	69
Figure 57: V850ES/Jx3-H - Plug it! schematics 3/3	70

List of Tables

Table 1: Mode setting, switch SW1	14
Table 2: Power supply selectors, JP1 and JP2	15
Table 3: Power LED1 setting.....	16
Table 4: CN4 connector	16
Table 5: Display Examples.....	17
Table 6: Pin Configuration of Connector USB1.....	18
Table 7: Solder-short pad connection	19
Table 8: OCD via TK-78K0R On-Board debug function	21
Table 9: OCD via QB-V850MINI emulator	23
Table 10: <i>V850ES/Jx3-H - Plug it!</i> CD-ROM directory structure.....	24
Table 11: WriteEZ action buttons	41
Table 12: Sample directory structure	42
Table 13: Example directory structure.....	42
Table 14: Source Modules USB Serial Conversation Sample	53
Table 15: Header Files USB Serial Conversation Sample.....	53
Table 16: Device Descriptor Serial Conversion Sample	53
Table 17: Configuration Descriptor Serial Conversion Sample.....	54
Table 18: Interface Descriptor Serial Conversion Sample	54
Table 19: Endpoint Descriptor Serial Conversion Sample.....	55
Table 20: String Descriptor Serial Conversion Sample.....	55
Table 21: Source Modules USB Mass Storage Sample	64
Table 22: Header Files USB Mass Storage Sample	64
Table 23: Device Descriptor Mass Storage Sample	64
Table 24: Configuration Descriptor Mass Storage Sample	65
Table 25: Interface Descriptor Mass Storage Sample	65
Table 26: Endpoint Descriptor Serial Conversion Sample.....	66
Table 27: String Descriptor Mass Storage Sample	66

1. Introduction

V850ES/Jx3-H - Plug it! is an USB-demonstration kit for the NEC V850 32-bit microcontroller family. It allows the development of an USB system based on the V850 70F3760 device. It supports onboard debugging and real time execution of application programs. The board is prepared to be connected to user hardware parts such as digital I/O or analog signals.

1.1 Main features of *V850ES/Jx3-H - Plug it!*

- Easy to use device demonstration capabilities
V850ES/Jx3-H - Plug it! contains elements to easily demonstrate simple I/O-functions, i.e. I/O lines, analog inputs and outputs, UART serial interface, USB interface etc.
- On-Board debug function
The *V850ES/Jx3-H - Plug it!* supports an On-Board debug function by using the IAR C-SPY debugger without a need of additional debug hardware. It allows FLASH downloading and standard debug functions like code execution, single stepping, breakpoints, memory manipulation etc.
- Power supply by USB interface or via external power supply
- Analog to digital signal conversion
- Various input / output signals available, such as
 - I/O ports prepared to be connected to user hardware
 - Timer input / output signals
 - Two or three wire serial I/O
 - 12 analog input lines
 - 2 analog output lines
 - Two Switches prepared for key interrupt generation
- The IAR Embedded Workbench for V850 and the IAR C-SPY debugger / simulator are included. These packages are restricted in such that maximum program code size is limited to 16 KB.
- Full documentation is included for the NEC V850 70F3760 microcontroller, IAR Systems Embedded Workbench and IAR Systems C-SPY debugger / simulator.

System requirements

HOST PC	A PC supporting Windows 2000, Windows XP or Windows Vista is required for the IAR Systems Embedded Workbench demo-version. A Pentium processor with at least 1 GHz CPU performance, with at least 256 Mbytes of RAM, allowing you to fully utilize and take advantage of the product features. 500 Mbytes of free disk space and an additional 10 Mbytes of free disk space on the Windows system drive.
Host interface	A web browser and Adobe Acrobat Reader to be able to access all the product documentation. USB interface that enables communication based on USB (Ver1.1 or later)

1.2 Package contents

Please verify that you have received all parts listed in the package contents list attached to the *V850ES/Jx3-H - Plug it!* package. If any part is missing or seems to be damaged, please contact the dealer from whom you received your *V850ES/Jx3-H - Plug it!*.

Note: Updates of the IAR Embedded Workbench for V850 documentation and/or utilities for *V850ES/Jx3-H - Plug it!*, available, may be downloaded from the NEC WEB page(s) at <http://www.eu.necel.com/update/index.html?id=287>

1.3 Trademarks

IAR Embedded Workbench, visualSTATE, IAR MakeApp and C-SPY are registered trademarks of IAR Systems AB. Microsoft and Windows are registered trademarks of Microsoft Corporation. Adobe and Acrobat Reader are registered trademarks of Adobe Systems Incorporated.

All other product names are trademarks or registered trademarks of their respective owners.

2. V850ES/Jx3-H - Plug it! system configuration

The V850ES/Jx3-H - Plug it! system configuration is given in the diagram below:

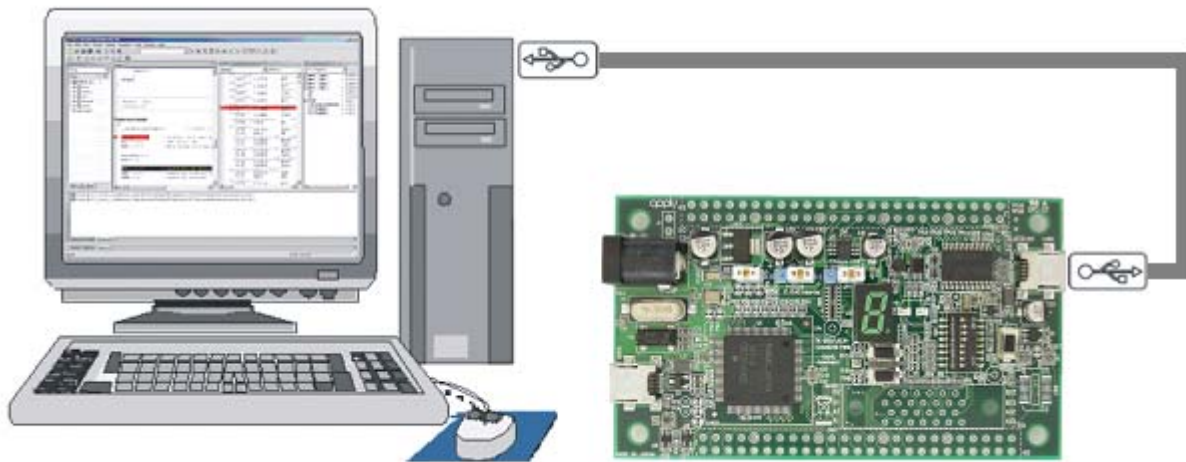


Figure 1: V850ES/Jx3-H - Plug it! system configuration

2.1 V850ES/Jx3-H - Plug it!

V850ES/Jx3-H - Plug it! is a USB-demonstration kit for the 70F3760 32-bit microcontroller of the V850 family. The demonstration board is connected to the host system via USB interface cable. The host system may be used for On-Chip debugging by using the IAR C-SPY debugger and to allow execution of application programs on the V850ES/Jx3-H - Plug it! starterkit.

2.2 Host computer

The USB host interface enables communication to the V850ES/Jx3-H - Plug it! board. The μ PD78F0730 78K0 8-Bit microcontroller with on-chip USB interface and the NEC virtual UART driver allows application software to access the USB device in the same way as it would access a standard RS232 interface. The NEC virtual UART driver appears to the windows system as an extra Com Port, in addition to any existing hardware Com Ports.

2.3 Power supply via USB interface

The V850ES/Jx3-H - Plug it! board is powered by the USB interface. Optional the power supply can be applied via the connectors CN4 or J1.

3. V850ES/Jx3-H - Plug it! components

The V850ES/Jx3-H - Plug it! board is equipped with USB-connector and with several connectors in order to be connected to host computers, FLASH programmer or any external target hardware.

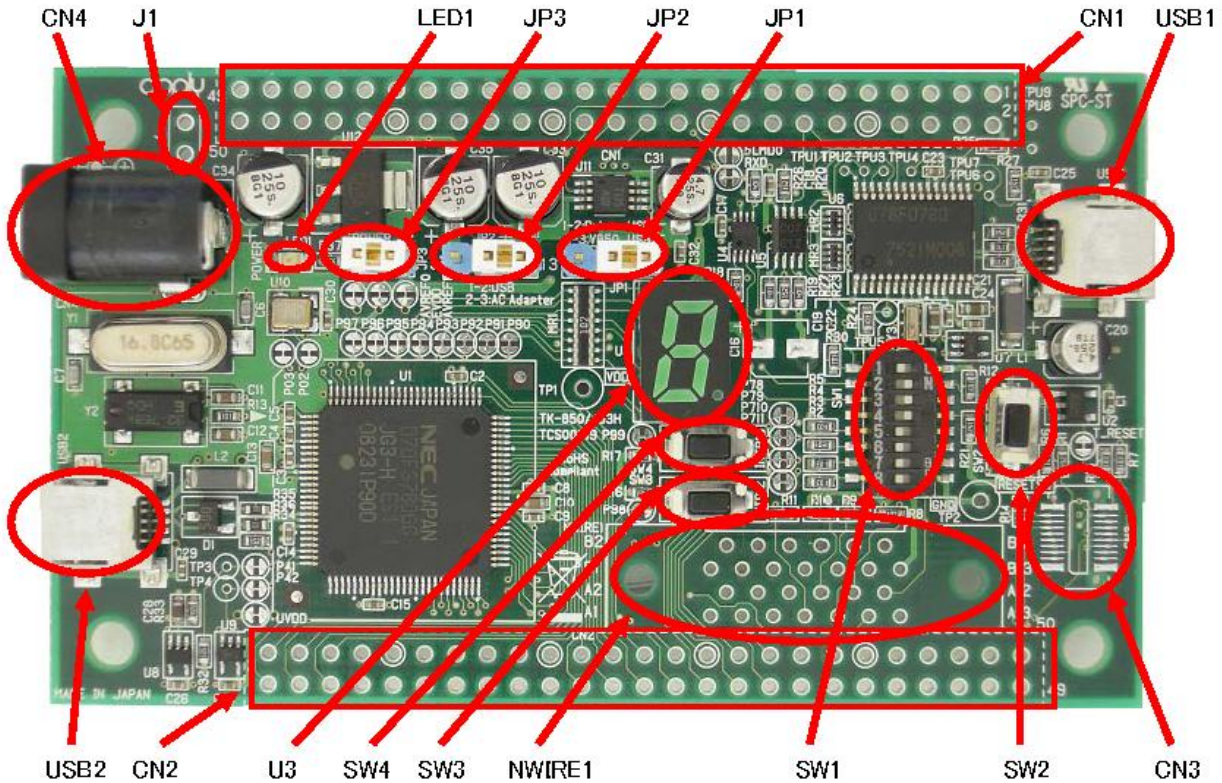


Figure 2: V850ES/Jx3-H - Plug it! Components

Some of the V850ES/Jx3-H - Plug it! components are free for user application hardware and software. Please read the user’s manual of the 70F3760 device carefully to get information about the electrical specification of the available I/O ports before you connect any external signals to the V850ES/Jx3-H - Plug it! board.

3.1 SW1, Configuration Switch (DIP-Switch)

The different operation modes of the *V850ES/Jx3-H - Plug it!* board can be set by switch SW1. The bits 1-4 of DIP switch SW1 are for the mode setting of the board, bits 5-8 are connected to the pins “P78/ANI8”, “P79/ANI9”, “P710/ANI10” and “P711/ANI11” of the V850 microcontroller and can be used for user application purpose.

The *V850ES/Jx3-H - Plug it!* starterkit can be used in the following operation modes:

- On-Board debug mode
 - o Start a debug session using the On-Board debug interface
- Flash-Programming Mode
 - o Program an application to the build-in flash memory of 70F3760 by WriteEZ1-programmer
- Stand alone mode
 - o Run a program stored in built-in flash memory of the 70F3760 device
- N-Wire debug mode
 - o Start a debug session using an N-Wire emulator, e.g. MINICUBE

SW1 / bit	On-Board Debug Mode	Flash Programming Mode	Stand alone Mode	N-Wire Debug Mode
1	OFF	OFF	OFF	OFF
2	ON	ON	OFF	OFF
3	ON	ON	OFF	OFF
4	ON	ON	OFF	OFF

Table 1: Mode setting, switch SW1

Note: After changing the configuration of SW1 bits 1-4 it is necessary to power-up the *V850ES/Jx3-H - Plug it!* board to make changing active. This can be done by simply dis- and reconnecting the USB interface cable.

3.2 SW2, RESET button

SW2 is the reset button. It activates the power on reset. Switch SW2 controls the reset input signal of the V850ES/Jx3-H - Plug it! microcontroller.

3.3 SW3, Switch (INTP12)

SW3 is a push button connecting VSS to external interrupt input INTP12 of the microcontroller. This is equal to port "P98/INTP12" of the 70F3760 device. The port may be programmed to generate the external interrupt INTP12. The necessary initialization for this purpose is described in the user's manual of the 70F3760 device.

3.4 SW4, Switch (INTP13)

SW4 is a push button connecting VSS to external interrupt input INTP13 of the microcontroller. This is equal to port "P99/INTP13" of the 70F3760 device. The port may be programmed to generate the external interrupt INTP13. The necessary initialization for this purpose is described in the user's manual of the 70F3760 device.

3.5 JP1 and JP2, Power Supply selectors

Jumper JP1 and JP2 are the power supply selectors of the V850ES/Jx3-H - Plug it! board.

JP2	JP1	Power Supply Source
1-2 short	1-2 short	USB connector USB1
	2-3 short	USB connector USB2
2-3 short	-	AC adapter via connector CN4 or J1

Table 2: Power supply selectors, JP1 and JP2

Note: If multiple connections made to USB1, USB2 and CN4/J1, set JP1 to the preferred power supply route.

3.6 JP3, Power LED

Jumper JP3 enables or disables power LED1

JP2	Configuration	Power LED1
1-2	Closed	Enabled
1-2	Opened	Disabled

Table 3: Power LED1 setting

3.7 LED1, power LED

LED1 is the power LED of the *V850ES/Jx3-H - Plug it!* board. It indicates if power is applied to the *V850ES/Jx3-H - Plug it!* board.

3.8 CN4, AC power supply connector

CN4 is the AC power supply connector of the *V850ES/Jx3-H - Plug it!* board. Please connect only a power supply of maximum +12V AC to the board.

JACK1	Input
Centre	VDD (+12V)
Ring	GND

Table 4: CN4 connector

3.9 J1, optional power supply connector

J1 (not assembled) is an optional connector for an external power supply.

3.10 NWIRE1, N-Wire connector

Connector NWIRE1 (not assembled) allows connecting to an N-Wire On-Chip debug emulator, e.g. MINICUBE, to *V850ES/Jx3-H - Plug it!* board. Please note, the QB-V850MINI On-Chip debug emulator is a separate product from NEC and it is not included in this starterkit package.

3.11 U3, Seven-segment-LED

The seven-segment-LED U3 is controlled by port P90 to P97. Please set port mode to output and output a low signal to light the corresponding segment.

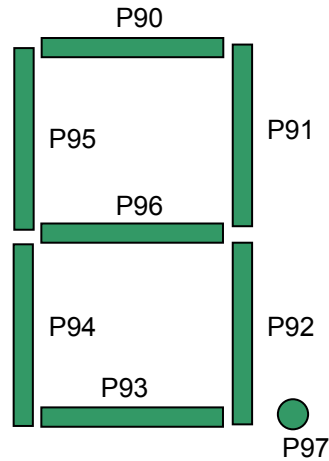


Figure 3: Seven-segment-LED U3

To display the characters '0' to '9' please write the following values to register P9:

Character	P9 value
0	0xC0
1	0xF9
2	0xA4
3	0xB0
4	0x99
5	0x92
6	0x83
7	0xf8
8	0x80
9	0x98

Table 5: Display Examples

3.12 USB1, serial interface connector

This interface allows connecting the IAR C-SPY debugger to the *V850ES/Jx3-H - Plug it!* board in order to use the On-Board debug function. The interface supports On-board FLASH erasing / programming and standard debug features like code execution, single stepping, breakpoints, memory manipulation etc.

The power supply of the *V850ES/Jx3-H - Plug it!* board is also provided by the USB1 connector.

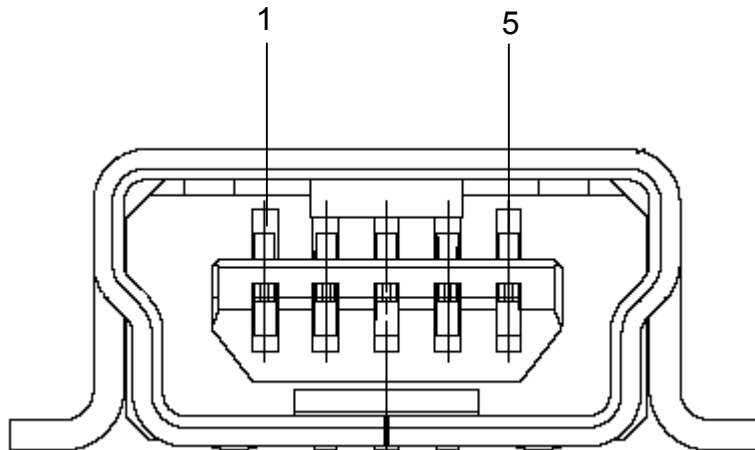


Figure 4: USB1, USB Mini-B Type Host Connector Pin Configuration

Connector USB1	Signal Name
1	VBUS
2	D-
3	D+
4	ID_NC
5	GND

Table 6: Pin Configuration of Connector USB1

For connection with the host machine, use a USB cable (Mini-B type). For confirmation, NEC Electronics used only the USB cable delivered with the *V850ES/Jx3-H - Plug it!* board.

3.13 USB2, universal serial bus connector

The interface allows connecting the built-in USB function controller of the μ PD70F3760 with a host system. It is also possible to provide the power supply of the *V850ES/Jx3-H - Plug it!* board by the USB2 connector.

For connection with the host machine, use a USB cable (Mini-B type). For confirmation, NEC Electronics used only the USB cable delivered with the *V850ES/Jx3-H - Plug it!* board.

3.14 Layout of solder-short pads

Several pins of the 70F3760 microcontroller are connected to solder short-pads. The pads can be opened by the user to add user specific functions. The signal connected to each solder-short pad is printed on the *V850ES/Jx3-H - Plug it!* board. To open a circuit, cut the narrow part of the pad with a knife. To short a circuit again, join the separated pad with a soldering iron.



Figure 5: Solder-short pad opened shape



Figure 6: Solder-short pad shortened shape

Solder-short pad name	Shipping state	Connection
P90-P97	Short	connection to seven-segment LED Open pads if port pins shall have other usage.
FLMD0	Open	FLMD0 to CPU port P37 Shorten pad if flash self programming is used.
EVDD	Short	EVDD to VDD Open if EVDD is driven by other voltage.
AVREF0	Short	VDD to AVREF0 Open when AVREF0 is driven by other voltages.
AVREF1	Short	VDD to AVREF1 Open when AVREF1 is driven by other voltages.
UVDD	Short	VDD to UVDD Open when UVDD is driven by other voltages.
RXD	Short	P30/TXDC0 Open when P30 isn't used for USB1 communication interface.
P78-P711	Short	SW1, bits 5-8 Open when P78-P711 isn't used for general-purpose input.
P02	Short	P02 to standby pin of 48MHz oscillator (U10) Open when on-board 48MHz oscillator isn't used as USB clock.
P03	Short	P03 to output pin of 48MHz oscillator (U10) Open when on-board 48MHz oscillator isn't used as USB clock.
P98	Short	P98 to SW3 Open when P98 is used for other purposes.
P99	Short	P99 to SW4 Open when P99 is used for other purposes.
P41	Short	P41 to USB D+ pull-up signal enable pin. Open when you don't use USB D+ pull-up enable.
P42	Short	P42 to USB D+ pull-up signal pin. Open when you don't use USB D+ pull up signal.
T_RESET	Short	U2 Pin4 to T_RESET Open when the on-board Reset Circuit isn't used.

Table 7: Solder-short pad connection

4. On-Chip debugging

The *V850ES/Jx3-H - Plug it!* board offers two possibilities to use On-Chip debugging (OCD). The TK-V850 On-Board debug function of *V850ES/Jx3-H - Plug it!* allows On-Chip debugging without a need of external debug hardware. Within this mode the default USB connection to the Host computer based on the virtual UART driver is used as debug interface. All standard debug functions are available in the On-Board debugging mode like FLASH programming / downloading, code execution, single stepping, breakpoints, memory manipulation etc.

Additionally *V850ES/Jx3-H - Plug it!* supports the QB-V850MINI On-Chip debug emulator in order to use On-Chip debug function of the 70F3760 device. The system configuration for On-Chip debugging is shown in figure below.

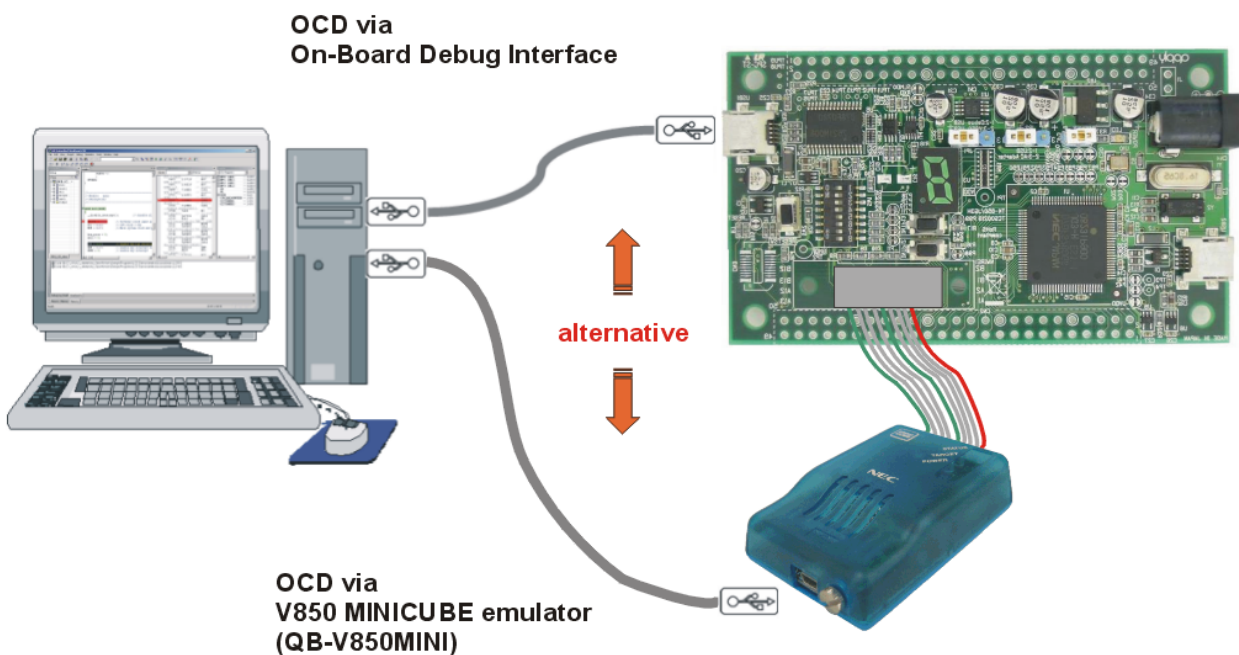


Figure 7: On-Chip debugging

4.1 OCD via TK-V850 On-Board debug function

To operate the *V850ES/Jx3-H - Plug it!* board within the On-Board debug mode, configure switch SW1 bits 1-4 as following:

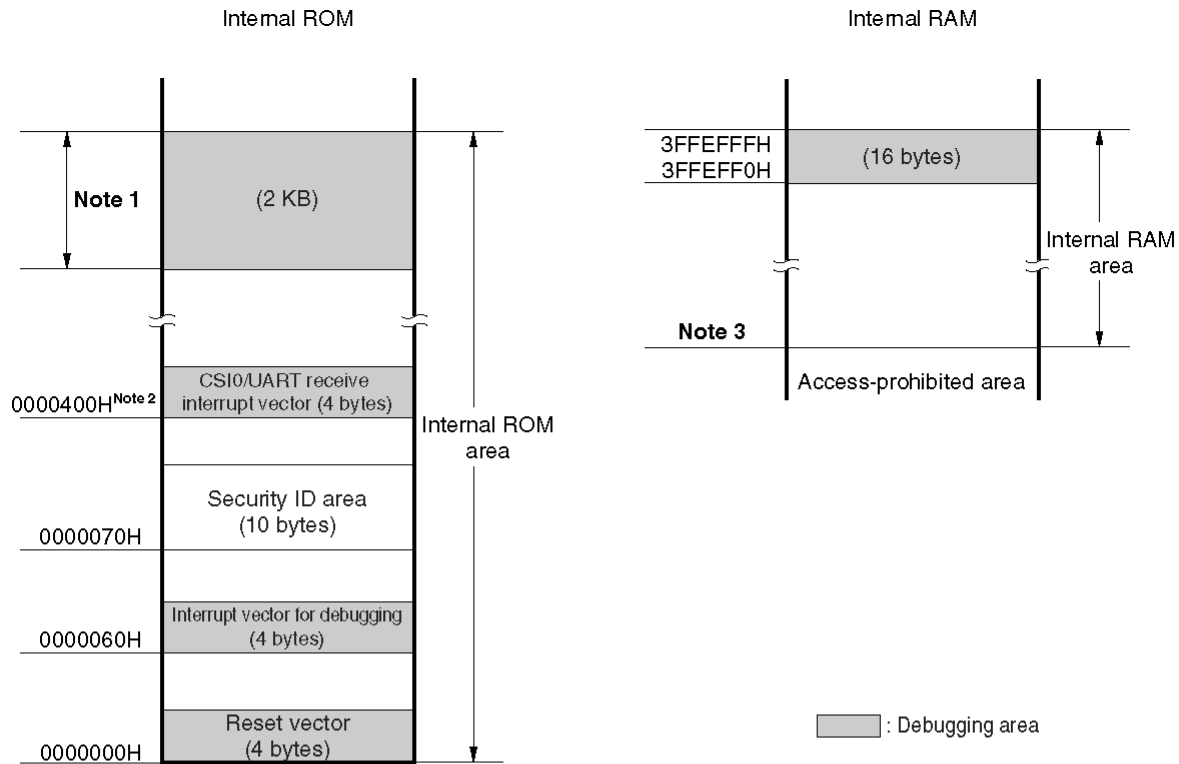
SW1/bit	Configuration
1	OFF
2	ON
3	ON
4	ON

Table 8: OCD via TK-78K0R On-Board debug function

4.1.1 Resources used by TK-V850 OCD Interface

Debugging via the On-board TK-V850 OCD Interface uses the user memory spaces (shaded portions in Figure 8) to implement communication with the target device, or each debug functions. Refer to the following descriptions and secure these spaces in the user program.

- addresses 0x00000060 – 0x00000063: Interrupt vector of debug monitor
This area is automatically reserved.
It is not allowed to use this area for any application segment.
- addresses 0x000004A0 - 0x000004A3: Interrupt vector of UARTC0
UARTC0 is used for the communication with the debugger
and must be reserved
It is not allowed to use this area for any application segment.
- addresses 0x00000070-0x00000079 Security ID
The security ID must be defined to configure the OCD Interface
Details are described in the μ PD70F3760 user's manual.
- addresses 0x0003F800 – 0x0003FFFF: 2K Debug Monitor area (ROM)
This area must be reserved in the linker control file (*.xcl).
It is not allowed to use this area for any application segment.
- addresses 0x3FFEFF0 – 0x3FFEFFFF: 16 Bytes Debug Monitor area (RAM)
This area must be reserved in the linker control file (*.xcl).
It is not allowed to use this area for any application segment.



Notes 1. Address values vary depending on the product.

	Internal ROM Size	Address Value
μ PD70F3760, 70F3765, 70F3770, 70F3771	256 KB	003F800H to 005FFFFH
μ PD70F3761, 70F3766	384 KB	005F800H to 005FFFFH
μ PD70F3762, 70F3767	512 KB	007F800H to 007FFFFH

- 2. This is the address when CSIF0 is used. It starts at 0000406H when CSIF3 is used, and at 00004A0H when UARTC0 is used.
- 3. Address values vary depending on the product.

	Internal RAM Size	Address Value
μ PD70F3760, 70F3765, 70F3770, 70F3771	32 KB	3FF7000H
μ PD70F3761, 70F3766	40 KB	3FF5000H
μ PD70F3762, 70F3767	48 KB	3FF3000H

Figure 8: V850 OCD Interface Resources

4.2 OCD via N-Wire emulator

To operate the *V850ES/Jx3-H - Plug it!* board together with an N-Wire On-Chip debug emulator, e.g. QB-V850MINI, configure switch SW1 bits1-4 as following:

SW1/bit	Configuration
1	OFF
2	OFF
3	OFF
4	OFF

Table 9: OCD via QB-V850MINI emulator

5. V850ES/Jx3-H - Plug it! installation and operation

5.1 Getting started

The IAR Embedded Workbench including the C-SPY debugger allows building and downloading application programs to the *V850ES/Jx3-H - Plug it!* starterkit. As communication interface between the PC host system and the *V850ES/Jx3-H - Plug it!* board a standard USB interface line is needed. Before you can download and run a program, software and hardware have to be installed properly.

5.1.1 CD-ROM contents

The CD-ROM shows following directory structure:








NEC V850ES/Jx3-H - Plug it!	CD-ROM ROOT
 Acrobat	- Acrobat Reader for 32Bit Windows OS
 Device File Package	- Device File Package for V850ES/Jx3-H
 Doc	- Documentation
 IAR	- IAR Embedded Workbench for V850
 SamplePrograms	- Sample programs for V850ES/Jx3-H - Plug it! <i>including:</i> <ul style="list-style-type: none"> ○ <i>Jx3H_PLUGIT_DEMO sample</i> ○ <i>Jx3H_PLUGIT_COM sample</i> ○ <i>Jx3H_PLUGIT_STORAGE sample</i>
 Segger	- <i>Segger USB Eval Package</i>
 WriteEZ1	- Flash Programmer WriteEZ1 incl. PRM files for μ PD70F3760

Table 10: V850ES/Jx3-H - Plug it! CD-ROM directory structure

6. Hardware installation

After unpacking *V850ES/Jx3H - Plug it!*, connect the board via connector USB1 to your host computer using the provided USB interface cable. When *V850ES/Jx3H - Plug it!* is connected, the USB driver needs to be installed on the host machine. Please refer to the following [chapter 7.5 USB Driver Installation](#).

7. Software installation

The *V850ES/Jx3H - Plug it!* package comes with the following software demo packages:

- IAR Systems Embedded Workbench for V850, including C compiler, assembler, linker, librarian and IAR C-SPY debugger / simulator
- Sample programs
- Segger USB Eval Package

The IAR Systems Embedded Workbench must be installed on your PC. For detailed installation hints, refer to the following chapters and to the corresponding documentation of the IAR Embedded Workbench.

7.1 IAR Systems Embedded Workbench for V850 installation

To install the IAR Systems Embedded Workbench for V850 including C-SPY debugger / simulator, select the `AUTORUN` program in the directory `\IAR\` of the CDROM. The setup dialogues will guide you through the installation process.

7.2 Device File Package installation

To use the IAR Embedded Workbench for V850 together with the `uPD70F3760`, you have to install the latest device files. You will find the installer for the Device file package on the *V850ES/Jx3H - Plug it!* CD. Select the file `Setup_V850ES-JX3-H_V1.00.exe` in the directory `\Device File Package\`. The setup dialogues will guide you through the installation process.

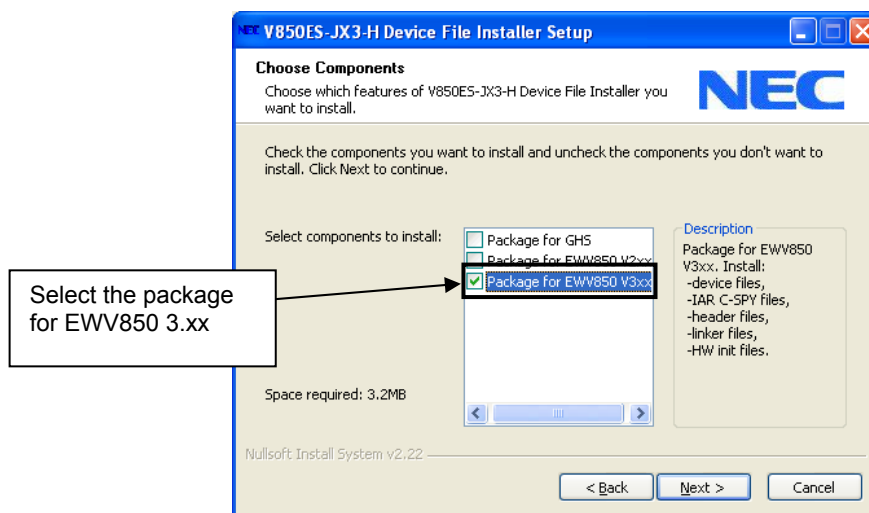


Figure 9: Device File Installation

7.3 Sample program installation

To install the sample/demonstration programs for the *V850ES/Jx3H - Plug it!* board select the `Jx3H-PLUGIT_Samples_V100.exe` in the directory `\SamplePrograms\` on the CDROM. The setup dialogues will guide you through the installation process.

7.4 Segger USB Eval Package installation

To install the Segger USB Eval Package for the *V850ES/Jx3-H - Plug it!* board select the `SeggerEval_Jx3H_PLUGIT.exe` in the directory `\Segger\` on the CDROM. The setup dialogues will guide you through the installation process. After the installation you will find a detailed documentation of the Segger USB Eval Package in the installation directory.

7.5 USB Driver Installation

In order to use the *V850ES/Jx3-H - Plug it!* board for On-Chip debugging the USB driver needs to be installed on the host machine. Install the driver according to the following procedure:

- Installation on Windows 2000 Page 26
- Installation on Windows XP Page 30

Note: The USB driver is part of the IAR Embedded Workbench software package. Therefore please install the IAR Embedded Workbench first.

7.5.1 Installation on Windows 2000

1. When the *V850ES/Jx3-H - Plug it!* board is connected with the host machine, the board is recognized by <Plug and Play>, and the wizard for finding new hardware is started. Click Next>.



Figure 10: Found New Hardware Wizard (Windows 2000)

- Following the window below is displayed. So, check that "Search for a suitable driver ..." is selected, then click **Next>**.

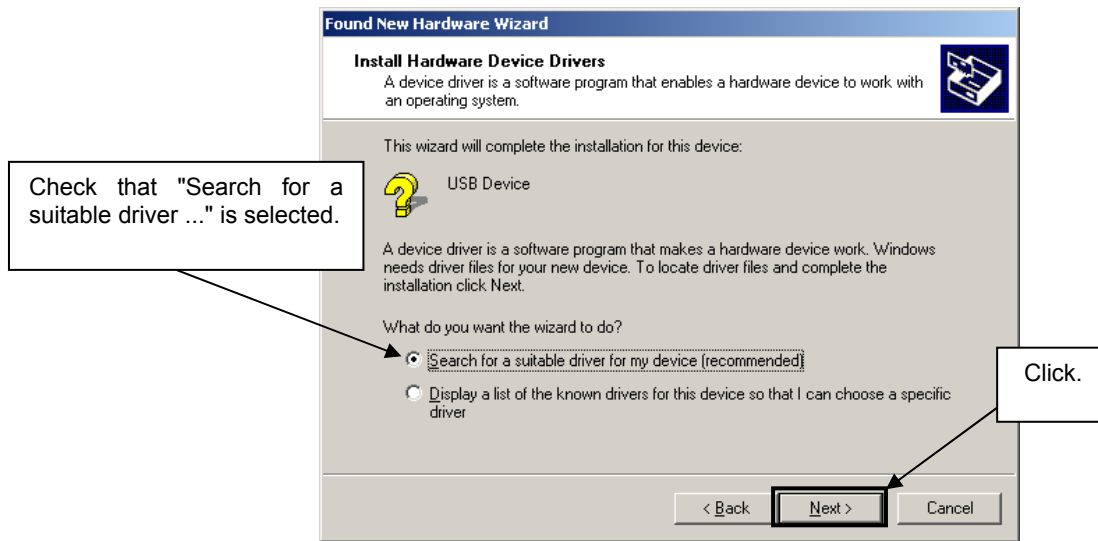


Figure 11: Search Method (Windows 2000)

- Check the "Specify a location" check box only, then click **Next>**.

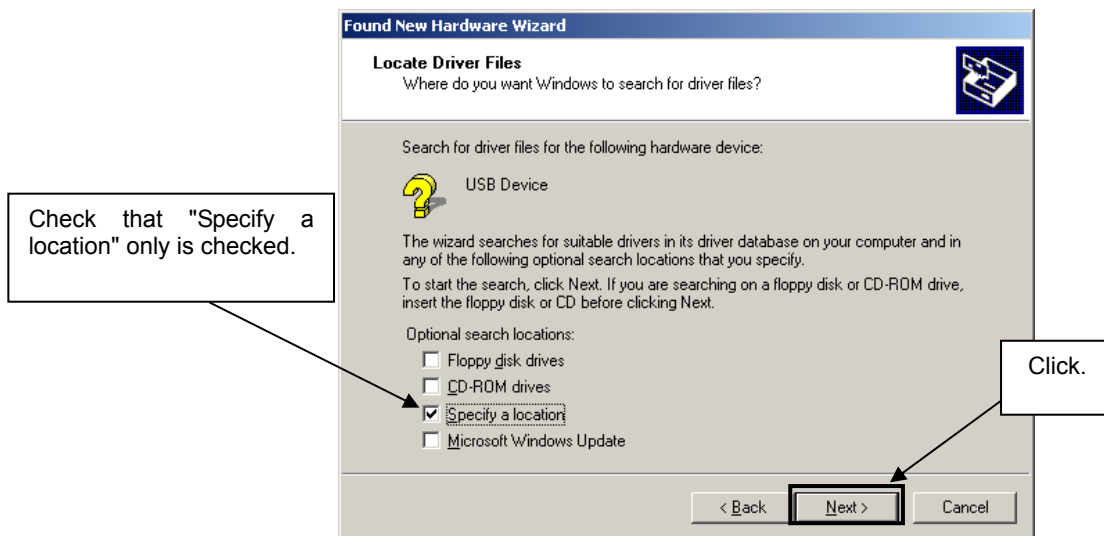


Figure 12: Driver File Location (Windows 2000)

4. Locate the folder " \Driver" on the CD-ROM.
5. The setup information file "MQB2SALL.inf" is automatic selected, then click **Open** to proceed within driver installation.

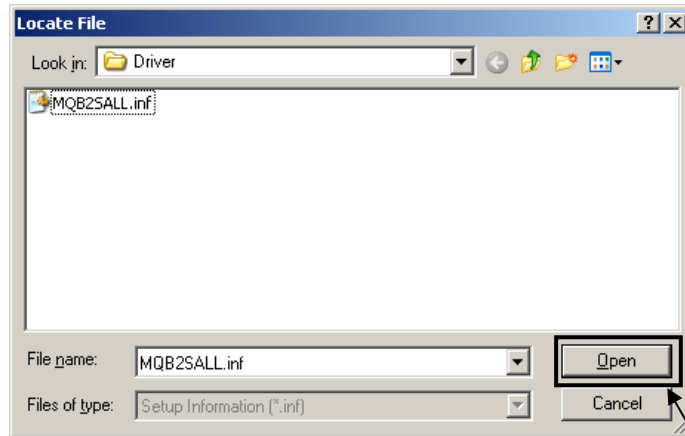


Figure 13: Address Specification 2 (Windows 2000)

Click.

6. After the location of the USB driver has been specified click **OK** to proceed.
5. Click **Next>**.

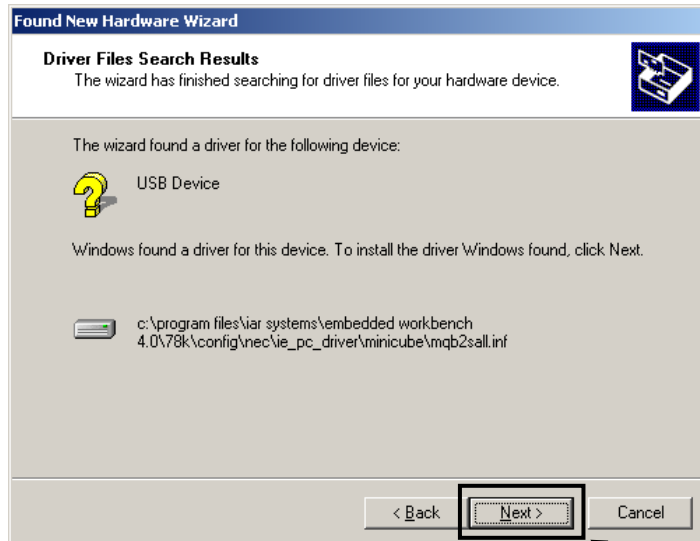


Figure 14: Driver File Search (Windows 2000)

Click.

6. Click **Finish** to complete the installation of the USB driver.

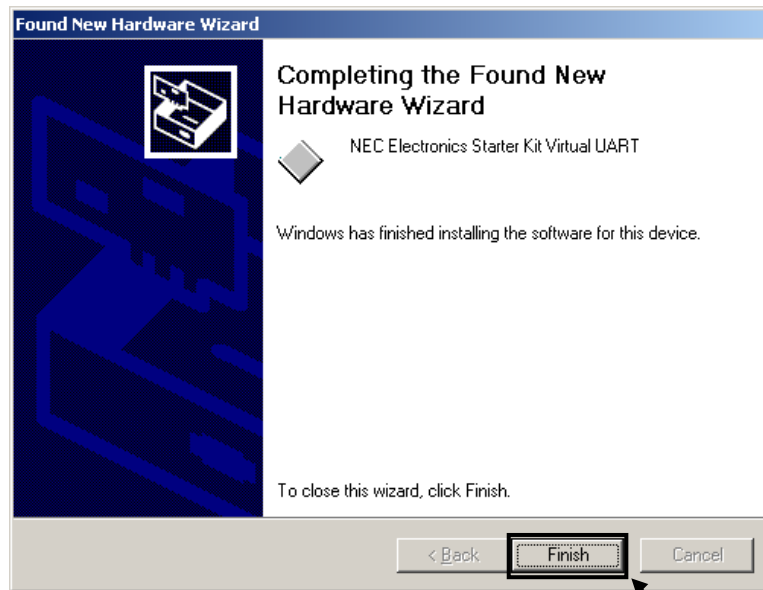


Figure 15: USB Driver Installation Completion (Windows 2000)

Click.

7.5.2 Installation on Windows XP

1. When the V850ES/Jx3-H - Plug it! board is connected with the host machine, the board is recognized by Plug and Play, and the wizard for finding new hardware is started. At first the hardware wizard will ask if windows should search on the windows update web, check "No, not this time" and then click **Next>**.



Figure 16: Found New Hardware Wizard 1 (Windows XP)

2. Check that "Install from a list or specific location (Advanced)" is selected, then click **Next>**.

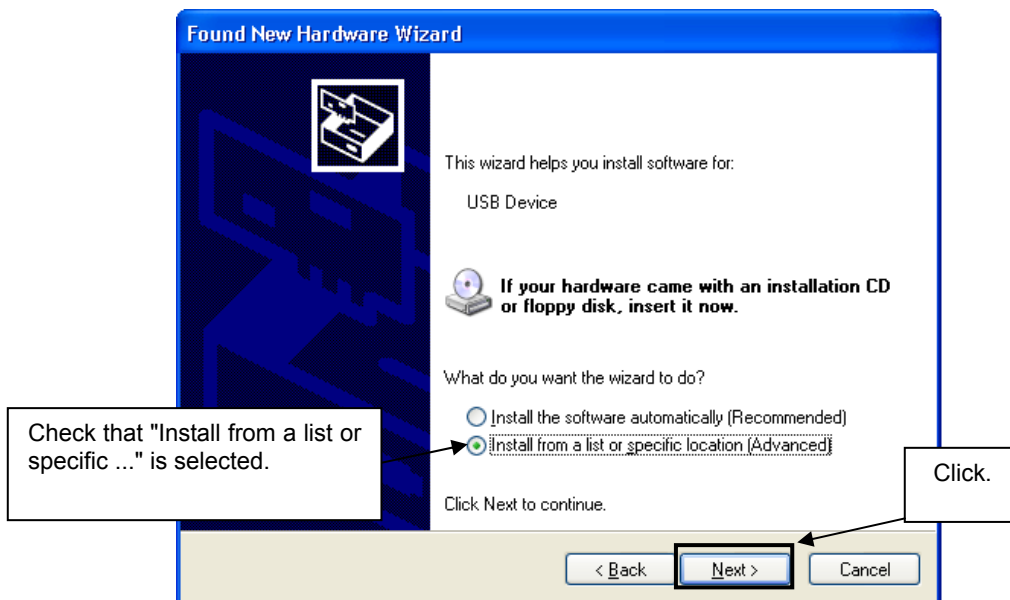


Figure 17: Found New Hardware Wizard 2 (Windows XP)

3. Check that "Search for the best driver in these locations." is selected. Select the "Include this location in the search:" check box and then click **Browse**.

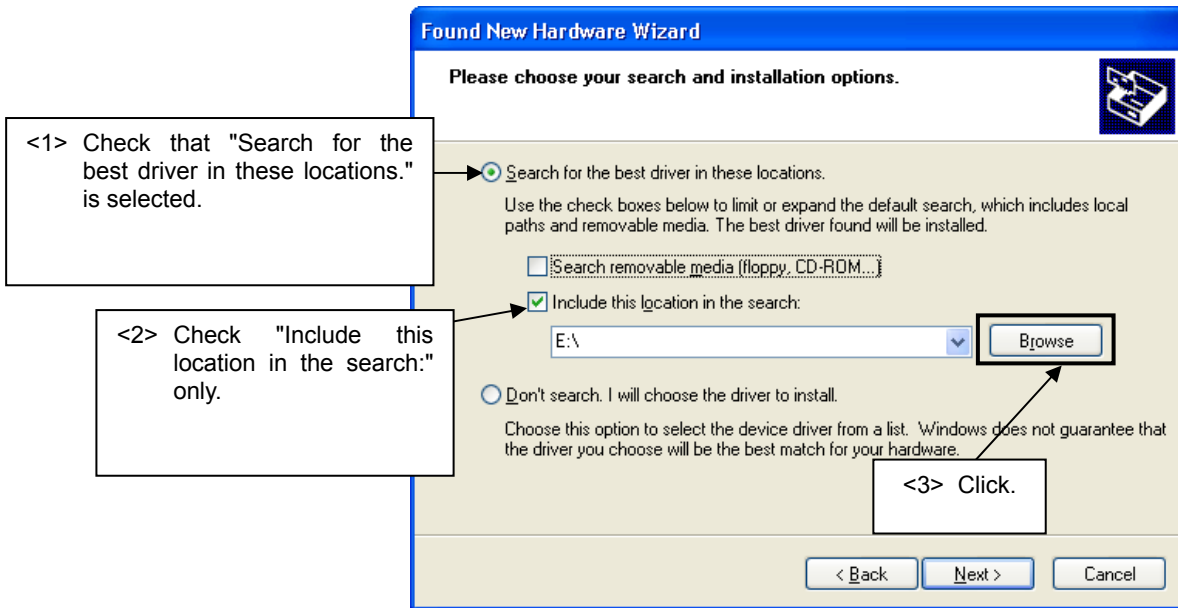


Figure 18: Search Location Specification 1 (Windows XP)

4. Locate the folder "\Driver" on the CD-ROM.
5. After the location of the USB driver has been specified click **Next>** to continue driver installation.

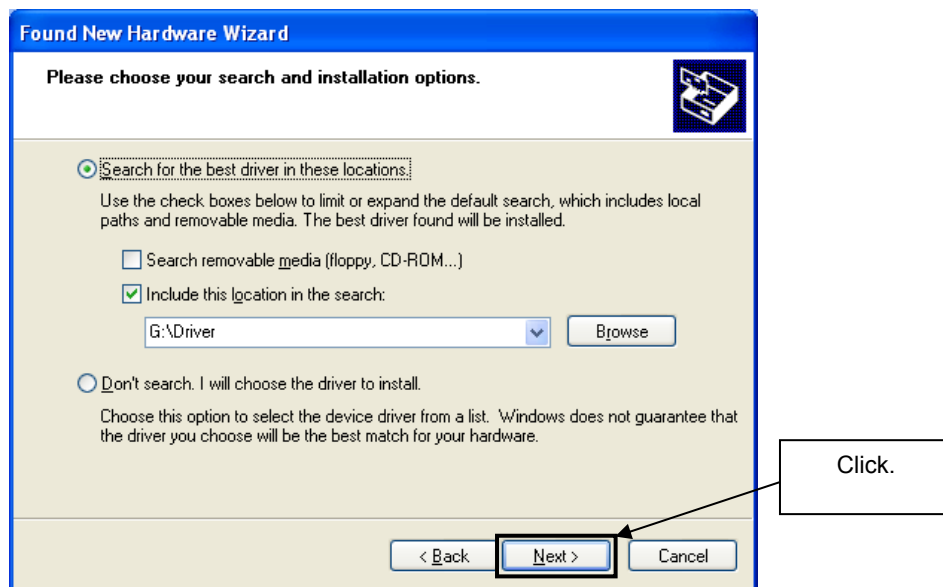


Figure 19: Search Location Specification 3 (Windows XP)

- As shown below, "NEC Electronics Starter Kit Virtual UART has not passed Windows Logo testing to verify its compatibility with Windows XP." is displayed. Click **Continue Anyway**.



Figure 20: Windows XP Logo Testing (Windows XP)

- After the installation of the USB driver is completed the window below is displayed. Click **Finish** to close the hardware wizard.

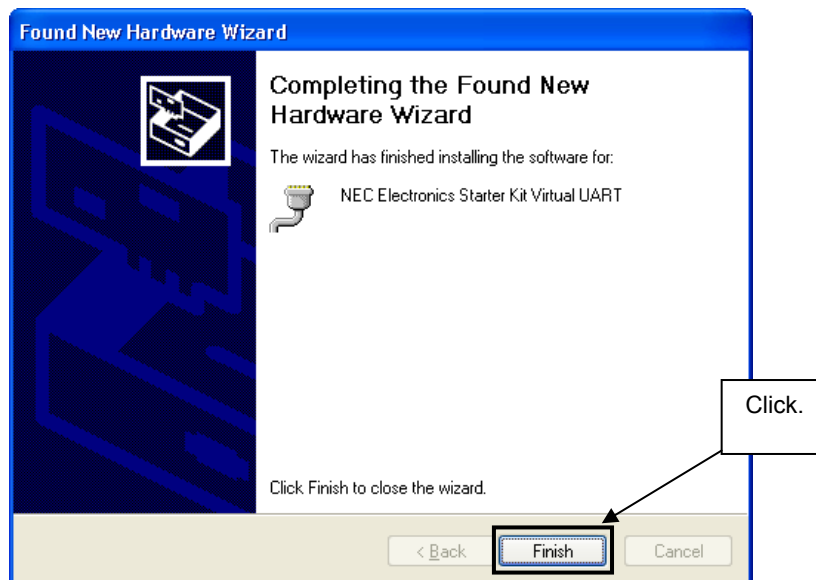


Figure 21: USB Driver Installation Completion (Windows XP)

7.6 Confirmation of USB Driver Installation

After installing the USB driver, check that the driver has been installed normally, according to the procedure below. When using the *V850ES/Jx3-H - Plug it!* board in combination with IAR C-SPY debugger the “NEC Electronics Starter Kit Virtual UART” should be present like in the figure below.

Please check in the Windows “Device Manager” within the Windows Properties (“Hardware” tab), that the driver is installed normally.

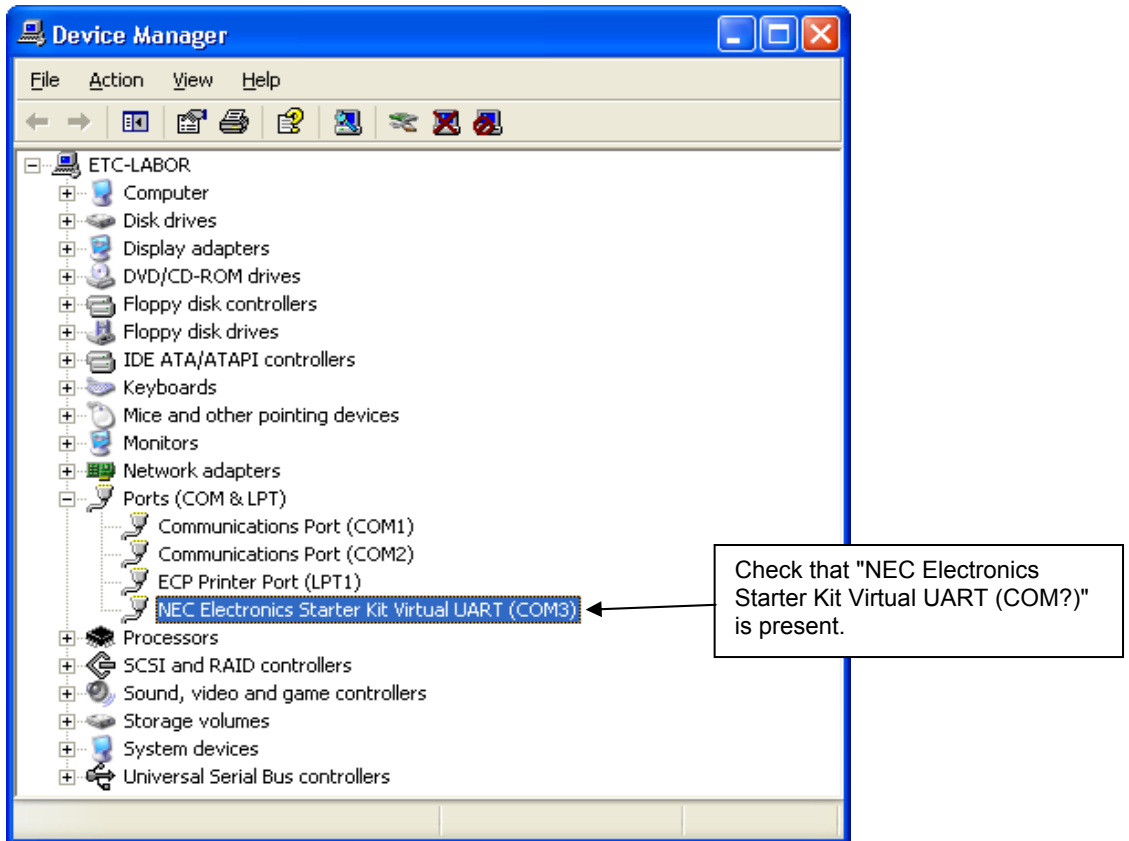


Figure 22: Windows Device Manager

8. IAR sample session

When everything is set up correctly the IAR Embedded Workbench can be started. To do so, start the Embedded Workbench from Windows “Start” menu > “Programs” > folder “IAR Systems” > “IAR Embedded Workbench Kickstart for NEC V850”. The following screen appears:

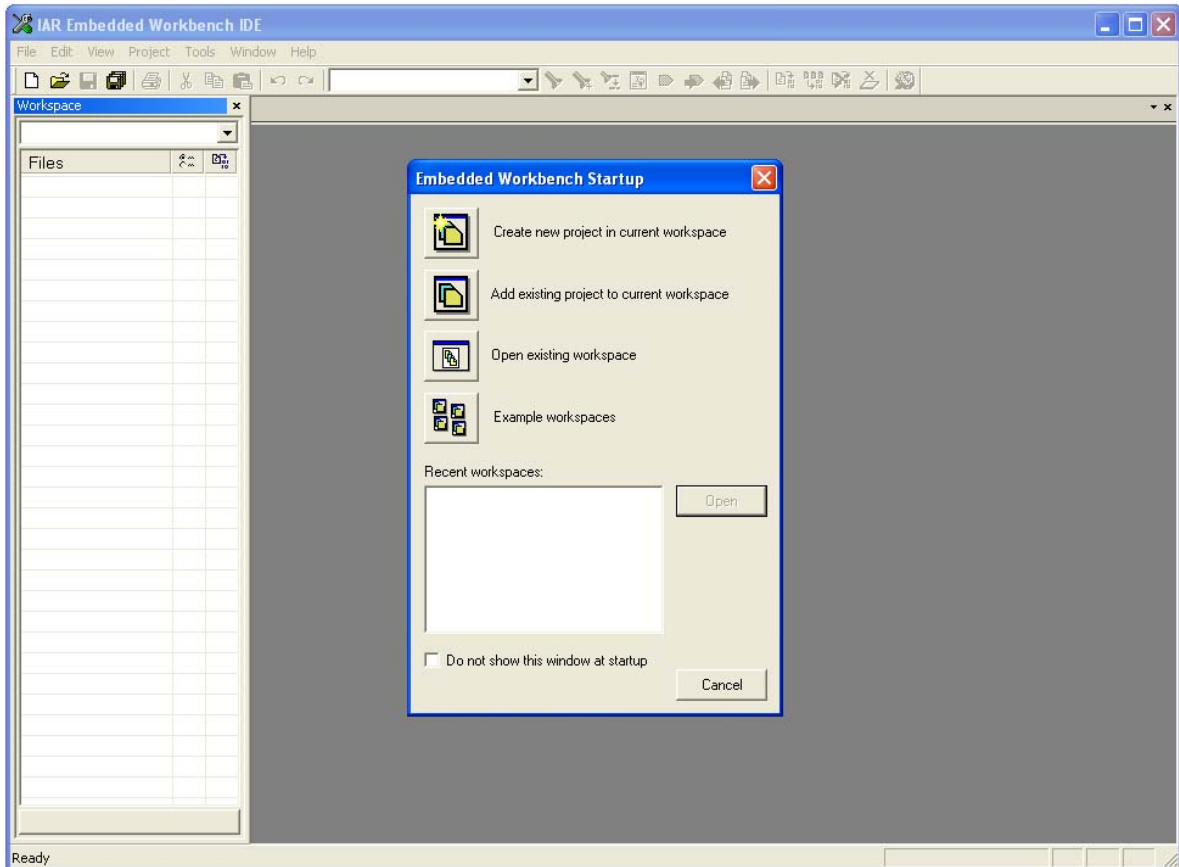


Figure 23: IAR Embedded Workbench

Now select the option “Open exiting workspace” from the “File” menu and locate the sample project folder and open the file “Jx3-H_PLUGIT.eww”. This is the workspace file that contains general information about all sample projects and corresponding settings.

After the sample workspace has been opened the projects included in the workspace are displayed. Please select the sample project “Jx3-H_PLUGIT_DEMO”. The screen should now look similar to this one:

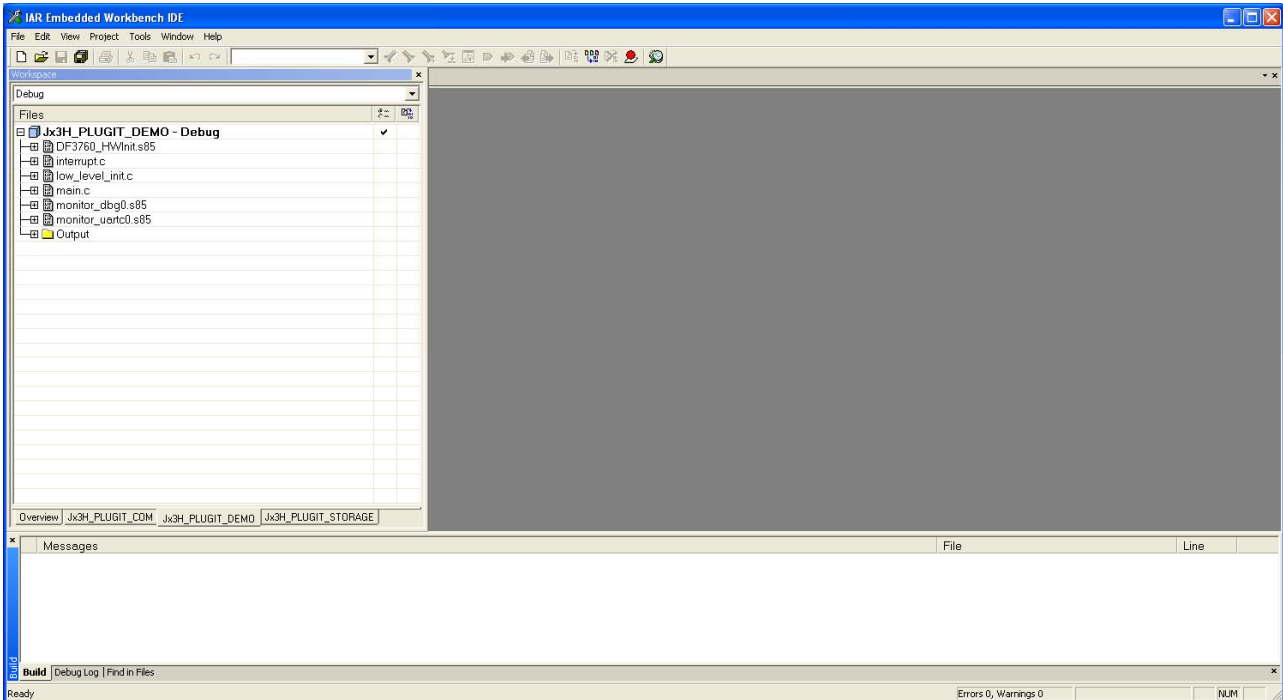


Figure 24: IAR Embedded Workbench Project Workspace

As a next step check some settings of the IAR Embedded Workbench that have to be made for correct operation of the demonstration samples and usage of the On-Board debug function of the V850ES/Jx3-H - Plug it! board. First highlight the upper project folder called “Jx3-H_PLUGIT_DEMO – Debug” in the workspace window. Then select “Project” > “Options” from the pull-down menus. Go to “General Options” and select the device μ PD70F3760 (V850ES \rightarrow V850ES – Jx3-H \rightarrow V850ES - μ PD70F3760).

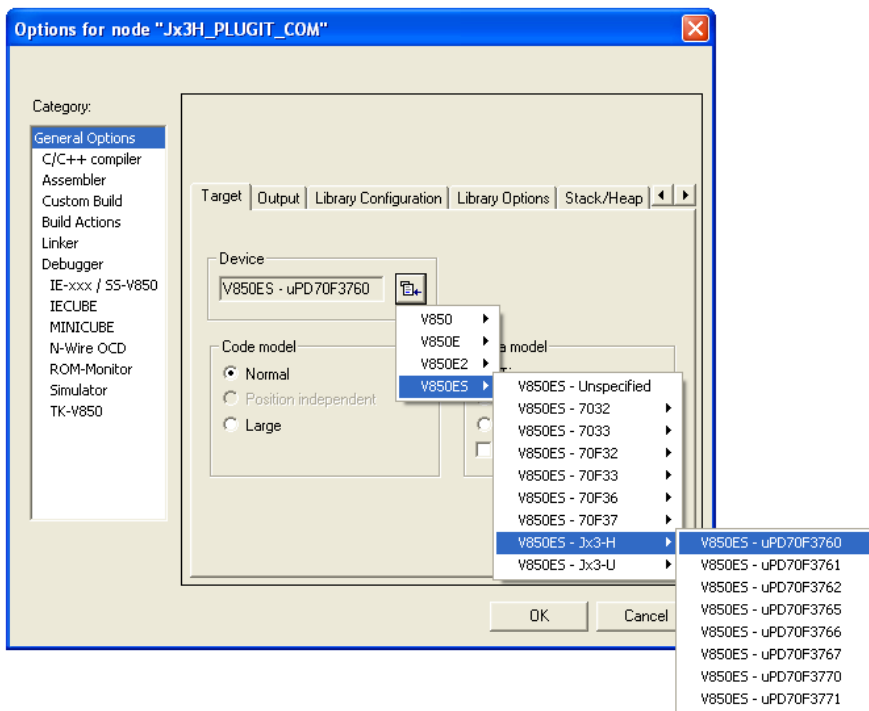


Figure 25 : Device selection

It is necessary to tell the compiler where the additional Header files of the demonstration samples are located. Select the category “C/C++ compiler” and then the tab “Preprocessor”. Now the additional Include path can be added to the field “Additional include directories”.

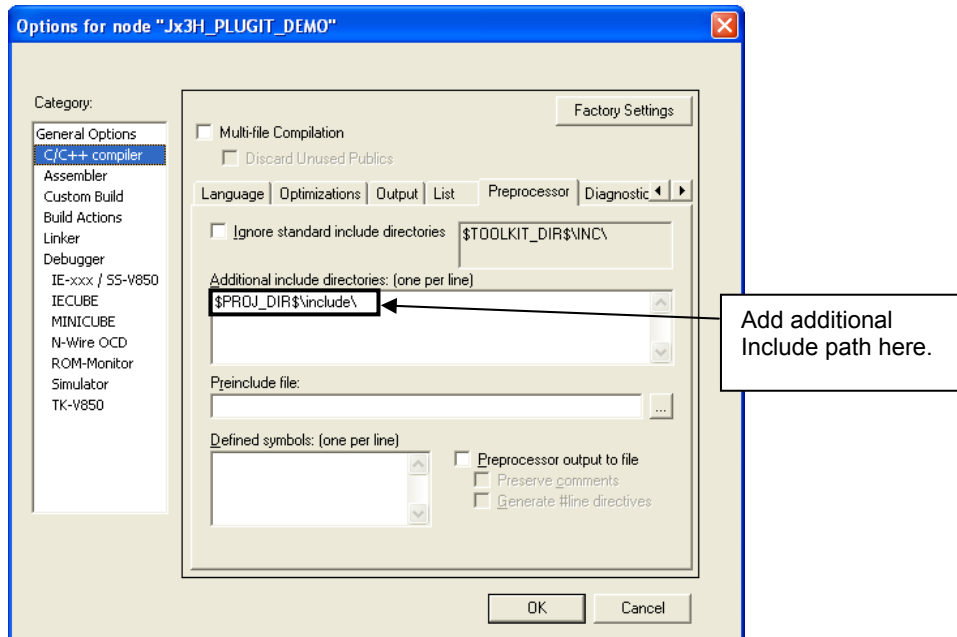


Figure 26: Include directory

As next step select the category “Debugger”. Make sure that the driver is set to “MINICUBE”. After this select the category “MINICUBE” down below and tick the box “Use MINICUBE2 (MINI2)” in order to use the On-Board debug function of the V850ES/Jx3-H - Plug it! board.

The corresponding COM port where the V850ES/Jx3-H - Plug it! board is connected to the host PC will be detected automatically by the IAR C-SPY debugger.

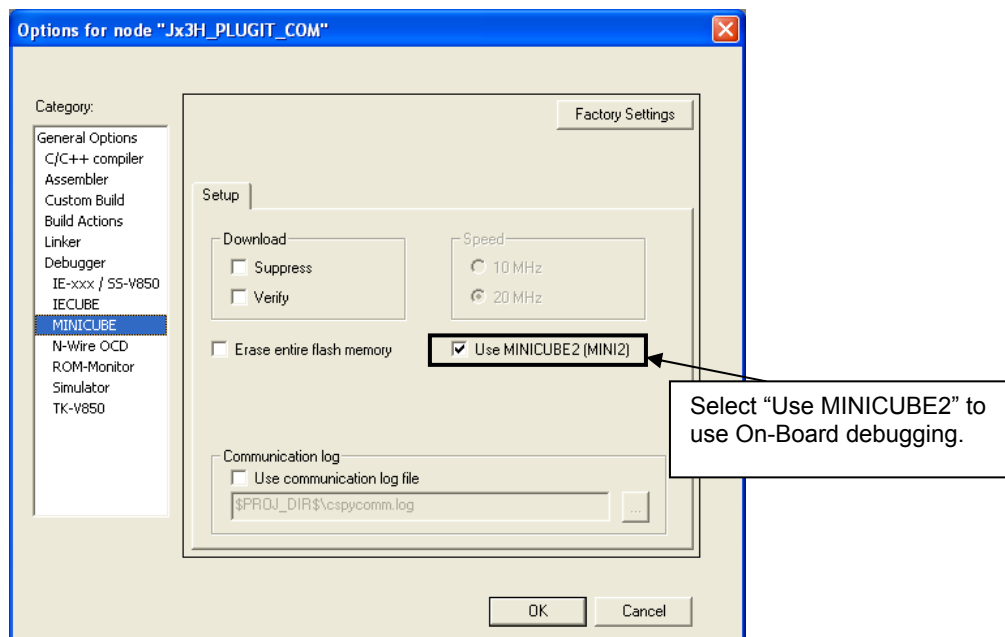


Figure 27: IAR debugger options

Note: Although the On-board debug interface is used, the MINICUBE2 C-Spy driver must be selected instead of the standard driver TK-V850 used for other starter kits. If the debug session via MINICUBE (QB-V850MINI) and the N-Wire connector shall be started, the MINICUBE C-Spy driver must be selected.

Next the correct linker settings of the demo project will be checked. This can be done in the “Linker” category as shown below. Select the “Config” tab and check that the linker command file “lnk70F3760.xcl” is selected. This file is used by the linker and contains information on where to place the different sections of code, data and constants that may be used within the demo project:

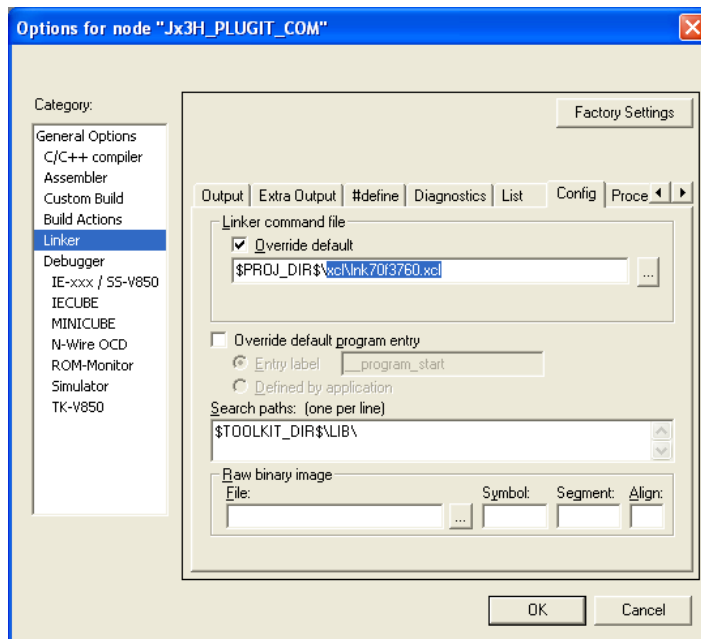



Figure 28: Embedded Workbench Linker Configuration

Now after everything has been setup correctly it's time to compile and link the demonstration project. Close the Options menu and select “Rebuild All” from the “Project” menu. If the project is compiled and linked without errors or warnings it can now be downloaded to the *V850ES/Jx3-H - Plug it!* board and debugged.

To start the IAR C-SPY debugger select the option “Debug” from the “Project” menu or press the () “Debugger” button.

In the next step the Emulator has to be configured before downloading a new application. Press the OK button to enter the emulator hardware setup. Set the configuration as show in the figure below and start the download by pressing the OK button.

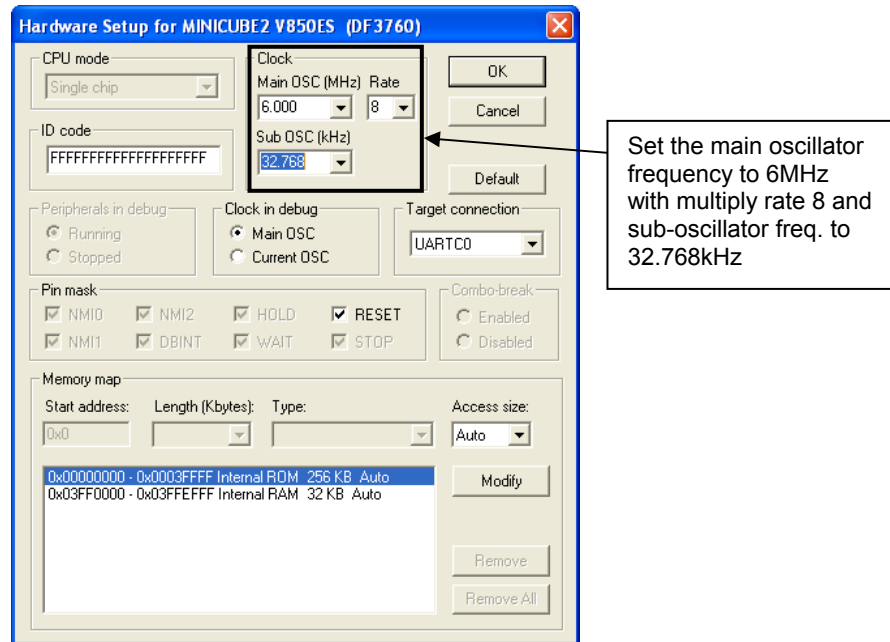



Figure 29: MINICUBE Hardware Setup Dialogue

Now the debugger is started and the demo project is downloaded to the *V850ES/Jx3-H - Plug it!* board. The progress of downloading is indicated by blue dots in the MINICUBE Emulator window. Please note that downloading of larger executables may take some time.

After the download was completed all debug features of IAR C-SPY debugger are available, i.e. Single Stepping, Step Over/-In/-Out, Go-Execution, Breakpoints, Register / Memory view etc. To run the sample application select the option “Go” from the “Debug” menu or press the () “Go” button.

To get more details on the debugger configuration and capabilities please refer to the “IAR Embedded Workbench IDE User Guide” of the IAR installation.

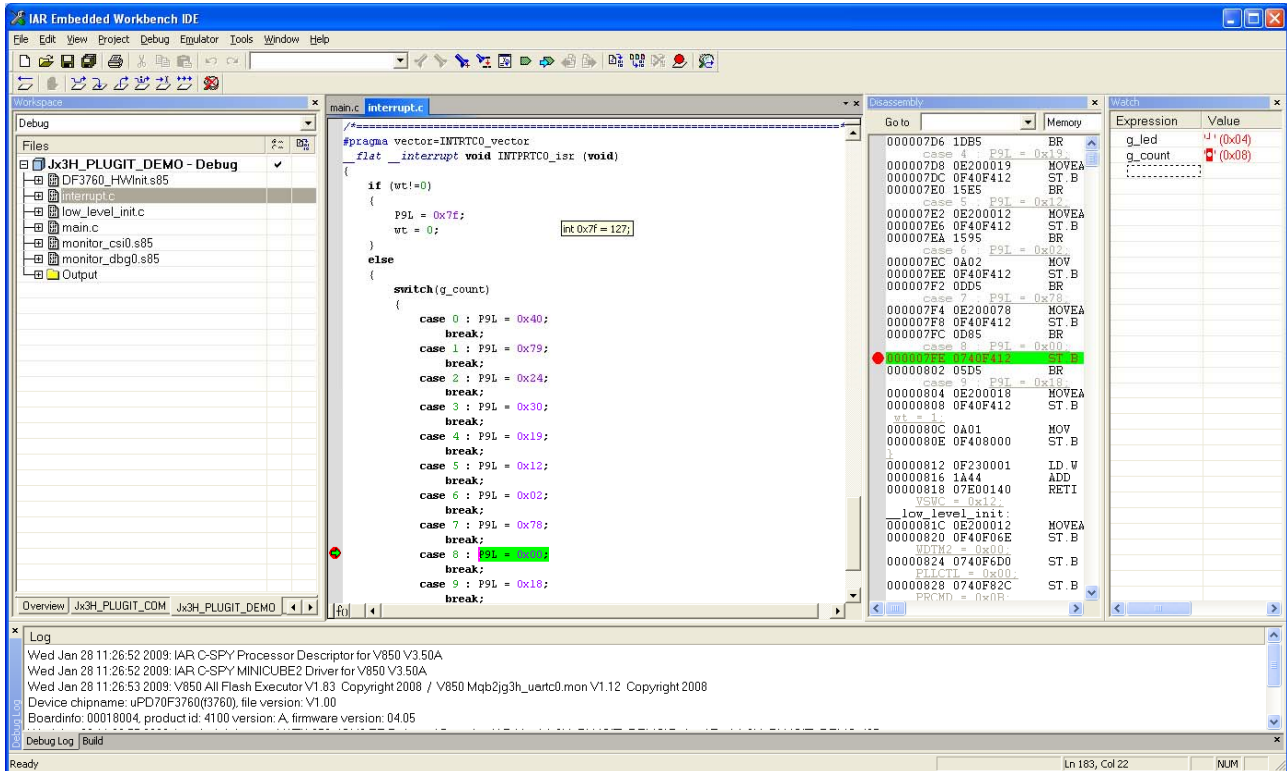


Figure 30: IAR C-SPY debugger

9. Troubleshooting

In some cases it might happen that the connection to the *V850ES/Jx3-H - Plug it!* board can not be established. This can be caused by the following three situations:

- Wrong security ID:** The security ID is required to prevent the FLASH memory of the 70F3760 microcontroller from being read by an unauthorized person. The security ID is located in the internal flash memory at addresses 0x0070-0x0079 of the 70F3760 microcontroller. The IAR C-SPY debugger starts only when the security ID that is set during debugger start-up and the security ID set at addresses 0x0070 to 0x0079 do match. In the above mentioned case it is necessary to erase the internal flash memory of the 70F3760 microcontroller to restore the security ID. Details about erasing and programming the internal flash memory of the 70F3760 are described in [chapter 10 "Flash Programmer WriteEZ1"](#)
- Wrong Emulator Driver:** Make sure that the driver in the category "Debugger" of the project options menu is set to "MINICUBE". After this select the category "MINICUBE" down below and tick the box "Use MINICUBE2 (MINI2)" in order to use the right driver settings for the *V850ES/Jx3-H - Plug it!* board.
- Wrong Oscillator Frequency:** Make sure that the clock settings in the Emulator Hardware Setup fit to the used oscillators on the board. In case of *V850ES/Jx3-H - Plug it!* the main oscillator frequency is 6 MHz with multiply rate 8. The sub-oscillator frequency is 32.768 kHz.

10. Flash Programmer WriteEZ1

The flash programmer Write EZ1 doesn't need to be installed, but can be directly started from the CDROM.

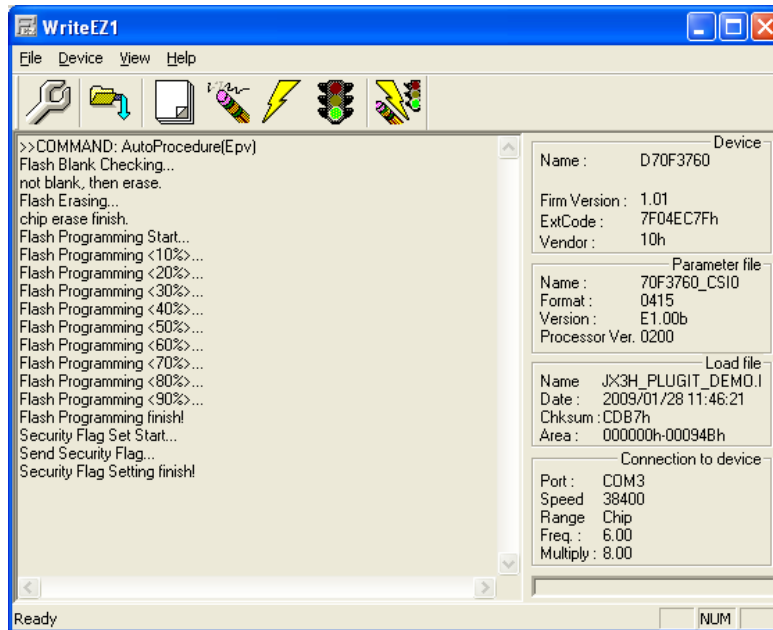


Figure 31: WriteEZ1 User Interface

10.1 Device Setup

To provide all necessary information about the device to be programmed, only the corresponding flash parameter file must be loaded. The parameter file (*.prm) for the μ PD70F3760 is located on the CDROM in the folder \WriteEZ1\PRM-70F3771_V100\. Please use the menu "Device -> Setup..." to open the following dialogue and the button "PRM File Read" to select the parameter file. Select the file 70F3760_CS10.prm.

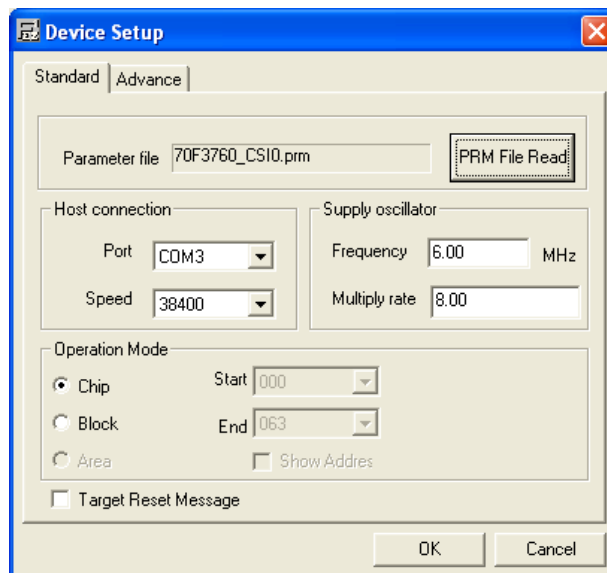


Figure 32: WriteEZ1 Device Setup Dialogue

Please check that the correct host communication port is selected. The used communication port can be seen in the [Windows Device Manager](#).

10.2 Using WriteEZ

After a successful device selection the internal flash memory can be blank-checked, erased, programmed or verified. WriteEZ can be controlled either by menu or by buttons

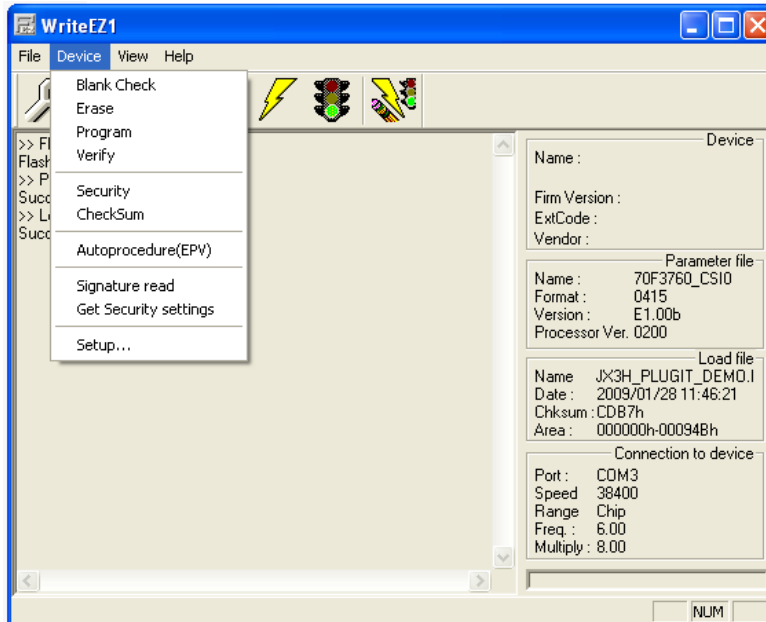


Figure 33: WriteEZ1 Device Menu








	device setup button
	load file button
	blank check button
	erase button
	program button
	verity button
	erase / program / verify button

Table 11: WriteEZ action buttons

WriteEZ1 supports Intel-Hex and Motorola S-record file formats as input file.

11. Sample programs

11.1 General Introduction

Each of the sample programs is located in a single directory, which will be called main-directory of the sample. This main directory of each sample contains the complete project inclusive all output files of the development tool. In the root directory of all sample programs the workspace file “**Jx3-H_PLUGIT.eww**” is located. The sample workspace includes all sample projects.






 settings	Workspace configuration files, IAR Embedded Workbench
 Jx3-H_PLUGIT_COM	Serial Conversion sample project
 Jx3-H_PLUGIT_DEMO	Demonstration sample project
 Jx3-H_PLUGIT_STORAGE	Mass Storage Device sample project
 Jx3-H_PLUGIT.eww	Workspace file, IAR Embedded Workbench

Table 12: Sample directory structure

As an alternative to open the sample-workspace each project file “<name>.ewp” can be added to any user created workspace.

All sample programs use the same directory structure:












 Jx3-H_PLUGIT_COM	Serial Conversion sample project
 Jx3-H_PLUGIT_DEMO	Demonstration sample project
 Debug	debug output files for IAR C-SPY debugger
 include	C header files
 settings	configuration files, IAR Embedded Workbench
 source	C source files
 xcl	Linker control file
 Jx3-H_PLUGIT_DEMO.dep	dependency information file, IAR Embedded Workbench
 Jx3-H_PLUGIT_DEMO.ewd	project setting file, IAR C-SPY debugger
 Jx3-H_PLUGIT_DEMO.ewp	project file, IAR Embedded Workbench
 Jx3-H_PLUGIT_STORAGE	Mass Storage Device sample project

Table 13: Example directory structure

The main directory contains only the project files for the IAR Systems Embedded Workbench for NEC V850. All source files are located in the subdirectory `/source`. The `/include` subdirectory contains the header files. The `/xcl` subdirectory contains the linker control file of the 70F3760 device. All output files including the object files, list files, debug information and finally the executable file are stored in the directory `/Debug`.

For details of using the IAR Embedded Workbench and the IAR C-SPY debugger please refer to the “V850 IAR Embedded Workbench IDE User Guide”.

11.2 “Jx3-H_PLUGIT_DEMO” sample program

This sample program is a simple random number generator between 0 and 9 to demonstrate the usage of the Jx3-H_PLUGIT_DEMO starterkit. After starting the generator by pressing SW3 the segments of the seven-segment-LED are flashing until a random number is generated by pressing SW4. The generated number is displayed and flashes at the seven-segment-LED U3. To start a new generation loop, please press again SW3.

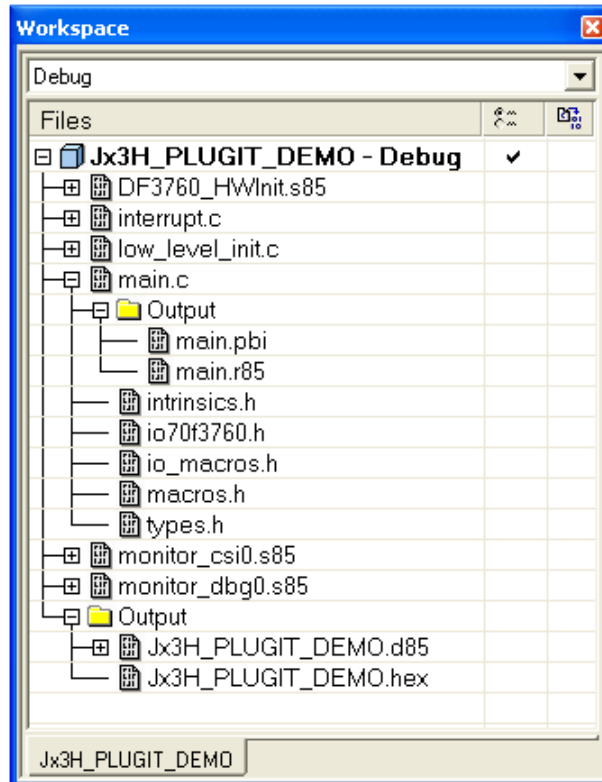


Figure 34: Project Window Jx3-H_PLUGIT_DEMO

11.2.1 How to run the sample program

After starting the application, the switches SW3, SW4 and the seven-segment LED are the user interface to the sample program.

- Switch SW3

By pressing this switch, the generation of a new random number is started. In this state the segments of the seven-segment display are flashing.

- Switch SW4

By pressing this switch, the generation of a new random number is stopped. The newly generated number is displayed and flashes at the seven-segment display.

11.3 “Jx3-H_PLUGIT_COM” Serial Conversion sample program

The USB serial conversion driver is a communication device class (CDC) sample driver for built-in USB function controller that is built of the μ PD70F3760. It is detected as communication device class (virtual COM port) when it connects with the host. The sample Application loads the user data received by the USB function controller and converts it to uppercase characters or to lowercase characters (only ASCII character data), and outputs the result on the USB to the virtual COM-Port.

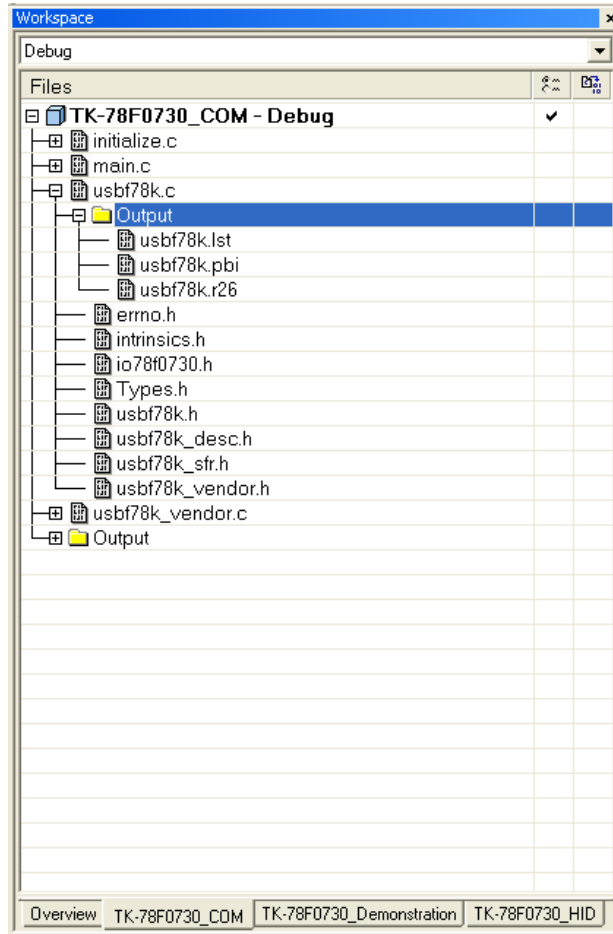


Figure 35: Project Window Jx3H_PLUGIT_COM

11.3.1 Development Environment

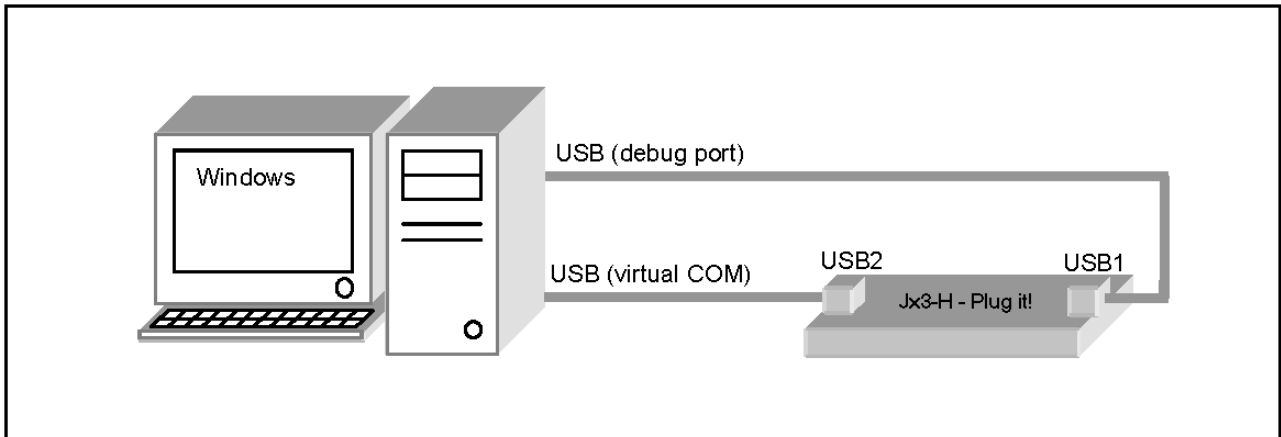


Figure 36: Development Environment

11.3.2 System Structure

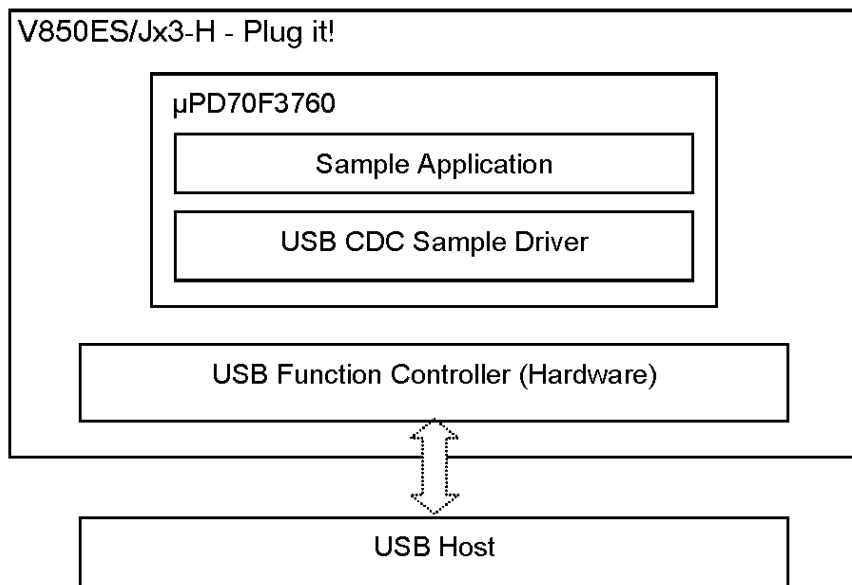


Figure 37: System Structure of the sample program

11.3.3 How to run the sample program

The sample application loads the communication data received by the built-in USB function controller of the μ PD70F3760. It converts the data to uppercase characters or to lowercase characters (only ASCII character data), and then executes the sending process by calling USB function controller sending process. The sample application process monitors the data receiving flag to start the process. As communication tool running on the host PC any terminal program supporting virtual communication ports can be used. To establish a connection and run the sample please use the following procedure:

- o Connect port USB1 and start a debug session. In the Windows Device Manager the debug port can be seen:

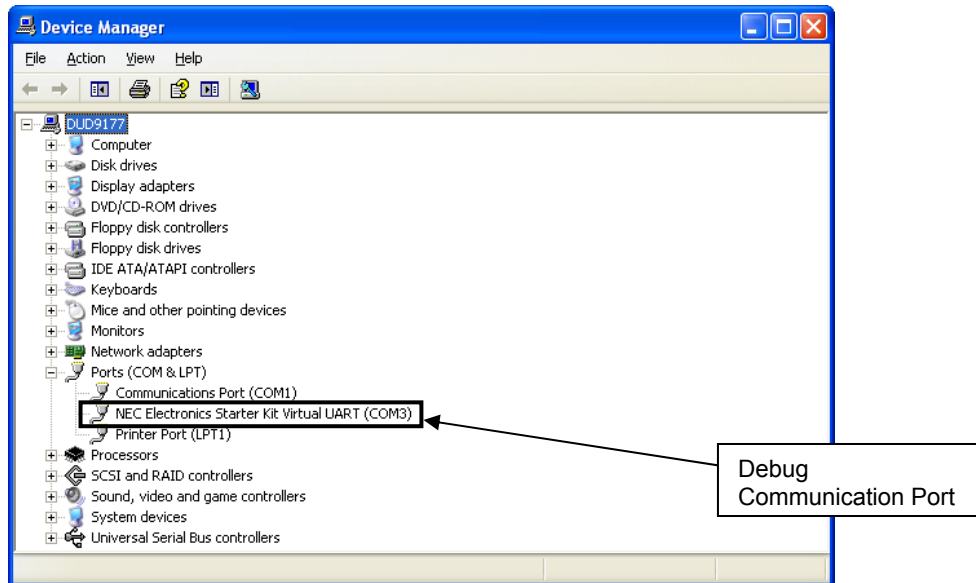


Figure 38: Jx3H_PLUGIT_COM-sample Debug Communication Port

- o Connect port USB2 and run the sample application. The Windows “Found New Hardware Wizard” will pop up. Check that “No, not this time” is selected and click **Next>**.



Figure 39: Found New Hardware Wizard 1 (Windows XP)

- o Check that "Install from a list or specific location (Advanced)" is selected, then click **Next>**.

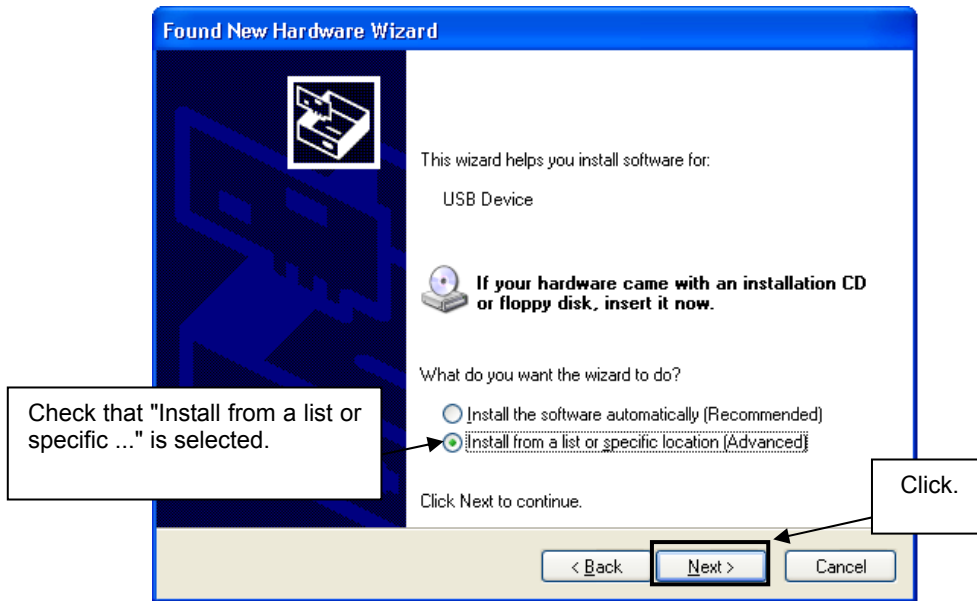


Figure 40: Found New Hardware Wizard 2 (Windows XP)

- o Check that "Search for the best driver in these locations." is selected. Select the "Include this location in the search:" check box and then click **Browse**.

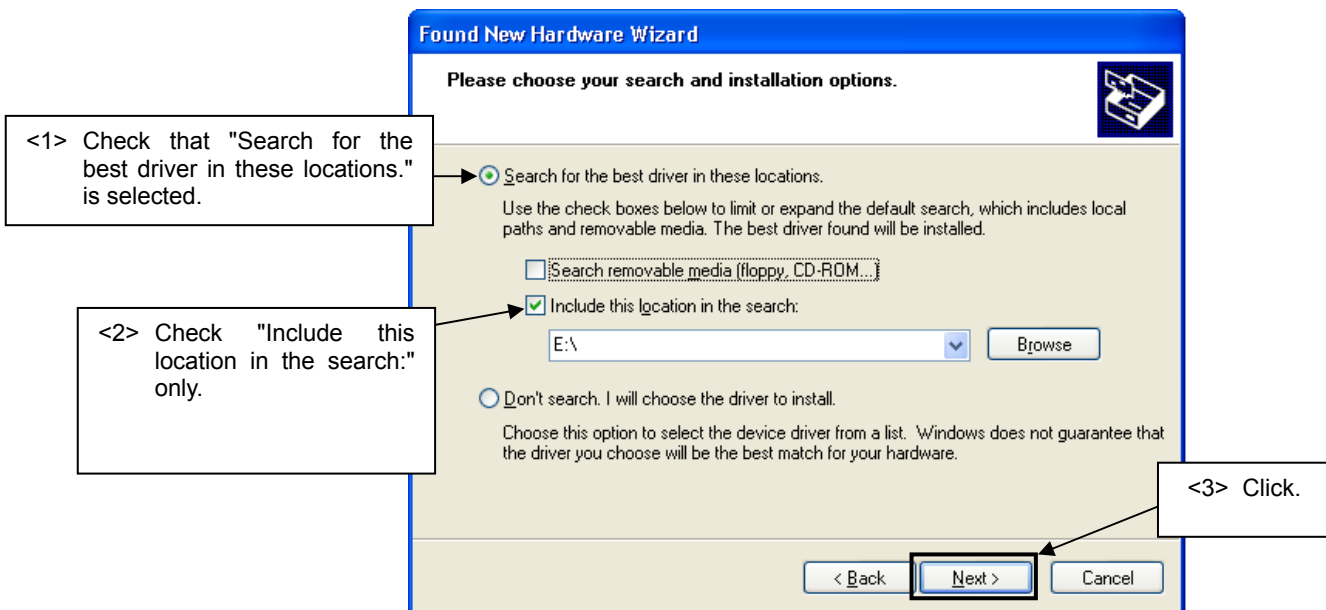


Figure 41: Search Location Specification 1 (Windows XP)

- o Locate the folder `\SamplePrograms\Jx3H_PLUGIT_COM\drivers\XP\` for Windows XP or `\SamplePrograms\Jx3H_PLUGIT_COM\drivers\VISTA\` for Windows Vista on the CD-ROM and click **OK**.

- In the Windows Device Manager a new virtual communication port named 'NEC Electronics Jx3H Virtual UART' will appear:

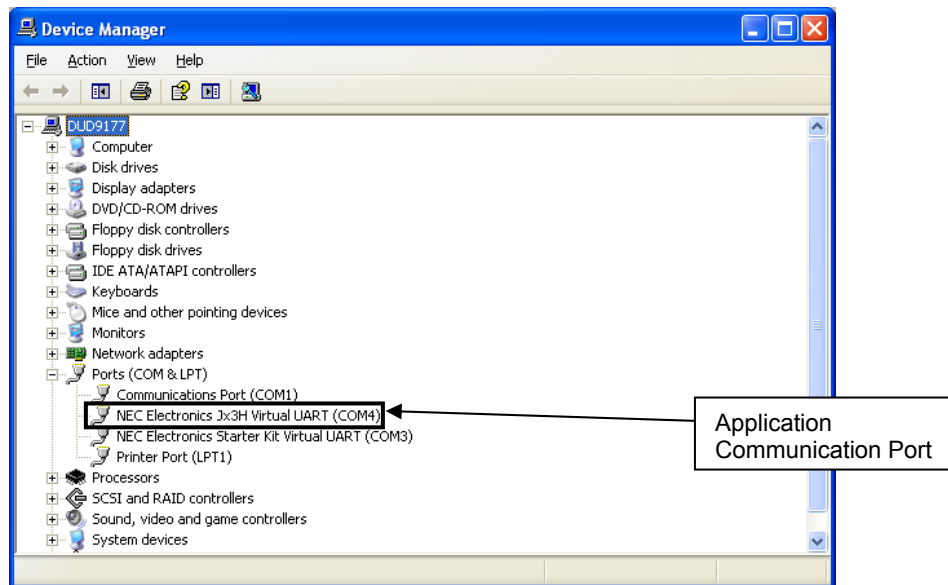


Figure 42: Jx3H_PLUGIT_COM-sample Application Communication Port

- Start the terminal program and setup a connection to the application virtual communication port
- Send some test characters

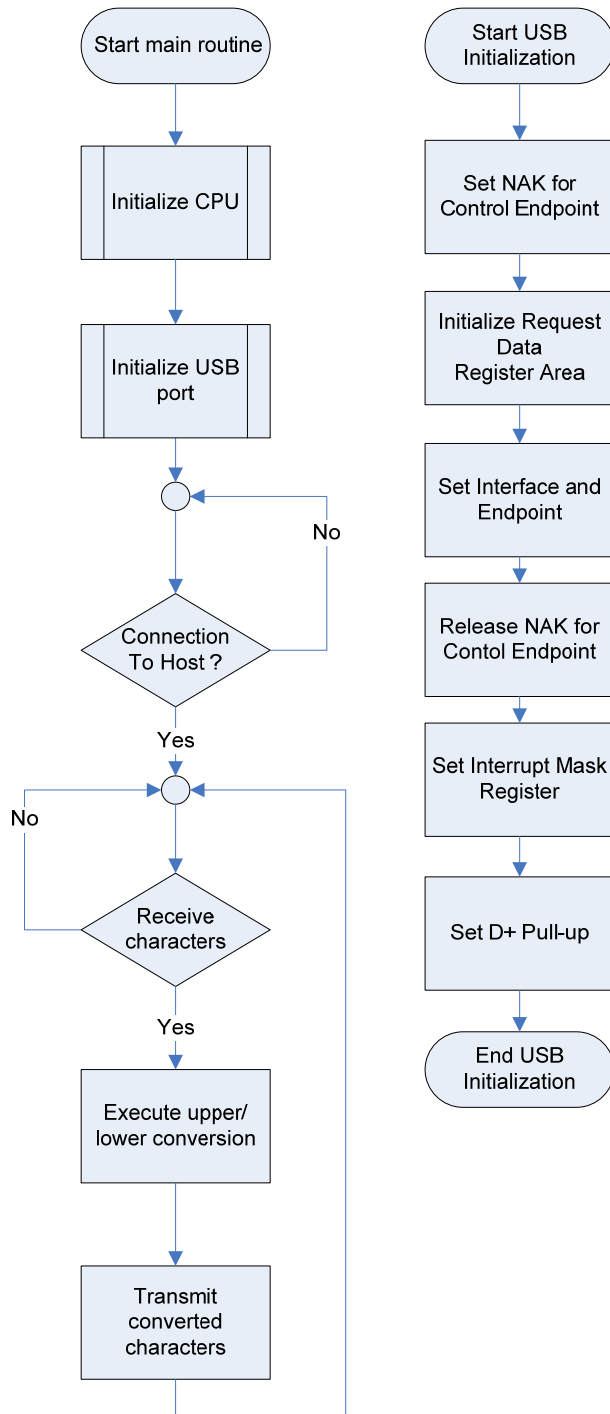
11.3.4 Communication Device Class (CDC)

For information about USB Communication Device Class (CDC), please refer to the USB CDC specification “Universal Serial Bus Class Definitions for Communication Devices Version 1.1”.

The sample is CDC Abstract Control Model, and responds to the following class requests.

- **SendEncapsulatedCommand**
To format the communication class interface control protocol for sending commands.
- **GetEncapsulatedResponse**
To format the communication class interface control protocol for requesting responses.
- **SetLineCoding**
Request for specifying the serial communication format.
- **GetLineCoding**
Request for retrieving the serial communication format.
- **SetControlLineState**
Request for control signals with RS-232/V.24 format.

11.3.5 Processing Flow

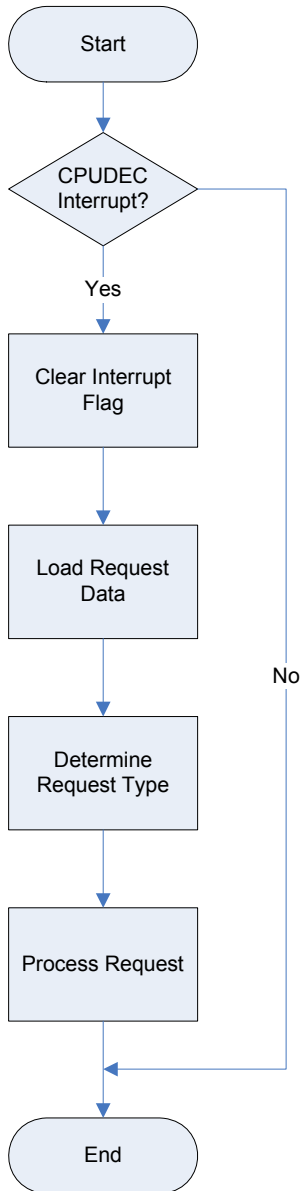


Process Description USB Initialization

- Set NAK for Control Endpoint
A NAK response is sent to all the requests including automatic execution requests. It sets for hardware not to return unexpected data in response to an automatic execution request until registration of data used for the automatic execution request is complete.
- Initialize Request Data Register Area
Descriptor data used to respond to a "Get Descriptor" request is registered in a register. Those data include device status, endpoint 0 status, device descriptor, configuration descriptor, interface descriptor, and endpoint descriptor.
- Set Interface and Endpoint
Set the number of supported interfaces, the status of alternative settings, and the relationship between the interface and endpoints.
- Release NAK for Control Endpoint
The NAK setting at control endpoint is released during registration of data for an automatic execution request is complete.
- Set Interrupt Mask Register
Set the mask for each interrupt source indicated in the interrupt status register of the USB function controller.
- Set D+ Pull-up
Pull-up the D+ signal, and let the host recognize a device is connected.

Figure 43: Flow Chart Jx3H_PLUGIT_COM application

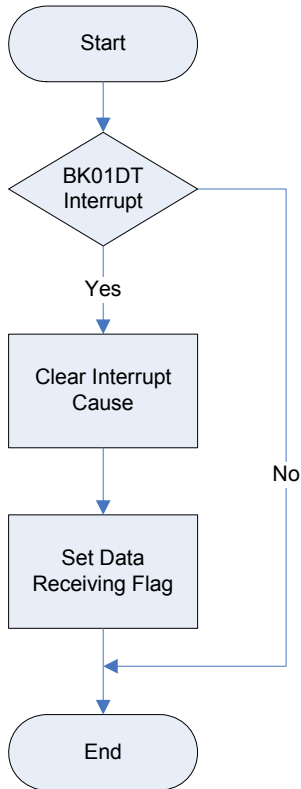
Endpoint 0 is the endpoint for control transferring. However, standard device requests, such as the request used by enumeration for plug-in, are automatically responded by hardware. Therefore, this should monitor standard requests that are not responded by hardware, class requests, and vendor requests.



Endpoint 0 Monitoring Process

- Check CPUDEC Interrupt
If an interrupt is occurred, CPUDEC bit of UF0IS1 becomes ON (1).
- Clear Interrupt Flag
Clear the interrupt flag by setting the CPUDEC bit of UF0IC1 to OFF (0).
- Load Request Data
Load received data from FIFO, and structure the request data.
- Determine Request Type
It distinguishes whether the request is standard request that is not responded by hardware automatically, class request, or vendor request.
- Process Request
Process the request depending on the request type.

Figure 44: Flow Chart Jx3H_PLUGIT_COM
Endpoint 0 Monitoring Process



Endpoint 1 Monitoring Process

- Check BK01DT Interrupt
It detects the completion of receiving interrupt if BK01DT bit of UF0IS3 is ON (1).
- Clear Interrupt Cause
Clear the cause of interrupt by setting BK01DTC bit of UF0IC3 to OFF (0).
- CBW Data Receiving Process
Execute CBW Data Receiving Process.

Figure 45: Flow Chart Jx3H_PLUGIT_COM
Endpoint 1 Monitoring Process

11.3.6 File Structure

File	Description
main.c	Main routine
low_level_init.c	CPU and board initialization
usb850.c	USB initialization. Interrupt process. Bulk transfer. Control transfer.
usb850_communication.c	CDC specific process
DF3760_HWinit.s85	Security-ID setting
monitor_dbg0.s85	Secures interrupt vector for Debugging
monitor_uartc0.s85	Secures interrupt vector of serial interface

Table 14: Source Modules USB Serial Conversation Sample

File	Description
main.h	Function prototypes defined in main.c
errno.h	Error code definitions
types.h	Datatype definitions
usb850.h	Macro definitions for USB function register setting
usb850_sfr.h	Macro definitions for USB function controller register access
usb850_desc.h	Descriptor definitions
usb850_communication.h	usb850_communication.c function prototype declaration

Table 15: Header Files USB Serial Conversation Sample

11.3.7 Descriptor Information

Field	Size	Description	Value
bLength	1	Descriptor size	0x12
bDescriptor	1	Descriptor type	0x01
bcdUSB	2	BCD format of USB release number	0x0200
bDeviceClass	1	Class code 0x00H: no class 0xFFH: vendor 0x01-0xFEH: specific	0x02
bDeviceSubClass	1	Sub-class code	0x00
bDeviceProtocol	1	Protocol code 0x00: no specific protocol 0xFF: vendor-specific protocol	0x00
bMaxPacketSize0	1	Maximum packet size at endpoint 0	0x40
idVendor	2	Vendor ID (USB IF assigns)	0x0409
idProduct	2	Product ID (vendor assigns)	0x01D0
bcdDevice	2	BCD format of device release number	0x0001
iManufacture	1	Index to string descriptor to indicate manufacturer	0x01
iProduct	1	Index to string descriptor to indicate product	0x02
iSerialNumber	1	Index to string descriptor to indicate serial number	0x03
bNumConfigurations	1	Number of devices that can be configured	0x01

Table 16: Device Descriptor Serial Conversion Sample

Field	Size	Description	Value
bLength	1	Descriptor size	0x09
bDescriptor	1	Descriptor type	0x02
wTotalLength	2	Total length of the configuration (configuration, interface, endpoint, and other descriptors)	0x0030
bNumInterfaces	1	Number of interfaces supported in the configuration	0x02
bConfigurationValue	1	Input value (>= 1) for selecting this configuration with SetConfiguration	0x01
iConfiguration	1	Index to string descriptor to indicate descriptor	0x00
bmAttributes	1	Configuration attributes with the unit of bit D7: "1" D6: self-powered D5: remote wake-up D4-D0: reserved (0)	0x80
bMaxPower	1	Maximum power consumption of bus with the unit of 2mA	0x1B

Table 17: Configuration Descriptor Serial Conversion Sample

Field	Size	Description	Value
bLength	1	Descriptor size	0x09
bDescriptor	1	Descriptor type	0x04
bInterfaceNumber	1	Index Number (0 based) to indicate this interface in the configuration	0x00
bAlternateSetting	1	Input value to select alternate setting in SetInterface	0x00
bNumEndpoints	1	Interface endpoint number (excluding endpoint 0)	0x01
bInterfaceClass	1	Class Code 0x00: no class 0xFF: vendor 0x01 – 0xFE: specific	0x02
bInterfaceSubclass	1	Subclass code	0x02
bInterfaceProtocol	1	Protocol code 0x00: no specific protocol 0xFF: vendor specific protocol	0x00
iInterface	1	Index to string descriptor to indicate interface	0x00

Table 18: Interface Descriptor Serial Conversion Sample

Field	Size	Description	Value
bLength	1	Descriptor size	0x07
bDescriptor	1	Descriptor type	0x05
bEndpointAddress	1	Endpoint address bits: D7: Direction 0: OUT, 1: IN D6-D4: Reserved (0) D4-D0: Endpoint number	0x87
bmAttributes	1	Attribute bits: D1-D0: Transfer type 0: Control 1: Isochronous 2: Bulk 3: Interrupt *D5-D2 is used only by isochronous endpoint D3-D2: Synchronization type 0: No synchronization 1: Asynchronous 2: Adaptive 3: Synchronous D5-D4: Usage type 0: Data endpoint 1: Feedback endpoint 2: Dependant feedback endpoint 3: (reserved)	0x03
wMaxPacketSize	2	Payload size bits: D10-D0: Maximum packet size D12-D11: auditory transaction number per μ frame (only high-speed isochronous and interrupt) 0: No addition (1 transaction / μ frame) 1: 1 (2 transaction / μ frame) 2: 2 (3 transaction / μ frame) 3: Not in use (reserved)	0x0008
bInterval	1	Polling interval for data transfer endpoint Full/low speed interrupt: specify with unit of ms (number of frames) High-speed isochronous/interrupt: specify N for 2 raised to the power of N-1 with unit of μ frame (for example, 1 polling in 8 μ frames when bInterval is 4) Full-speed isochronous: specify N for 2 raised to the power of N-1 with unit of 1ms High-speed bulk/control: specify the maximum NAK rate for endpoint with unit of μ frame 0 means that it does not respond NAK on OUT/DATA transaction	0x0A

Table 19: Endpoint Descriptor Serial Conversion Sample

Field	Size	Description	Value
bLength	1	Descriptor size	0x07
bDescriptor	1	Descriptor type	0x05
bString	41	Language Code: 0x09 0x04	
		Manufacture: "NEC Electronics Co."	
		Product: "CDCDrv"	
		Serial Number: "0_98765432"	

Table 20: String Descriptor Serial Conversion Sample

11.4 “Jx3-H_PLUGIT_STORAGE” Mass Storage sample program

This sample program is a mass storage class (MSC) sample driver for the built-in USB function controller of the μ PD70F3760. It is detected as bulk-only mass storage class device when it connects with the host and the Windows standard driver for mass storage devices will be used. It can be formatted by the host system with the FAT file system. Data such as files and directories can be written to an area of 24Kbyte to the built-in RAM. The stored data will be lost by pressing the Reset SW or turning off the power of the device.

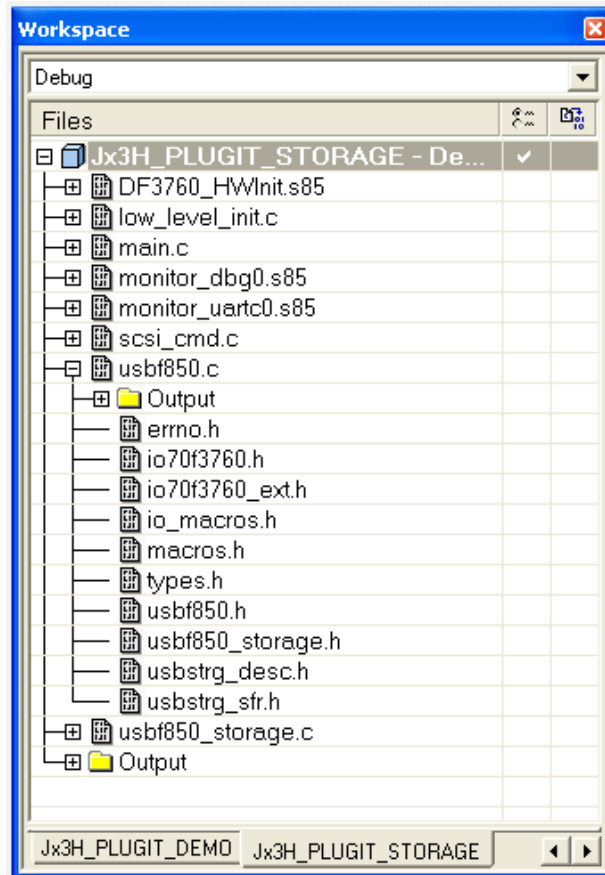


Figure 46: Project Window Jx3H_PLUGIT_STORAGE

11.4.1 Development Environment

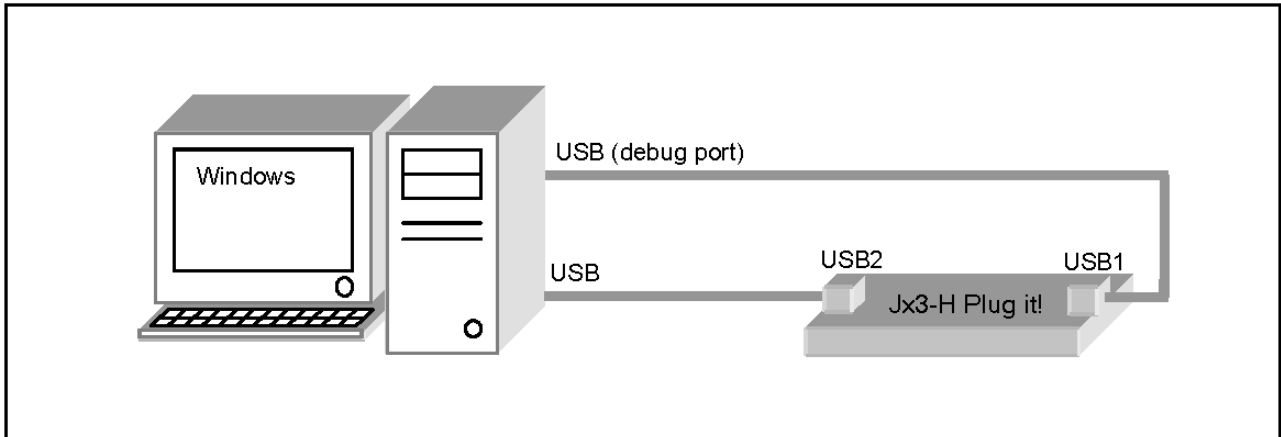


Figure 47: Development Environment

11.4.2 System Structure

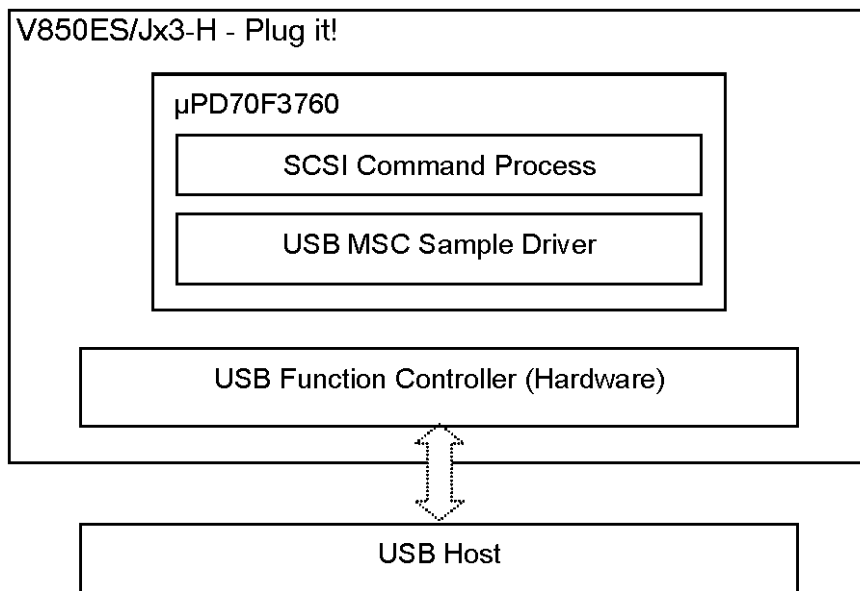


Figure 48: System Structure of the sample program

11.4.3 How to run the sample program

The mass storage class (MSC) sample driver uses the Windows standard drivers for mass storage devices. To run the sample please use following procedure:

- Connect port USB1 and start a debug session.
- Connect port USB2 and run the sample application. The driver will be automatically installed and a new Removable Disk Drive will appear in the Windows Explorer.
- Format the drive with the FAT file system and store some test files on it.

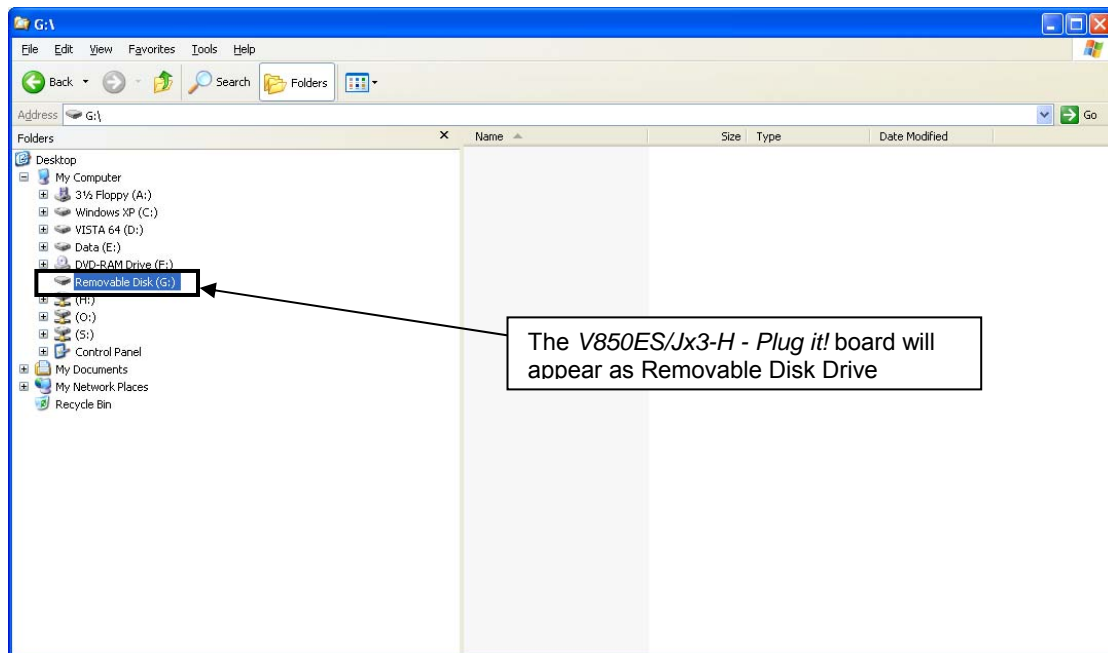


Figure 49: Jx3H_PLUGIT_STORAGE sample Mass Storage Device

11.4.4 Mass Storage Class (MSC)

The sample driver is a bulk-only mass storage class (MSC) driver. Bulk-only transport transfers commands, status, and data with bulk transferring. Commands are sent from host to device with bulk-out transferring. This is defined in the Command Block Wrapper (CBW) Format. When the command needs to transfer data as well, it executes to input/output the data with bulk-in and bulk-out transferring.

The status data is transferred from device to host with bulk-in transferring. This is defined in the Command Status Wrapper (CSW) Format. For information about USB mass storage class (MSC), please refer to the MSC specification "Universal Serial Bus Mass Storage Class Bulk-Only Transport Revision 1.0". The sample driver responds to the following class requests:

- GET_MAX_LUN
To retrieve the logical unit number for the mass storage device.
- MASS_STORAGE_RESET
To reset the interface that is related to the mass storage device.

11.4.5 SCSI Command Process

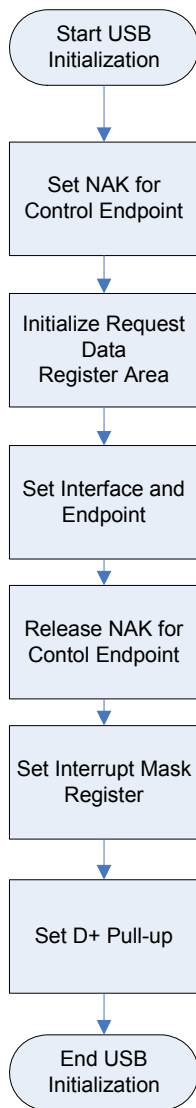
In the sample program, SCSI command process is structured on the MSC sample driver. It processes encapsulated SCSI commands to the Command Block Wrapper (CBW).

The sample driver supports following SCSI commands:

- | | |
|--------------------------|---------------------|
| - TEST_UNIT_READY | - SEEK |
| - START_STOP_READY | - SYNCHRONIZE_CACHE |
| - REQUEST_SENSE | - INQUIRY |
| - READ6 | - READ10 |
| - MODE_SENSE6 | - MODE_SENSE10 |
| - READ_FORMAT_CAPACITIES | - READ_CAPACITY |
| - WRITE6 | - WRITE10 |
| - MODE_SELECT | - MODE_SELECT10 |
| - VERIFY | - WRITE_VERIFY |
| - WRITE_BUFF | - PREVENT |

*Responds STALL when it receives commands other than above.

11.4.6 Processing Flow

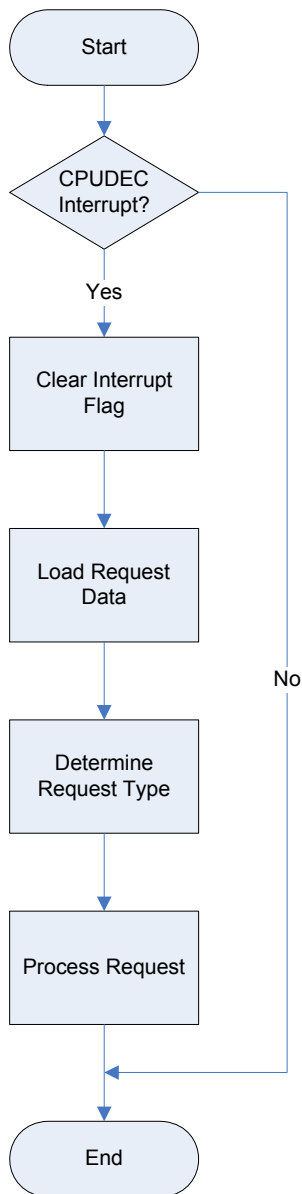


Process Description USB Initialization

- Set NAK for Control Endpoint
A NAK response is sent to all the requests including automatic execution requests. It sets for hardware not to return unexpected data in response to an automatic execution request until registration of data used for the automatic execution request is complete.
- Initialize Request Data Register Area
Descriptor data used to respond to a "Get Descriptor" request is registered in a register. Those data include device status, endpoint 0 status, device descriptor, configuration descriptor, interface descriptor, and endpoint descriptor.
- Set Interface and Endpoint
Set the number of supported interfaces, the status of alternative settings, and the relationship between the interface and endpoints.
- Release NAK for Control Endpoint
The NAK setting at control endpoint is released during registration of data for an automatic execution request is complete.
- Set Interrupt Mask Register
Set the mask for each interrupt source indicated in the interrupt status register of the USB function controller.
- Set D+ Pull-up
Pull-up the D+ signal, and let the host recognize a device is connected.

Figure 50: Flow Chart Jx3H_PLUGIT_COM
USB Initialization Process

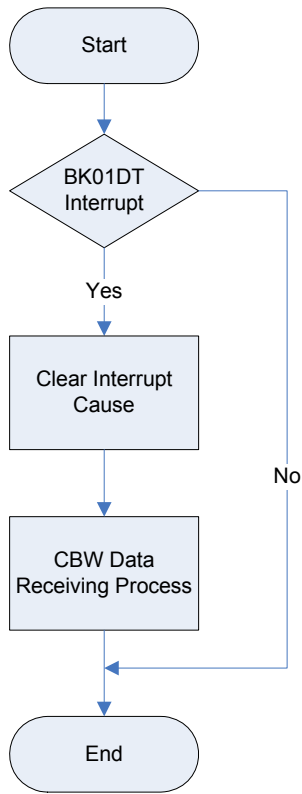
Endpoint 0 is the endpoint for control transferring. However, standard device requests, such as the request used by enumeration for plug-in, are automatically responded by hardware. Therefore, this should monitor standard requests that are not responded by hardware, class requests, and vendor requests.



Endpoint 0 Monitoring Process

- Check CPUDEC Interrupt
If an interrupt is occurred, CPUDEC bit of UF0IS1 becomes ON (1).
- Clear Interrupt Flag
Clear the interrupt flag by setting the CPUDEC bit of UF0IC1 to OFF (0).
- Load Request Data
Load received data from FIFO, and structure the request data.
- Determine Request Type
It distinguishes whether the request is standard request that is not responded by hardware automatically, class request, or vendor request.
- Process Request
Process the request depending on the request type.

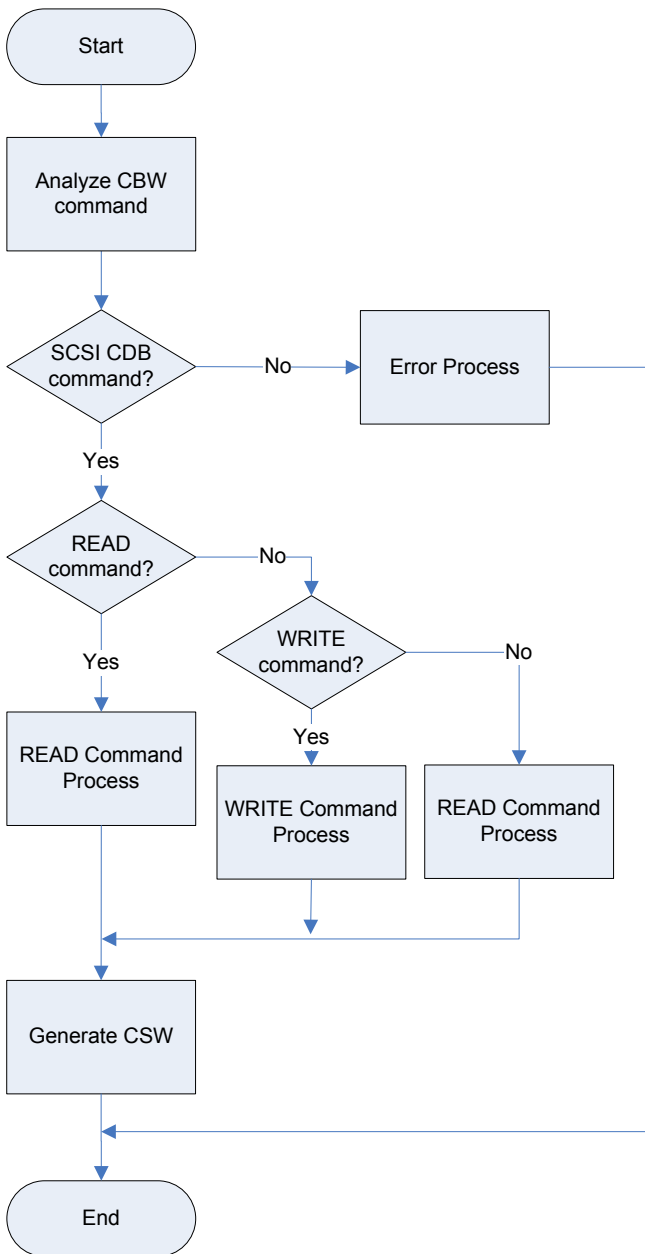
Figure 51: Flow Chart Jx3H_PLUGIT_COM
Endpoint 0 Monitoring Process



Endpoint 1 Monitoring Process

- Check BKO1DT Interrupt
It detects the completion of receiving interrupt if BKO1DT bit of UF0IS3 is ON (1).
- Clear Interrupt Cause
Clear the cause of interrupt by setting BKO1DTC bit of UF0IC3 to OFF (0).
- CBW Data Receiving Process
Execute CBW Data Receiving Process.

Figure 52: Flow Chart Jx3H_PLUGIT_COM
Endpoint 1 Monitoring Process



SCSI Command Process

- Analyze CBW Command
Analyze the content of CBW after receiving CBW data. It saves CBW tag and checks valid data number for CBWCB and command direction, and then executes READ, WRITE, or NO DATA process.
- Error Process
Process for errors such as invalid command.
- READ Command Process
Process READ related commands in SCSI commands.
- WRITE Command Process
Process WRITE related commands in SCSI commands.
- NO DATA Command Process
Process NO DATA related commands in SCSI commands.
- Generate CSW
Generate and send CSW data with results from command processes.

Figure 53: Flow Chart Jx3H_PLUGIT_COM SCSI Command Process

11.4.7 File Structure

File	Description
main.c	Main routine
low_level_init.c	CPU and board initialization
usb850.c	USB initialization. Interrupt process. Bulk transfer. Control transfer.
usb850_storage.c	MSC specific process
scsi_cmd.c	SCSI command process
DF3760_HWinit.s85	Security-ID setting
monitor_dbg0.s85	Secures interrupt vector for Debugging
monitor_uartc0.s85	Secures interrupt vector of serial interface

Table 21: Source Modules USB Mass Storage Sample

File	Description
main.h	Function prototypes defined in main.c
errno.h	Error code definitions
types.h	Datatype definitions
scsi.h	SCSI macro declaration
usb850.h	Macro definitions for USB function register setting
usbstrg_sfr.h	Macro definitions for USB function controller register access
usbstrg_desc.h	Descriptor definitions
usb850_storage.h	usb850_storage.c function prototype declaration

Table 22: Header Files USB Mass Storage Sample

11.4.8 Descriptor Information

Field	Size	Description	Value
bLength	1	Descriptor size	0x12
bDescriptor	1	Descriptor type	0x01
bcdUSB	2	BCD format of USB release number	0x0200
bDeviceClass	1	Class code 0x00H: no class 0xFFH: vendor 0x01-0xFEH: specific	0x00
bDeviceSubClass	1	Sub-class code	0x00
bDeviceProtocol	1	Protocol code 0x00: no specific protocol 0xFF: vendor-specific protocol	0x00
bMaxPacketSize0	1	Maximum packet size at endpoint 0	0x40
idVendor	2	Vendor ID (USB IF assigns)	0x0409
idProduct	2	Product ID (vendor assigns)	0x01D2
bcdDevice	2	BCD format of device release number	0x0001
iManufacture	1	Index to string descriptor to indicate manufacturer	0x01
iProduct	1	Index to string descriptor to indicate product	0x00
iSerialNumber	1	Index to string descriptor to indicate serial number	0x00
bNumConfigurations	1	Number of devices that can be configured	0x01

Table 23: Device Descriptor Mass Storage Sample

Field	Size	Description	Value
bLength	1	Descriptor size	0x09
bDescriptor	1	Descriptor type	0x02
wTotalLength	2	Total length of the configuration (configuration, interface, endpoint, and other descriptors)	0x0020
bNumInterfaces	1	Number of interfaces supported in the configuration	0x01
bConfigurationValue	1	Input value (>= 1) for selecting this configuration with SetConfiguration	0x01
iConfiguration	1	Index to string descriptor to indicate descriptor	0x00
bmAttributes	1	Configuration attributes with the unit of bit D7: "1" D6: self-powered D5: remote wake-up D4-D0: reserved (0)	0x80
bMaxPower	1	Maximum power consumption of bus with the unit of 2mA	0x1B

Table 24: Configuration Descriptor Mass Storage Sample

Field	Size	Description	Value
bLength	1	Descriptor size	0x09
bDescriptor	1	Descriptor type	0x04
bInterfaceNumber	1	Index Number (0 based) to indicate this interface in the configuration	0x00
bAlternateSetting	1	Input value to select alternate setting in SetInterface	0x00
bNumEndpoints	1	Interface endpoint number (excluding endpoint 0)	0x02
bInterfaceClass	1	Class Code 0x00: no class 0xFF: vendor 0x01 – 0xFE: specific	0x08
bInterfaceSubclass	1	Subclass code	0x06
bInterfaceProtocol	1	Protocol code 0x00: no specific protocol 0xFF: vendor specific protocol	0x50
iInterface	1	Index to string descriptor to indicate interface	0x00

Table 25: Interface Descriptor Mass Storage Sample

Field	Size	Description	Value
bLength	1	Descriptor size	0x07
bDescriptor	1	Descriptor type	0x05
bEndpointAddress	1	Endpoint address bits: D7: Direction 0: OUT, 1: IN D6-D4: Reserved (0) D4-D0: Endpoint number	0x81
bmAttributes	1	Attribute bits: D1-D0: Transfer type 0: Control 1: Isochronous 2: Bulk 3: Interrupt *D5-D2 is used only by isochronous endpoint D3-D2: Synchronization type 0: No synchronization 1: Asynchronous 2: Adaptive 3: Synchronous D5-D4: Usage type 0: Data endpoint 1: Feedback endpoint 2: Dependant feedback endpoint 3: (reserved)	0x02
wMaxPacketSize	2	Payload size bits: D10-D0: Maximum packet size D12-D11: auditory transaction number per μ frame (only high-speed isochronous and interrupt) 0: No addition (1 transaction / μ frame) 1: 1 (2 transaction / μ frame) 2: 2 (3 transaction / μ frame) 3: Not in use (reserved)	0x0040
bInterval	1	Polling interval for data transfer endpoint Full/low speed interrupt: specify with unit of ms (number of frames) High-speed isochronous/interrupt: specify N for 2 raised to the power of N-1 with unit of μ frame (for example, 1 polling in 8 μ frames when bInterval is 4) Full-speed isochronous: specify N for 2 raised to the power of N-1 with unit of 1ms High-speed bulk/control: specify the maximum NAK rate for endpoint with unit of μ frame 0 means that it does not respond NAK on OUT/DATA transaction	0x00

Table 26: Endpoint Descriptor Serial Conversion Sample

Field	Size	Description	Value
bLength	1	Descriptor size	0x07
bDescriptor	1	Descriptor type	0x05
bString	41	Language Code: 0x09 0x04 Manufacture: "NEC Electronics Co."	

Table 27: String Descriptor Mass Storage Sample

12. Cables

12.1 USB interface cable (Mini-B type)

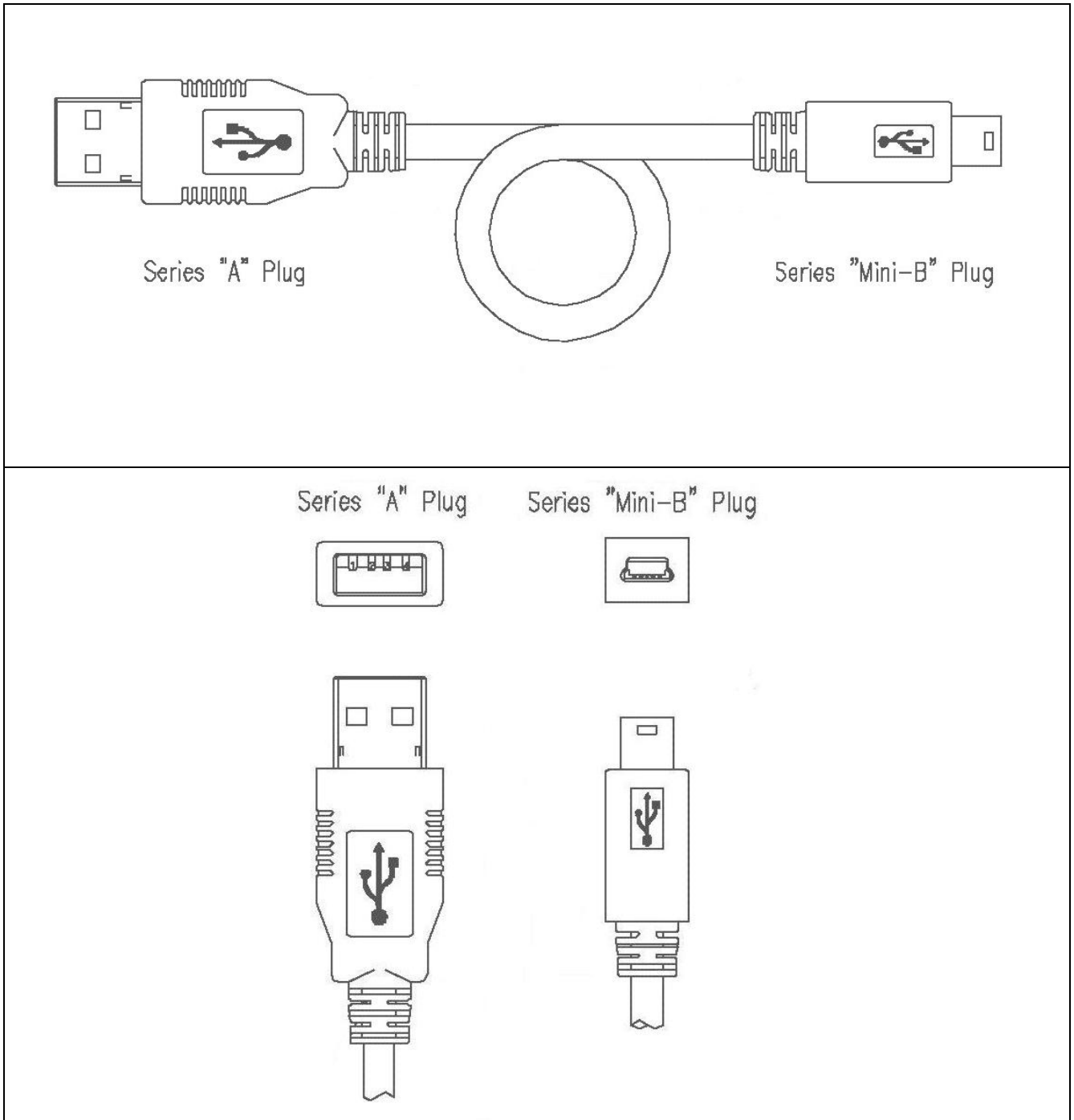


Figure 54: USB interface cable (Mini-B type)

13. Schematics

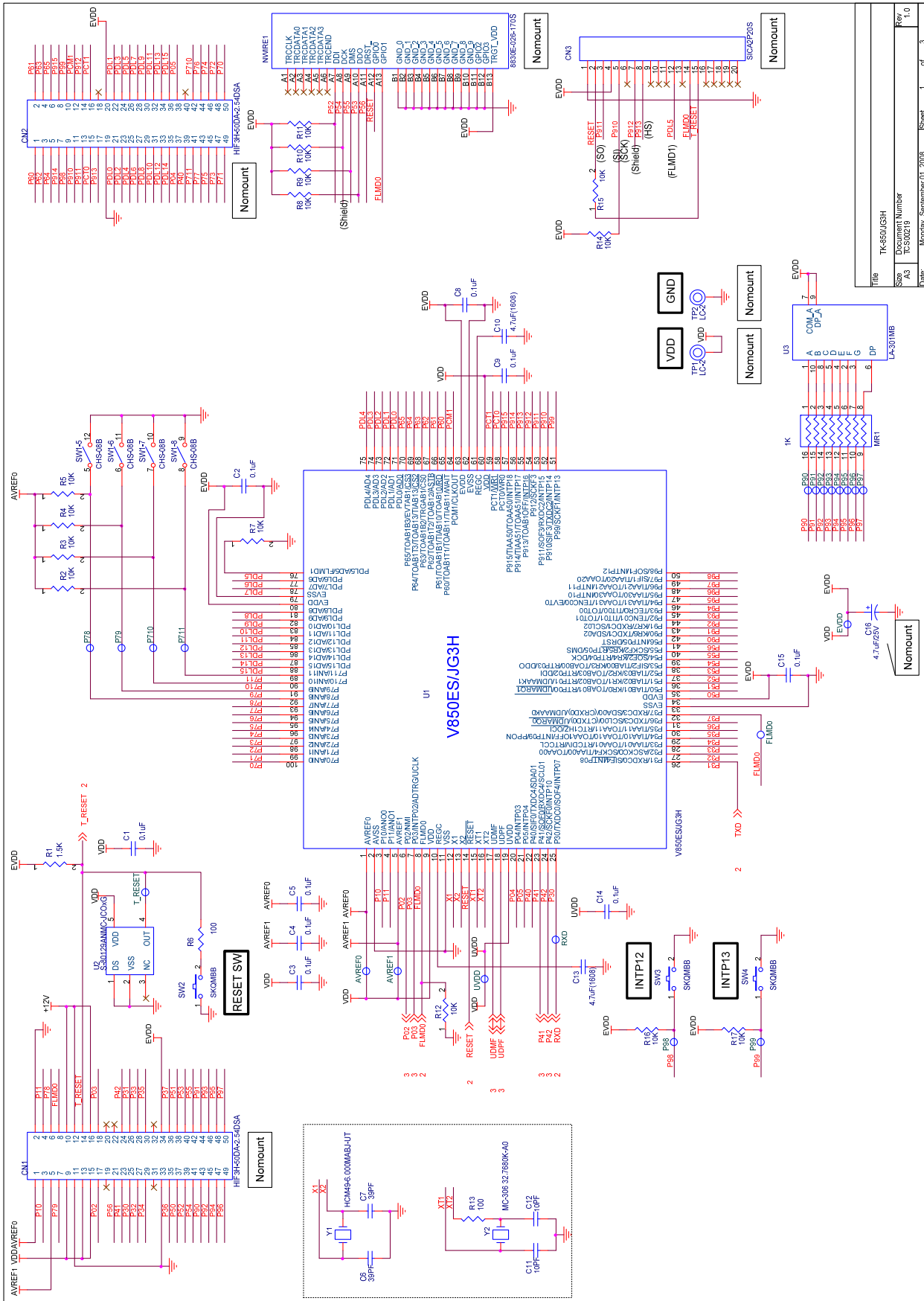
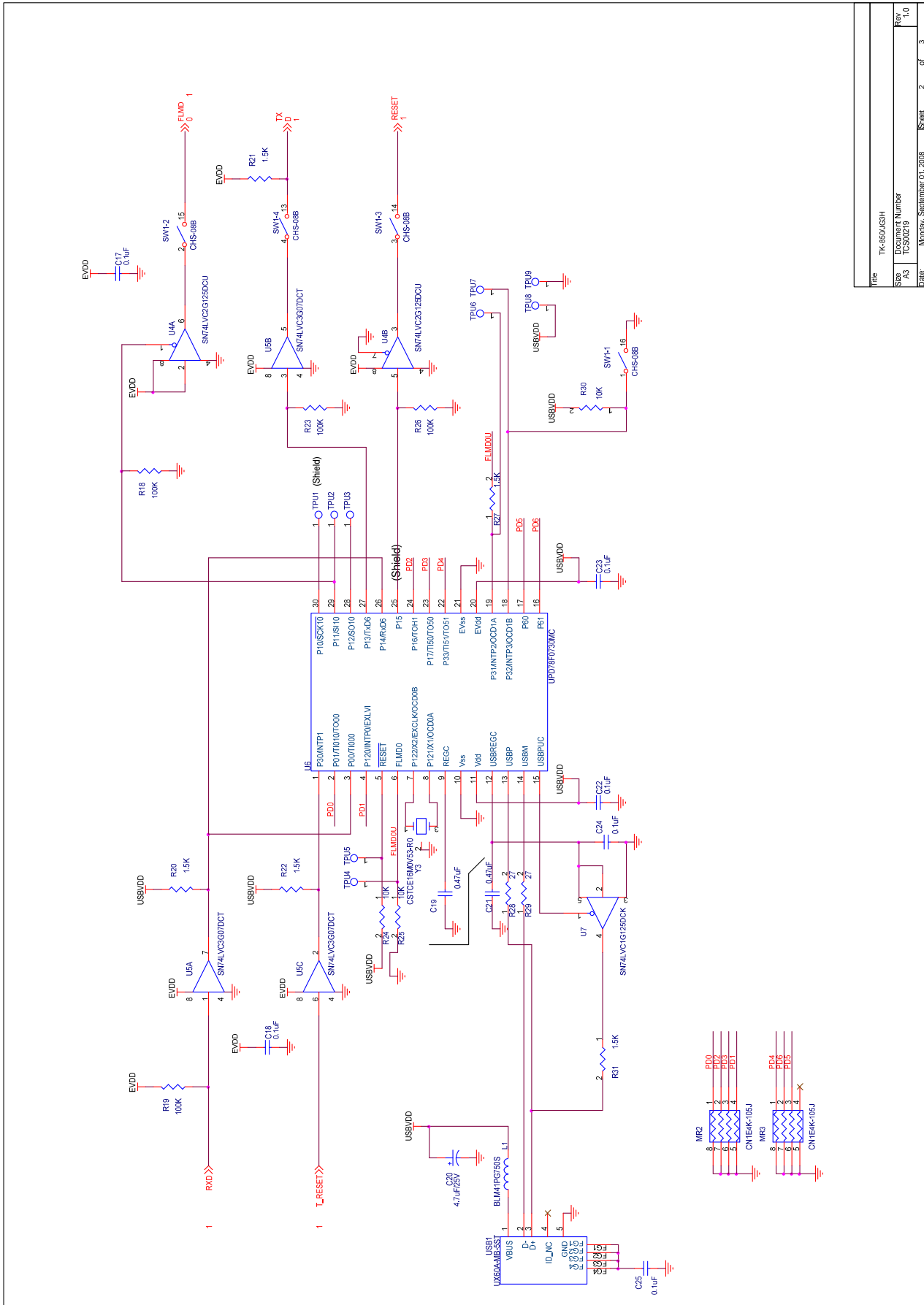
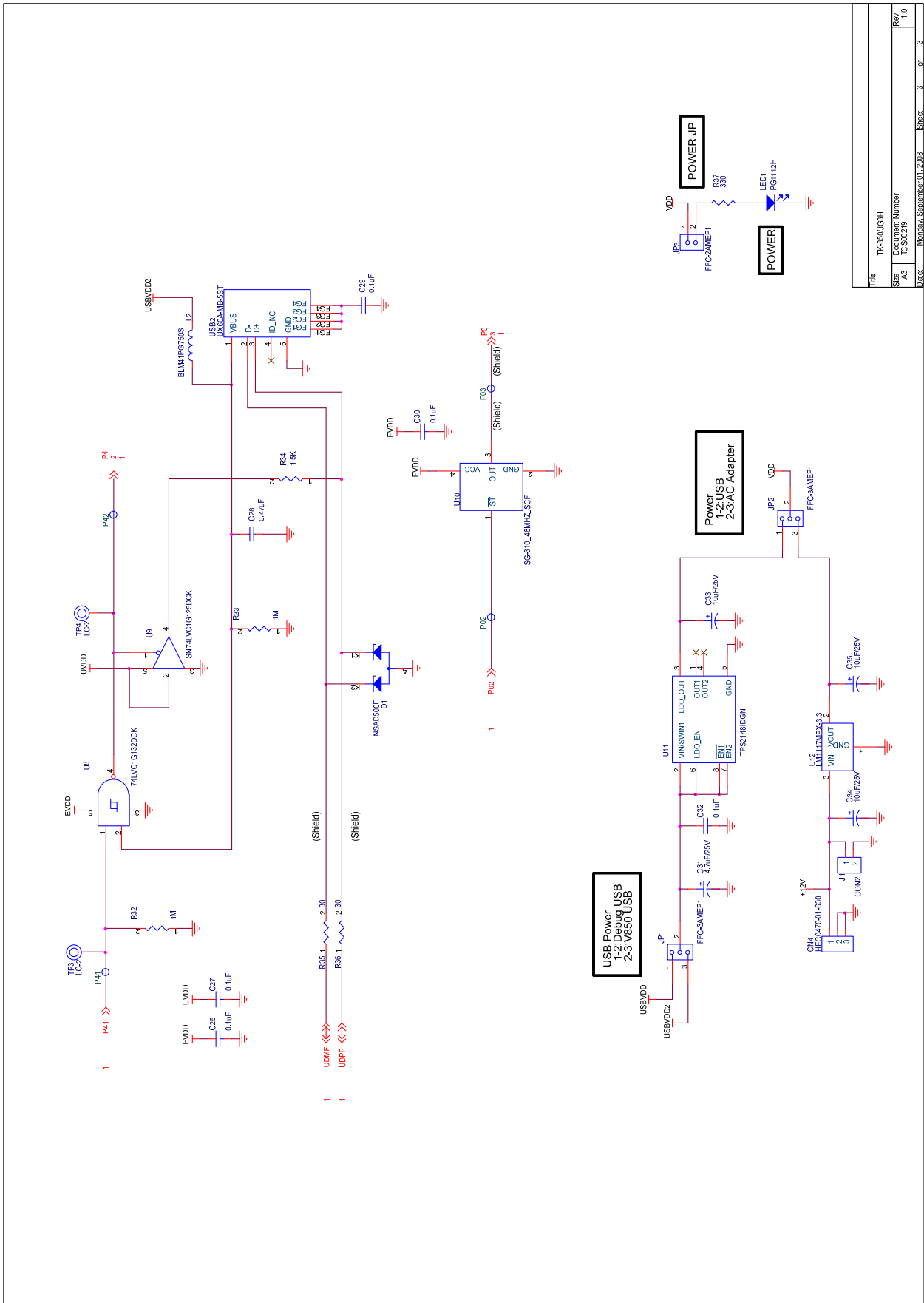


Figure 55: V850ES/Jx3H - Plug it! schematics 1/3



File	TK-850JG3H
Doc Number	TC85219
Size	A3
Date	Monday, September 01, 2008
Sheet	2 of 3
Rev	1.0

Figure 56: V850ES/Jx3-H - Plug it! schematics 2/3



Title	TK-850/CSH
Sheet	DocuSheet Number
PAGE	TC-850218
Page	Monday, September 01, 2008
Sheet	3
of	3
Rev	1.0

Figure 57: V850ES/Jx3-H - Plug it! schematics 3/3

[MEMO]