To our customers,

Old Company Name in Catalogs and Other Documents

On April 1st, 2010, NEC Electronics Corporation merged with Renesas Technology Corporation, and Renesas Electronics Corporation took over all the business of both companies. Therefore, although the old company name remains in this document, it is a valid Renesas Electronics document. We appreciate your understanding.

Renesas Electronics website: http://www.renesas.com

April 1st, 2010
Renesas Electronics Corporation

Issued by: Renesas Electronics Corporation (http://www.renesas.com)

Send any inquiries to http://www.renesas.com/inquiry.

Notice

1. All information included in this document is current as of the date this document is issued. Such information, however, is subject to change without any prior notice. Before purchasing or using any Renesas Electronics products listed herein, please confirm the latest product information with a Renesas Electronics sales office. Also, please pay regular and careful attention to additional and different information to be disclosed by Renesas Electronics such as that disclosed through our website.
2. Renesas 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 Renesas Electronics products or technical information described in this document. No license, express, implied or otherwise, is granted hereby under any patents, copyrights or other intellectual property rights of Renesas Electronics or others.
3. You should not alter, modify, copy, or otherwise misappropriate any Renesas Electronics product, whether in whole or in part.
4. Descriptions of circuits, software and other related information in this document are provided only to illustrate the operation of semiconductor products and application examples. You are fully responsible for the incorporation of these circuits, software, and information in the design of your equipment. Renesas Electronics assumes no responsibility for any losses incurred by you or third parties arising from the use of these circuits, software, or information.
5. When exporting the products or technology described in this document, you should comply with the applicable export control laws and regulations and follow the procedures required by such laws and regulations. You should not use Renesas Electronics products or the technology described in this document for any purpose relating to military applications or use by the military, including but not limited to the development of weapons of mass destruction. Renesas Electronics products and technology may not be used for or incorporated into any products or systems whose manufacture, use, or sale is prohibited under any applicable domestic or foreign laws or regulations.
6. Renesas Electronics has used reasonable care in preparing the information included in this document, but Renesas Electronics does not warrant that such information is error free. Renesas Electronics assumes no liability whatsoever for any damages incurred by you resulting from errors in or omissions from the information included herein.
7. Renesas Electronics products are classified according to the following three quality grades: “Standard”, “High Quality”, and “Specific”. The recommended applications for each Renesas Electronics product depends on the product’s quality grade, as indicated below. You must check the quality grade of each Renesas Electronics product before using it in a particular application. You may not use any Renesas Electronics product for any application categorized as “Specific” without the prior written consent of Renesas Electronics. Further, you may not use any Renesas Electronics product for any application for which it is not intended without the prior written consent of Renesas Electronics. Renesas Electronics shall not be in any way liable for any damages or losses incurred by you or third parties arising from the use of any Renesas Electronics product for an application categorized as “Specific” or for which the product is not intended where you have failed to obtain the prior written consent of Renesas Electronics. The quality grade of each Renesas Electronics product is “Standard” unless otherwise expressly specified in a Renesas Electronics data sheets or data books, etc.

   “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.

   “High Quality”: 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; medical equipment or systems for life support (e.g. artificial life support devices or systems), surgical implantations, or healthcare intervention (e.g. excision, etc.), and any other applications or purposes that pose a direct threat to human life.
8. You should use the Renesas Electronics products described in this document within the range specified by Renesas Electronics, especially with respect to the maximum rating, operating supply voltage range, movement power voltage range, heat radiation characteristics, installation and other product characteristics. Renesas Electronics shall have no liability for malfunctions or damages arising out of the use of Renesas Electronics products beyond such specified ranges.
9. Although Renesas Electronics endeavors to improve the quality and reliability of its products, semiconductor products have specific characteristics such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Further, Renesas Electronics products are not subject to radiation resistance design. Please be sure to implement safety measures to guard them against the possibility of physical injury, and injury or damage caused by fire in the event of the failure of a Renesas Electronics product, such as safety design for hardware and software including but not limited to redundancy, fire control and malfunction prevention, appropriate treatment for aging degradation or any other appropriate measures. Because the evaluation of microcomputer software alone is very difficult, please evaluate the safety of the final products or system manufactured by you.
10. Please contact a Renesas Electronics sales office for details as to environmental matters such as the environmental compatibility of each Renesas Electronics product. Please use Renesas Electronics products in compliance with all applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive. Renesas Electronics assumes no liability for damages or losses occurring as a result of your noncompliance with applicable laws and regulations.
11. This document may not be reproduced or duplicated, in any form, in whole or in part, without prior written consent of Renesas Electronics.
12. Please contact a Renesas Electronics sales office if you have any questions regarding the information contained in this document or Renesas Electronics products, or if you have any other inquiries.

(Note 1) “Renesas Electronics” as used in this document means Renesas Electronics Corporation and also includes its majority-owned subsidiaries.

(Note 2) “Renesas Electronics product(s)” means any product developed or manufactured by or for Renesas Electronics.

8

ALE300L

H8/300L Series Low-Cost Emulator

Microcomputer Development Environment System

User's Manual

2004.01

Renesas System Solutions Asia Pte. Ltd.
www.rss.renesas.com

Microcomputer Development Environment System

ALE300L
H8/300L Series Low-cost Emulator
User's Manual

ALE300L – Low-cost Emulator for H8/300L Series Microcomputer
User's Manual

Published by : Renesas System Solutions Asia Pte. Ltd.
Date : January 5th, 2004, Version 2.0

Trademarks

(a) General

All brands or product names in this manual are trademarks or registered trademarks of their respective companies or organisation.

(b) Specific

Microsoft MS-DOS is registered trademark.
MS-Windows is a trademark of Microsoft Corporation.
Pentium is a registered trademark of Intel.

IMPORTANT INFORMATION

- **READ this user's manual before using this emulator product.**

- **KEEP the user's manual handy for future reference.**

Do not attempt to use the emulator product until you fully understand its mechanism.

ALExxxx & CExxxx Emulator:
Throughout this document, the term "ALExxxx emulator" & "CExxxx emulator" shall be defined as the ALExxxx or CExxxx emulator, user system interface cable, PC interface board, and optional SIMM memory module produced only by Renesas System Solutions Asia Pte. Ltd. excludes all subsidiary products.

The user system or a host computer is not included in this definition.

Purpose of the Emulator Product:
This emulator product is a software and hardware development tool for systems employing the H8/300L series microcomputer. This emulator product must only be used for the above purpose.

Improvement Policy:
Renesas System Solutions Asia Pte Ltd (hereafter collectively referred to as Renesas) pursues a policy of continuing improvement in design, performance, and safety of the emulator product. Renesas reserves the right to change, wholly or partially, the specifications, design, user's manual, and other documentation at any time without notice.

Target User of the Emulator Product:
This emulator product should only be used by those who have carefully read and thoroughly understood the information and restrictions contained in the user's manual. Do not attempt to use the emulator product until you fully understand its mechanism.
It is highly recommended that first-time users be instructed by users that are well versed in the operation of emulator product.

LIMITED WARRANTY

Renesas warrants its emulator products to be manufactured in accordance with published specifications and free from defects in material and/or workmanship. Renesas, at its option, will repair or replace any emulator products returned intact to the factory, transportation charges prepaid, which Renesas, upon inspection, shall determine to be defective in material and/or workmanship. The foregoing shall constitute the sole remedy for any breach of Renesas warranty. This warranty extends only to you, the original Purchaser. It is not transferable to anyone who subsequently purchases the emulator product from you. Renesas is not liable for any claim made by a third party or made by you for a third party.

DISCLAIMER

RENESAS MAKES NO WARRANTIES, EITHER EXPRESS OR IMPLED, ORAL OR WRITTEN, EXCEPT AS PROVIDED HEREIN, INCLUDING WITHOUT LIMITATION THEREOF, WARRANTIES AS TO MARKETABILITY, MECRCHANTABILITY, FITNESS FOR ANY PARTICULAR PURPOSE OR USE, OR AGAINST INFRINGEMENT OF ANY PATENT. IN NO EVENT SHALL RENESAS BE LIABLE FOR ANY DIRECT, INCIDENTAL OR CONSEQUENTIAL DAMAGES OF ANY NATURE, OR LOSSES OR EXPENSES RESULTING FROM ANY DEFECTIVE EMULATOR PRODUCT, THE USE OF ANY EMULATOR PRODUCT OR ITS DOCUMENTATION, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. EXCEPT AS EXPRESSLY STATED OTHERWISE IN THIS WARRANTY, THIS EMULATOR PRODUCT IS SOLD “AS IS”. AND YOU MUST ASSUME ALL RISK FOR THE USE AND RESULTS OBTAINED FROM THE EMULATOR PRODUCT.

State Law:
Some states do not allow the exclusion or limitation of implied warranty or liability for incidental or consequential damages, so the above limitation or exclusion may not apply to you. This warranty gives you specific legal rights, and you may have other rights which may varies from state to state.

The Warranty is Void in the Following Cases:
Renesas shall have no liability or legal responsibility for any problems caused by misuse, abuse, misapplication, neglect, improper handling, installation, repair or modifications of the emulator product without Renesas's prior written consent or any problems caused by the user system.

Restrictions:
1. Earthing (applies only to manual for Renesas hardware products)
   This hardware is designed for use with equipment that is fully earthed.
   Ensure that all equipments used are appropriately earthed.
   Failure to do so could lead to danger for the operator or damaged to equipments.

2. Electrostatic Discharge Precautions (applies only to manuals for Renesas hardware products)
   This hardware contains devices that are sensitive to electrostatic discharge.
   Ensure appropriate precautions are observed during handling and accessing connections.
   Failure to do so could result in damage to the equipment.

Other Important Things to Keep in Mind:
1. Circuitry and other examples described herein are meant merely to indicate the characteristics and performance of Renesas Technology's semiconductor products. Renesas assumes no responsibility for any intellectual property claims or other problems that may result from applications based on the examples described herein.
2. No license is granted by implication or otherwise under any patents or other rights of any third party or Renesas.
3. MEDICAL APPLICATIONS: Renesas Technology's products are not authorized for use in MEDICAL APPLICATIONS without the written consent of the appropriate officer of Renesas Technology (Asia Sales company). Such use includes, but is not limited to, use in life support systems. Buyers of Renesas Technology's products are requested to notify the relevant Renesas Technology (Asia Sales offices) when planning to use the products in MEDICAL APPLICATIONS.

Figures:
Some figures in this user's manual may show items different from your actual system.

Limited Anticipation of Danger:
Renesas cannot anticipate every possible circumstance that might involve a potential hazard. The warnings in this user's manual and on the emulator product are therefore not all inclusive. Therefore, you must use the emulator product safely at your own risk.

PREFACE

This guide explains how to set-up and use the ALE300L emulator for the H8/300L series of microcomputer.

Section 1 Introduction
Gives an introduction to the system, package, specification and functions.

Section 2 Installation
Explains how to setup and install ALE300L emulator and High-Performance Embedded Workshop (HEW).
Target system connection is illustrated in this section too.

Section 3 Usage Note
Highlights the difference of using the ALE300L and the actual microcomputers.

Section 4 Emulation Functions
Describes various functions used in ALE300L.

Section 5 Tutorial
Provides a step by step guide in using the ALE300L to perform emulation.

Section 6 Troubleshooting
Advises on some basic fault locating methods and commonly make mistakes.

Section 7 Diagnostic
A self-test procedure to determine the working state of ALE300L

Related Manuals:

- H8S, H8/300 series C/C++ Compiler, Assembler, Optimizing Linkage Editor User's Manual
- H8/300L Series Hardware Manual

Table of Contents

Section 1. Introduction 1-1

1.1 OVERVIEW 1-1

1.2 PACKAGE 1-2
1.2.1 Software Components 1-2
1.2.2 Hardware Components 1-2

1.3 SPECIFICATIONS 1-3
1.3.1 Summary of ALE300L Functions 1-4

Section 2. Installation 2-1

2.1 EXPRESS SETUP 2-1

2.2 COMMUNICATION PORT BAUD RATE SELECTION SETTINGS 2-2

2.3 ALE300L PACKAGE USED (DIP SWITCHES SETTING) 2-2

2.4 CONNECTING TO TARGET SYSTEM 2-3
2.4.1 Clock Settings 2-4
2.4.2 User Power Supply 2-5
2.4.3 DIP Switch Settings (for different ALE300L packages) 2-7
2.4.4 User Cable 2-8
2.4.5 Target Probe 1 & 2 2-9

2.5 HEW (PURE DEBUGGER) FOR ALE300L INSTALLATION 2-10
At the end of the installation, icons for HEW (Pure Debugger) ALE300L will be created into the Start Menu and ready for execution. 2-13
2.5.1 Installation Details 2-13
2.5.2 What Next? 2-14

Section 3. Usage Notes 3-1

3.1 DIFFERENCES BETWEEN THE ALE300L AND ACTUAL MCU 3-1
3.1.1 Items Requiring Special Attention under ALE300L Emulation 3-1
3.1.2 Items Requiring Special Attention under Actual MCU Operation 3-1

3.2 TARGET SYSTEM INTERFACE 3-2
3.2.1 Signal Protection 3-2
3.2.2 User Interface Circuits 3-2
3.2.3 Special Attention for SD97ALE3834, SD98ALE3880, SD00ALE3880R & SD00ALE3800 3-4

Section 4. Emulation Functions 4-1

4.1 EMULATION BY GO COMMAND 4-1

4.2 SINGLE-STEP EMULATION BY STEP COMMAND 4-1

4.3 BREAK FUNCTION 4-2
4.3.1 Combination Breakpoint 4-2

4.4 RUN-TIME MEASUREMENT 4-3

4.5 TRACE 4-3

4.6 DURING RUN STATE 4-3

4.7 RESET CPU 4-3

Section 5. Tutorial 5-1

5.1 INTRODUCTION 5-1
5.1.1 Overview 5-1

5.2 HOW THE TUTORIAL PROGRAM WORKS 5-2

5.3 RUNNING HEW (PURE DEBUGGER) FOR ALE300L 5-5
5.3.1 Selecting the Target 5-5

5.4 SETTING UP THE ALE300L HEW (PURE DEBUGGER) 5-7
5.4.1 Configuring the Platform 5-7
5.4.2 Mapping the Memory 5-8

5.5 DOWNLOADING THE TUTORIAL PROGRAM 5-10
5.5.1 Loading the Object File 5-10
5.5.2 Displaying the Program Listing 5-11

5.6 USING BREAKPOINTS 5-13
5.6.1 Setting a Program Breakpoint 5-13
5.6.2 Executing the Program 5-14
5.6.3 Reviewing the Breakpoints 5-15
5.6.4 Examining Registers 5-16

5.7 STEPPING THROUGH A PROGRAM 5-17
5.7.1 Single Stepping 5-17

5.8 EXAMINING MEMORY AND VARIABLES 5-19
5.8.1 Viewing Memory 5-19
5.8.2 Watching Variables 5-20

5.9 USING THE COMBINATION BREAKPOINT 5-23
5.9.1 Defining a Combination Breakpoint 5-23

5.10 WATCHING LOCAL VARIABLES 5-25

5.11 USING THE TRACE BUFFER 5-26
5.11.1 Displaying the Trace Buffer 5-26

5.12 SAVE THE SESSION 5-27
5.12.1 What Next? 5-27

Section 6. Troubleshooting 6-1

Section 7. Diagnostic 7-1

Appendix A: ALE300L User Connector Pin Configuration A-1

APPENDIX B: ALE300L Part Number Look-up Table B-1

APPENDIX C: ALE300L Assembly C-1

APPENDIX D: ALE300L User Connector Layout..... D-1

APPENDIX E: Frequently Asked Questions Ver1.2 .....E-1

Renesas Technology (Asia Sales Offices)

Section 1. Introduction

1.1 Overview

The **ALE300L** emulator is the ALE series of real-time, ease-of-use, cost-effective support tools for Asia market. ALE300L is specially designed to aid both hardware and software designers to develop their target system. It is part of the range of PC-based support tools for the H8/300L series microcomputer.

The supporting PC control software, HEW (Pure Debugger) for ALE300L, is a superset of HDI inheriting debugging features of ALE300L emulator with the enhancement of building, compiling and debugging, a all in one development environment (compiler cum debugger tool). The Graphical User Interface (GUI) applied in HEW is performed in a familiar windows environment. It encapsulates the usual User-friendly environment of a general MS-Windows based program, providing on-line help and emulation commands with clicks of the mouse buttons.

Figure 1-1 ALE300L

1.2 Package

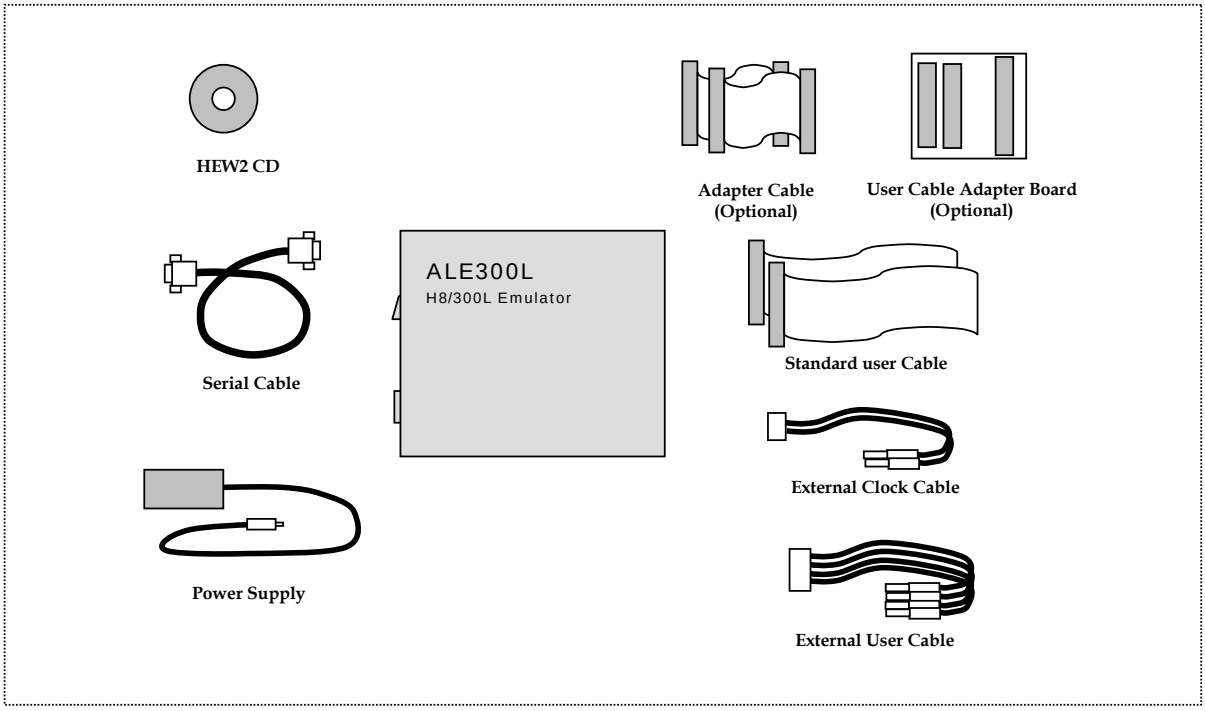


Figure 1-2 ALE300L Package

1.2.1 Software Components

The software components included in the package are listed below.

- 1 x HEW (Pure Debugger) Installation CD
  - Setup.exe <- Click to Install HEW (Pure Debugger)

1.2.2 Hardware Components

The hardware components included in the package are listed below.

- 1 x ALE300L Emulator
- 2 x ALE Standard User cables
- 2 x ALE User Adapter cables (Optional)
- 1 x Serial cable
- 1 x Power Supply Adapter
- 1 x ALE External User cable (UserVcc, Gnd, Probe1, Probe2)
- 1 x ALE External clock cable (Clk, Gnd)
- 1 x ALE User Cable Adapter Board (Optional)

1.3 Specifications

Table 1-1 ALE300L Specification

ITEM	SPECIFICATION
Supported MCU series	• *H8/3724, H8/3714, H8/3614, H8/3834(U)(S), H8/3814 (U)(S), H8/3644(R), H8/3657, H8/3927, H8/3877(U)(N), H8/3637, H8/3627, H8/3947, H8/3867, H8/3887, H8/3827, H8/3847, H8/3937, H8/3827(R)(S), H8/3847(R)(S), H8/3937R, H8/3802, H8/38024, H8/38004.
Host PC	• Minimum Pentium™ PC or above • Microsoft Windows 3.x / Window 95 / Window 98 / WinNT 4.0/ Window 2000 • One free serial port • Approximately 4 Mbytes of free hard disk space
Host-Interface	• RS232C Serial • Baud rate : 19200, 38400, 57600, 115200 bps
Dimension	• 215 x 210 x 65 mm
Power supply	• Adapter Input : 110/230 VAC 50 Hz • ALE Input : 7 ~ 9 VDC • Current : 1.5A (Max)
Environmental	• Operating Temperature: 10°C to 35°C • Humidity: 30% to 85% RH (no condensation) • Corrosive Gas: None

Note: *please refer to the latest Product Brochure for the supported devices

1.3.1 Summary of ALE300L Functions

Table 1-2 ALE300L Functions

ITEM	SPECIFICATION
Emulation	• Performs real-time emulation of a target program • Performs single step execution • Supports real-time emulation up to the MCU's maximum frequency: 16 MHz (H8/3880R and H8/3800 series) • Displays PC address and emulator mode status (Idle, Run) during run • Modifies and displays MCU registers • Resets MCU
File	• Loads target program (Sysrof or Motorola S-type format) • Saves target program
Memory functions	• Modifies and displays memory contents (including memory mapped peripheral registers) • Dumps a range of memory contents • Fills data with specified pattern • Standard 64K high-speed emulation memory (zero wait state) • Specifies memory attribute: ROM / Guarded
Breakpoint	• 256 maximum PC breakpoints • 1 combination break-point on: > address > data > data access type (Read / Write) > 2 external probes • Write protect break - when writing into the MCU's ROM area • Access-inhibit break - when accessing MCU's Guarded area • Forced break - Host PC ESC key
Trace	• Trace memory size: 48 bits x 32K bus cycles > 24 bit address bus (16 bit valid for H8/300L) > 16 bit data bus > 1 bit user NMI > 2 bit external probe signal • MCU control signal (R/W, operating mode, IACK signal, interrupt flag) • Other emulation controls signals (e.g. LIR) • Displays mnemonics of instructions being executed during emulation. • Searches for specified trace information: address, data
Execution Time Measurement	• Resolution : 0.32 usec • Maximum measuring time : 22.9mins
Single step	• Executes target program in step/s.
Clock selection	• Three kinds of clocks are hardware selectable: > User system clock - via ALE user cable > Emulator internal clock - inside ALE main unit > External clock - via external clock cable
User voltage	• Support : 1.8 ~ 5 Volts (MCU dependent)
Auto ID detect	• Detects the EV chip's ID code

Section 2. Installation

2.1 Express Setup

- Unpack & verify parts as in packing list
- Connect Serial cable to PC Communication Port 1 or 2
- Connect Power supply
- Power on PC & ALE300L
- Startup Window (either Win 3.x or Win95 or Win98 or WinNT 4.0 or Win2000)
- Install HEW - - Run > | CD-ROM drive path | :\Setup.exe (refer to section2.5)
- Startup HEW - - Start button > Programs > High-Performance Embedded Workshop (Pure Debugger)
  High-Performance Embedded Workshop
- OR - Run > | Installed path | :\HEW2.exe

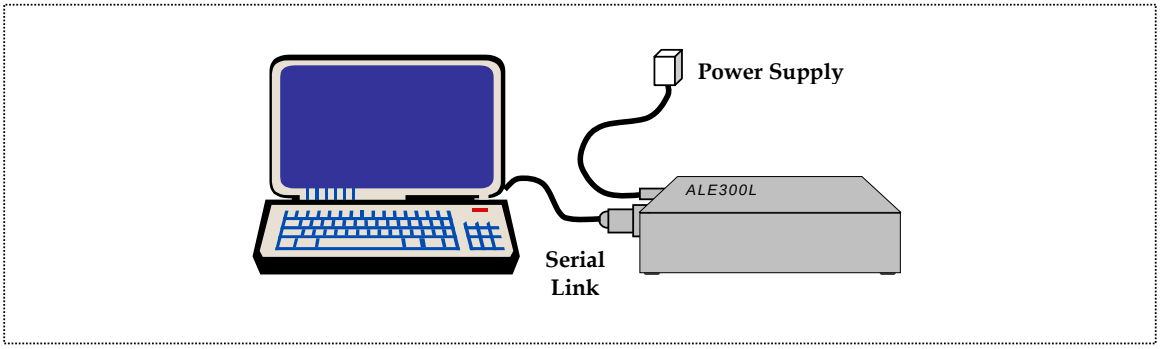


Figure 2-1 Basic setup of ALE300L

Note: Please refer to Session 6. Troubleshooting or
the following for the detailed setup if problems are encountered.

The setting up of ALE300L is made simple.
Users are reminded to note ***FOUR*** main points.

1. ***Communication setting with PC***
2. ***ALE300L package used***
3. ***Target system connection***
4. ***HEW (Pure Debugger) operation***

Note: In order to setup certain features of the ALE300L, the casing may have to be dis-assembled.
Detailed steps are illustrated in Appendix C. Users are reminded to assemble ALE300L totally, before any operation.

2.2 Communication Port Baud Rate Selection Settings

Table 2-1 Baud rate setting

Baud Rate	S1-1	S1-2
19200	Off	On
38400	On	Off
57600	Off	Off
115200 *	On	On

Note: * default setting.

When a new baud-rate is selected, ALE300L has to be power-up again.

115200 bps is not supported by Win 3.x, unless a 3rd party communication driver is used.

Baud Rate DIP switch is located on main board, in order to change the baud rate, the casing must be dis-assembled (refer to Appendix C).

2.3 ALE300L package used (DIP Switches setting)

Different ALE300L Packages will require different DIP Switches setting. These setting are fixed and verified before shipment to the users. However, some ALE300L packages do not have certain DIP switches, e.g. SD97ALE3734 do not have DIP SW 1-4 and SD98ALE3880, SD00ALE3800 do not have DIP SW 2-4.

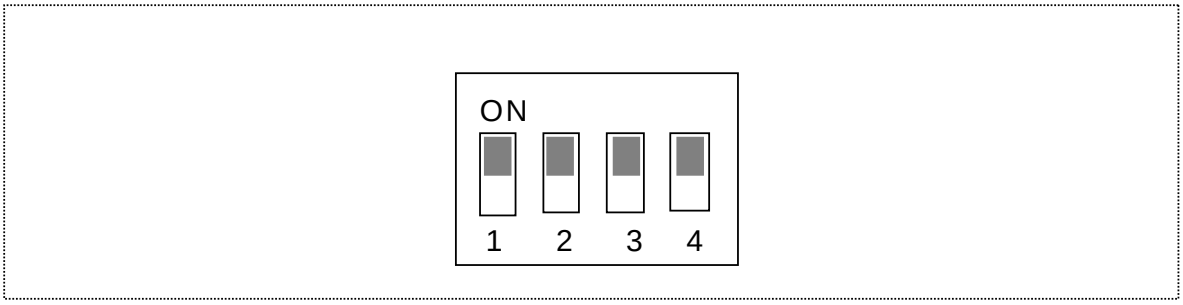


Figure 2-2 DIP Switch Labeling

Note: The different DIP-switches settings are given in section 2.4.3.

2.4 Connecting to Target System

The main considerations are

1. Setting of clock
2. Required User voltage
3. DIP-switch setting
4. Type of User cable

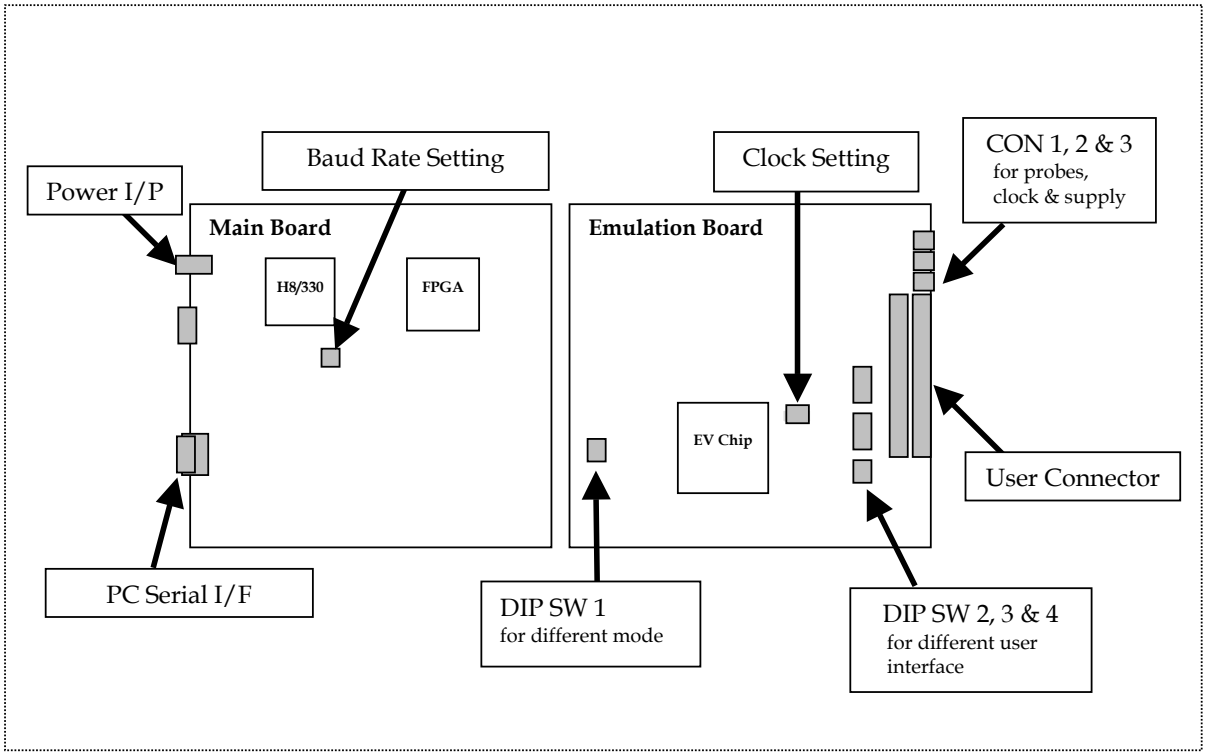


Figure 2-3 Layout of Switches & Connector

Note: There are NO DIP switches 1-4 in SD97ALE3734.
There is NO DIP switches 2-4 in SD98ALE3880, SD00ALE3880R and SD00ALE3800.
There is NO CON3 for SD00ALE3800.

2.4.1 Clock Settings

The system can be configured to use either the Emulator Internal clock, the External clock or the Target System clock. The jumper settings are given below.

Table 2-2 Jumper Settings for clock selections

Sub Clock Selection	INT	TGT
Emulator Internal clock (32.768KHz)*	On	Off
External clock	Off	On

Main Clock Selection	INT	EXT	TGT
Emulator Internal clock (16/10/4MHz)[1, 2]	On	Off	Off
External clock	Off	On	Off
Target system clock	Off	Off	On

Note:
1 Default setting using Emulator Internal clock
2 Maximum Operating clock depends on MCU's specification
The voltage limit of the external clock shall be limited within the UserVcc.
Sub-Clock selection is available in SD98ALE3880, SD00ALE3880R and SD00ALE3800 only.
The Oscillator frequency for the internal clock found in the emulator is halved.

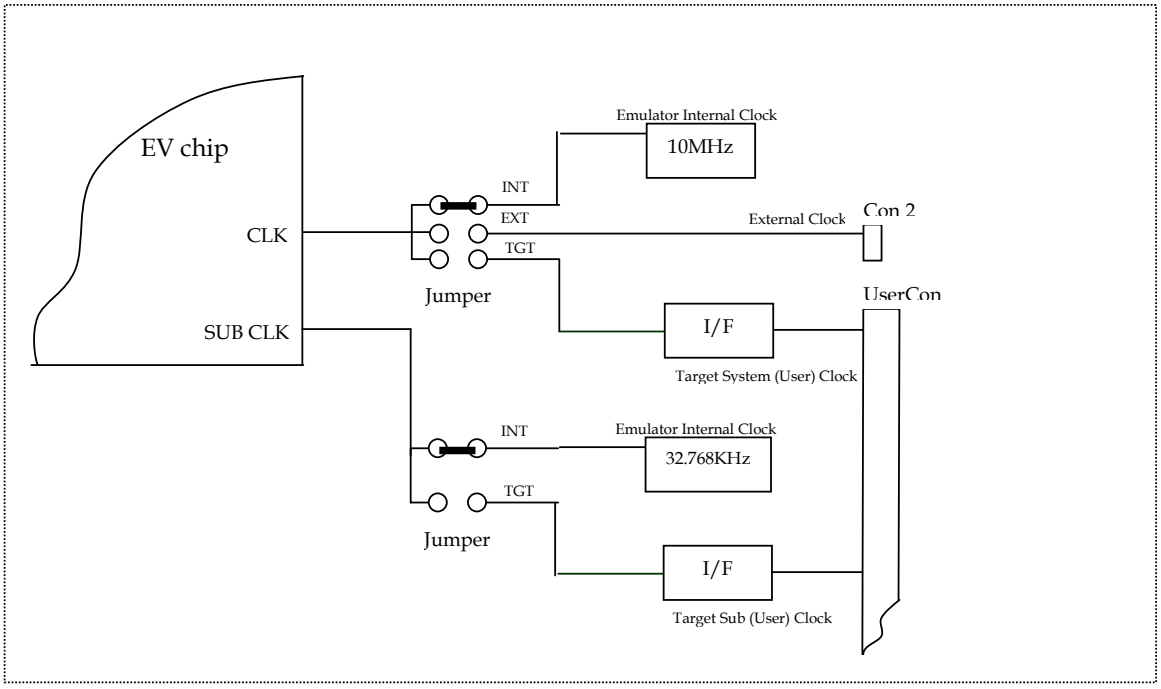


Figure 2-4 Clock input setting

2.4.2 User Power Supply

This is the actual supply (UserVCC) that user operates with its target application. There are three different ways of connecting the user power terminal:

Note : 5V is output from emulator, &
UserVCC is the supply input to the emulator where the operating voltage of user target system is based.

A. Without Target connected

The emulated MCU is running at 5 Volts, the emulator internal supply.

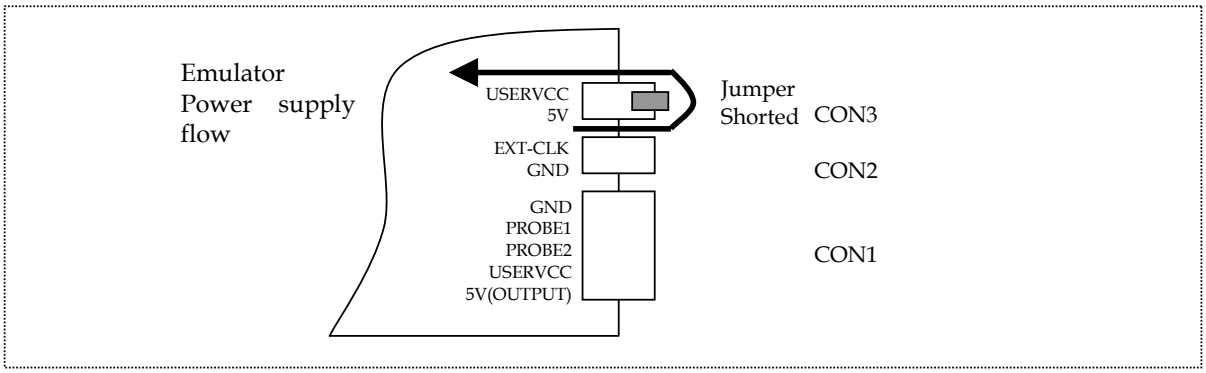


Figure 2-5 Without Target connected [Except SD00ALE3800]

The emulated MCU is running at 5 Volts, the emulator internal supply input by Auto-Power Switching circuitry on the emulation board.

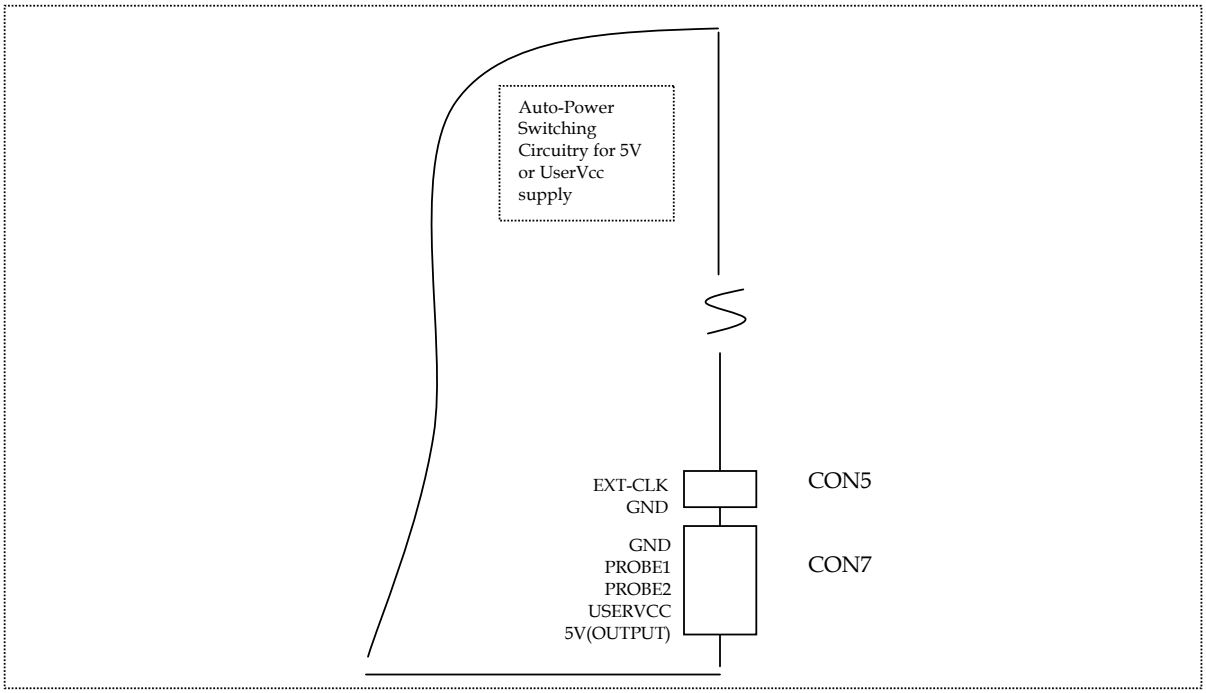


Figure 2-6 Without Target connected [For SD00ALE3800]

B. With Target Connected

B1. USING TARGET POWER SUPPLY

- The emulated MCU and target system will be running at UserVCC. ie. Drawing power from User target system. The voltage range of UserVCC will depend on the actual MCU's specification. Generally, ALE300L can support 1.8 - 5 Volts.

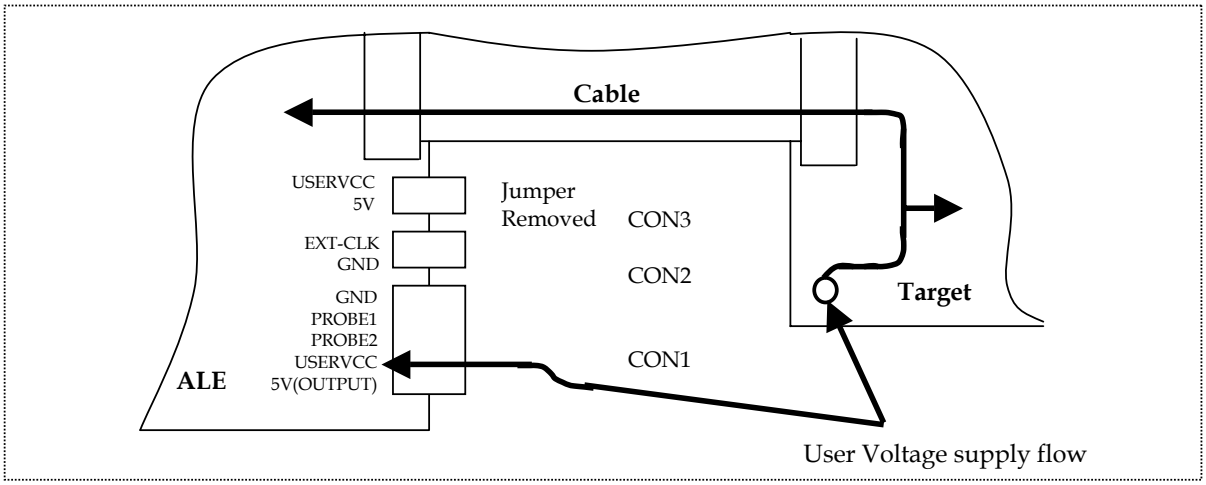


Figure 2-7 With Target connected running at UserVcc [Except SD00ALE3800]

Note: Ensure that the jumper is removed if a target voltage (1.8~ 5 Volts) is input into UserVcc pin. If otherwise, UserVcc will be shorted to VCC (5 Volts), which may cause damages to the emulator.

- The emulated MCU and target system will be running at UserVCC. ie. Drawing power from User target system. The voltage range of UserVCC will depend on the actual MCU's specification. Generally, ALE300L can support 1.8 - 5 Volts.

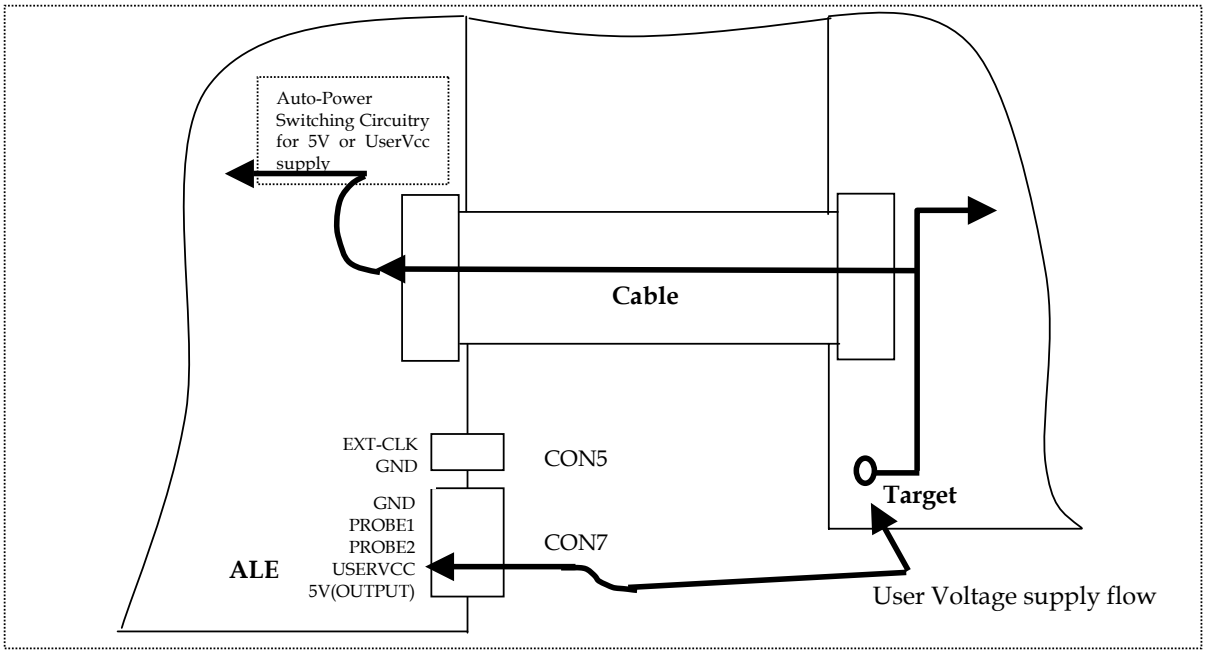


Figure 2-8 With Target connected running at emulator supply, 5V [For SD00ALE3800]

2.4.3 DIP Switch Settings (for different ALE300L packages)

The DIP-switches are used to configure the ALE300L, so as to emulate different MCU. These DIP-switches are set before shipment to users.

Table 2-3 DIP switch SW1 settings

Pin No.	SD97ALE3834	SD97ALE3876 SD98ALE3877R	SD97ALE3927	SD97ALE3947	SD97ALE3644	SD00ALE3880R	SD00ALE3800
1	On	On	On	On	On	Off	On
2	*On	On	Off	X	Off	*On	*On
3	On	On	Off	On	Off	Off	On
4	X	Off	X	X	X	On	On

Note: * On -- use V1, V2, V3 (refer to section 3.2.3)
Off – use PA4, PA5, PA6

Table 2-4 DIP switch SW2 settings

Pin No.	SD97ALE3834	SD97ALE 3876 SD98ALE3877R	SD97ALE 3927	SD97ALE 3947	SD97ALE3644
1	On	On	Off	Off	Off
2	On	Off	On	On	On
3	On	Off	On	On	On
4	Off	Off	Off	On	Off
5	On	Off	On	On	On
6	Off	Off	Off	Off	Off
7	On	On	On	On	On
8	Off	On	Off	On	Off

Table 2-5 DIP switch SW3 settings

Pin No.	SD97ALE3834	SD97ALE 3876 SD98ALE3877R	SD97ALE 3927	SD97ALE 3947	SD97ALE3644
1	On	On	Off	Off	Off
2	On	On	Off	Off	Off
3	On	On	On	On	On
4	Off	Off	Off	Off	Off
5	Off	On	On	Off	On
6	Off	On	On	Off	On
7	On	On	On	On	On
8	Off	Off	Off	Off	Off

Table 2-6 DIP switch SW4 settings

Pin No.	SD97ALE3834	SD97ALE 3876 SD98ALE3877R	SD97ALE 3927	SD97ALE 3947	SD97ALE3644
1	Off	Off	Off	Off	Off
2	On	On	On	On	On
3	Off	Off	Off	Off	Off
4	Off	On	On	Off	On

2.4.4 User Cable

There are three types of user cable.
The following demonstrates the various ways of connecting the user cables.

ALE Standard User Cable
ALE300L uses the readily available standard cable (0.05″ pitch) to connect to the users target system. The pins assignment of the ALE user cables is listed in Appendix A. And the layout of the user connectors is illustrated in Appendix D. There are two types of ALE standard user cables:

1. 50-pins type (ALE-50P)
2. 60-pins type (ALE-60P)

ALE User Cable
Several types of popular low cost user-cable are available. This cable can be plugged onto the actual MCU footpint (A IC socket must be used)

E1000/E6000 User Cable
E1000/ E6000 user cables support the whole MCU series. ALE300L supports the E1000/E6000 User Cable via an ALE User Cable Adaptor.

Table 2-7 User cables support

Microcomputer series	ALE Standard User Cable	User Cable Adapter
H8/3734	ALE-50P	E1000
General	ALE-60P	E1000
H8/3644	ALE-60P	E6000
Others	ALE-60P	-

Note: Please refers to your nearest Sales Office to obtain the latest part number.

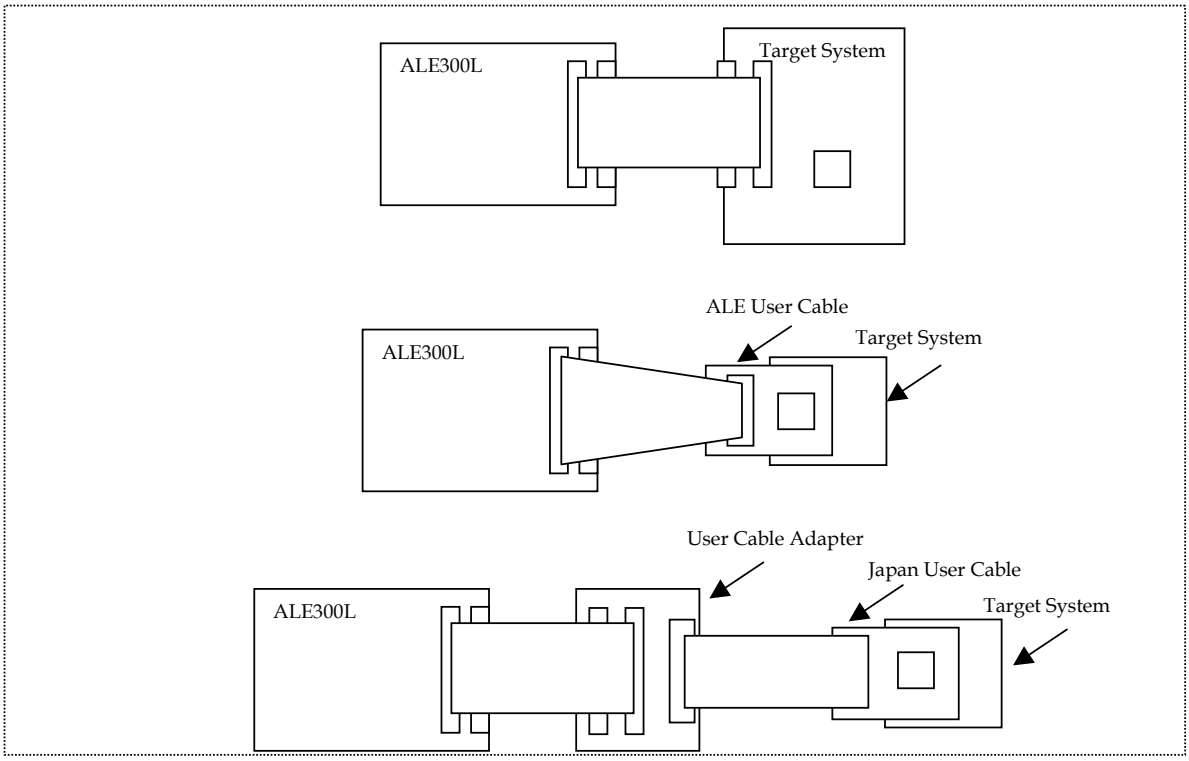


Figure 2-9 Various User Cable Connection methods

2.4.5 Target Probe 1 & 2

Two external probes located at CON 1 are used to generate EVENT Break (refer to Section 5 for the setting of Combinational Break-Signal). When these probes are set accordingly, its status will cause the running user program to halt/break. The voltage level of these inputs is to be limited within the UserVCC.

The Probes are internally tied high to Vcc through a 4.7 Kohm resistor.

2.5 HEW (Pure Debugger) for ALE300L Installation

To install the HEW (Pure Debugger) for ALE300L from the installation disk, proceed as follows:

- Insert the HEW (Pure Debugger) for ALE300L installation CD.
- Run Windows if it is not already running.
- Close all other applications that are running.
- Choose *Run* from the Program Manager File menu.
- Type *Setup* and click OK

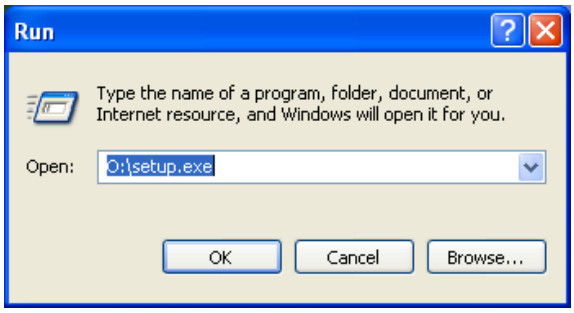


Figure 2-10 Run Dialogue box

This runs the HEW application installer in exe format and the following Welcome! Screen will be displayed:

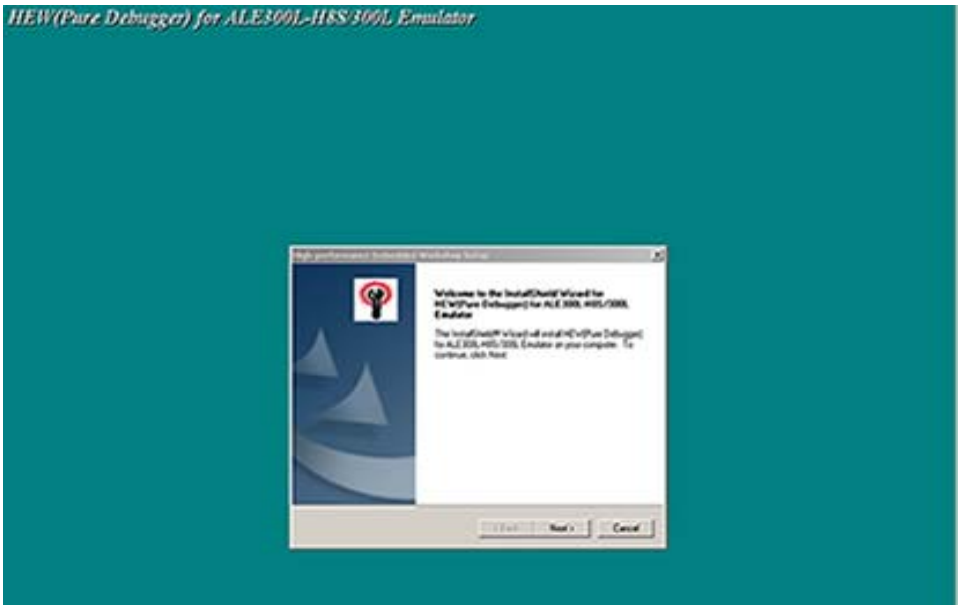


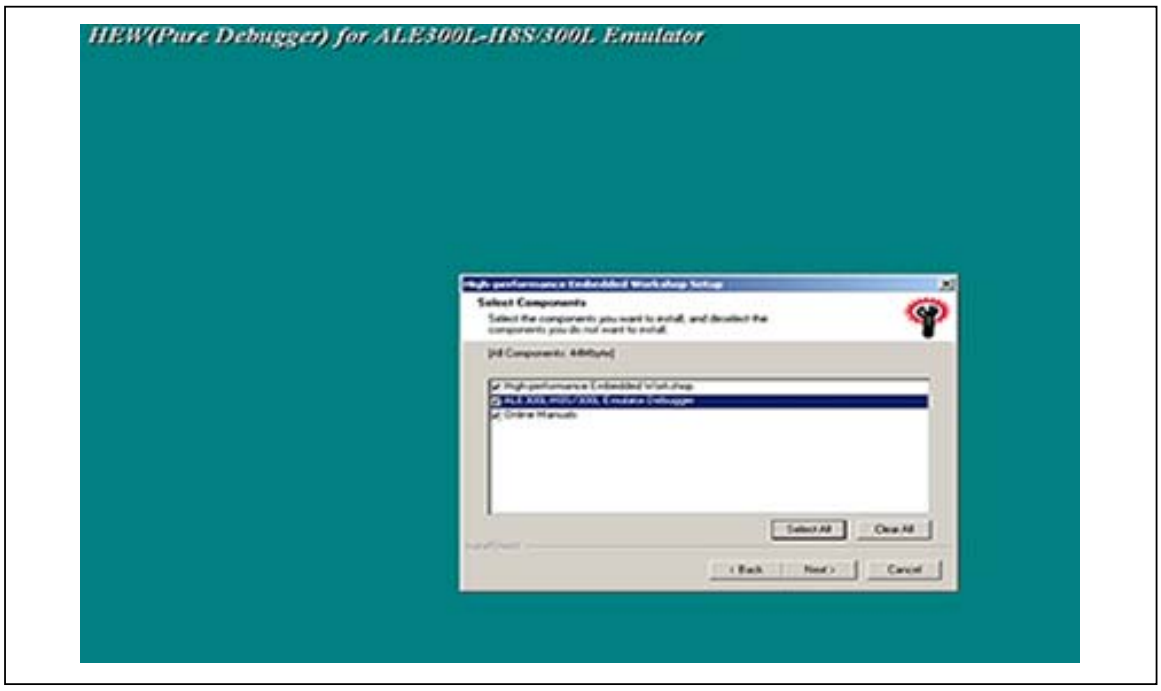
Figure 2-11 HEW for ALE300L Installer Welcome! Screen

- Click on **NEXT >** button to proceed with the installation.
- Next shows the `history-update on HEW for ALE300L` for any important information concerning the installation. Click on **NEXT >** `button` to proceed.

- Check the *License Agreement* concerning installation and then click *Next* to proceed.

Figure 2-12 Update Information Screen

- The following dialogue box enables the selection of directory in which user can install the HEW (Pure Debugger) for ALE300L. Alternatively to specify other directory to install HEW for ALE300L, click on the **Browse** button.



- Click *Next* to install into the default directory *C:\Hew2* or *C:\Program Files\Hew2,* or specify an alternative directory by clicking on Browse-button.

If user chooses *Next*, the following dialogue box will confirm each installation directory you selected [Note: Always ensure that all components are installed in the same required directory]

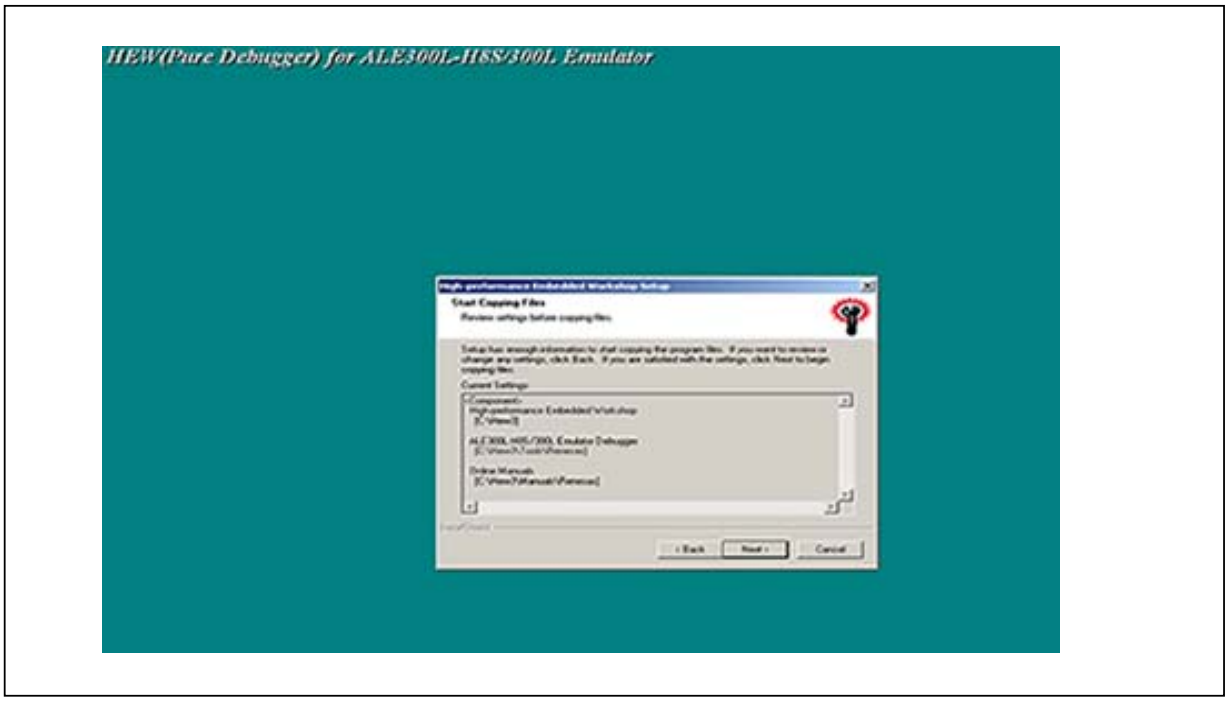


Figure 2-14 Select Program Group Screen

- Click *Next* to begin installation.

The installer then copies the HEW (Pure Debugger) for ALE300L files to the specified directory:

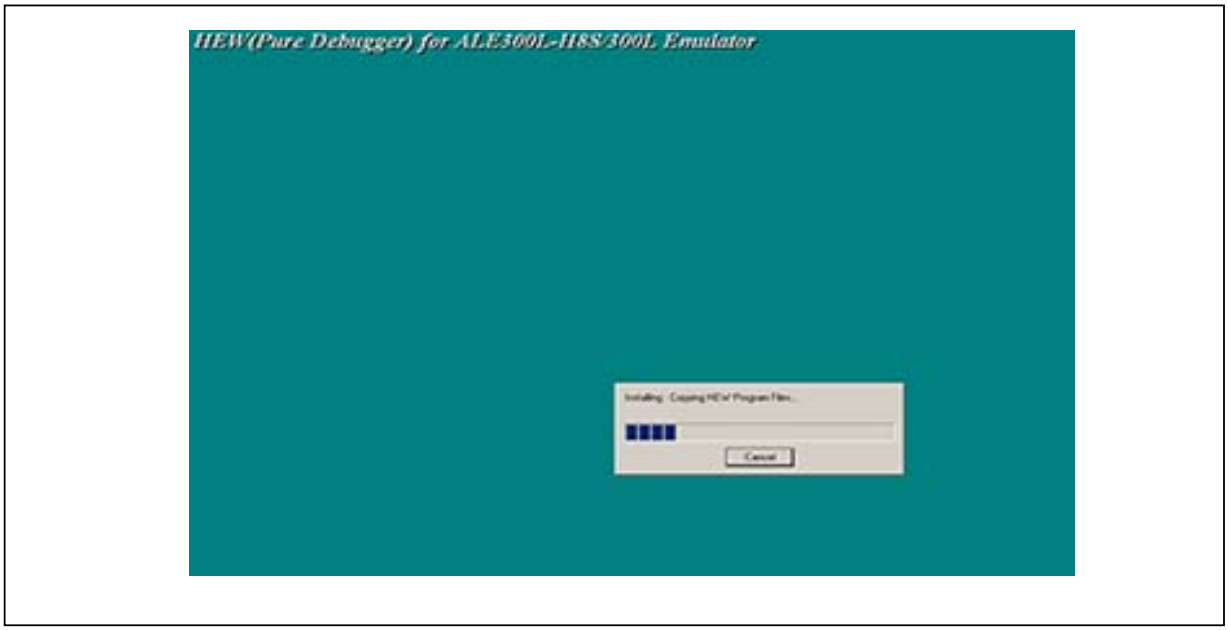


Figure 2-15 Installing Screen

The installation will complete with the Completion screen:

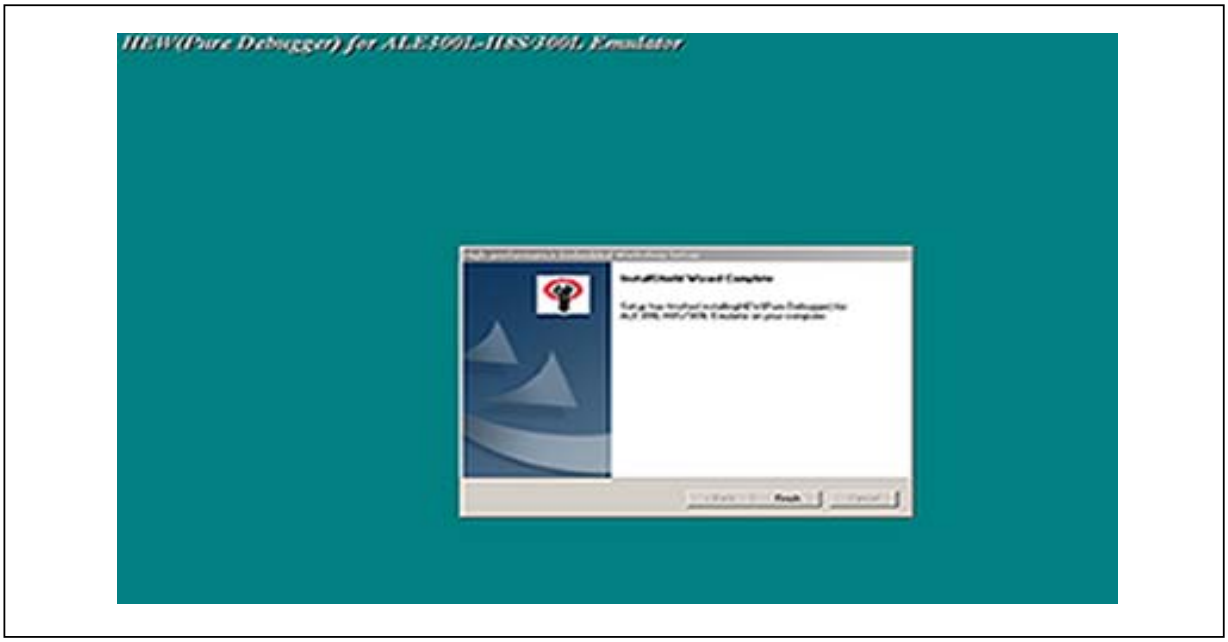


Figure 2-16 Completion Screen

At the end of the installation, icons for HEW (Pure Debugger) ALE300L will be created into the *Start Menu* and ready for execution.

2.5.1 Installation Details

The installer creates the following icons in the program group you specified, by default HEW (Pure Debugger):

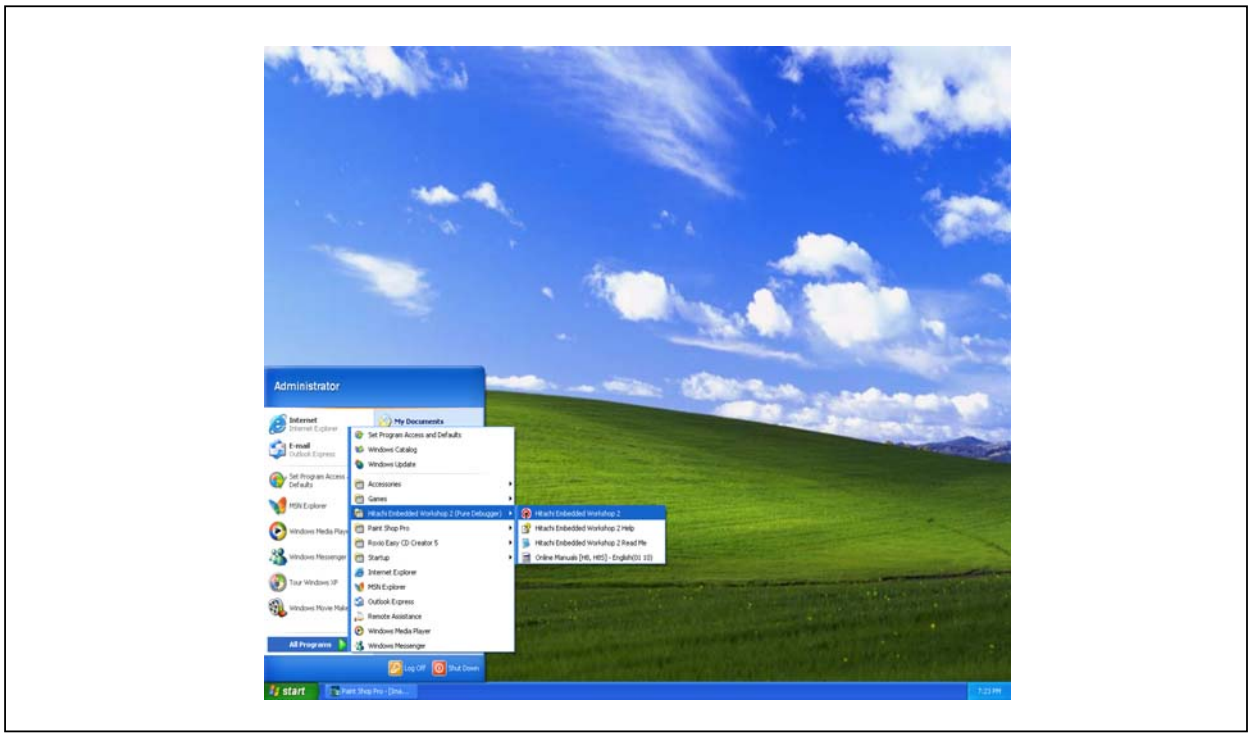


Figure 2-17 HEW(Pure Debugger) Program Startup

- "HEW (Pure Debugger" : Activates HEW for ALE300L
- "Online Manuals" : Leads to a PDF format of available help manuals
- "HEW3 Help" : A HEW application help context

2.5.2 What Next?

The ALE300L emulator is now correctly set up and ready for use. We recommend you work through section 5, *Tutorial,* to familiarize yourself with the key features of the ALE300L emulator, and to learn how to use the ALE300L emulator to develop and debug programs for H8/300L series MCU.

Section 3. Usage Notes

3.1 Differences between the ALE300L and Actual MCU

There may be a few occasions when operations that run correctly under emulation do not do so under the actual MCU, and vice versa. The following examples illustrate how to solve these problems.

3.1.1 Items Requiring Special Attention under ALE300L Emulation

Timing Difference Between the ALE300L and the Actual MCU: When using the ALE300L, a 10 to 20 ns timing delay occurs due to the target system cable and target system interface circuit on the ALE300L.

Sub-Clock: The sub-clock of the H8/300L uses the ALE300L internal clock. The target system's sub-clock is isolated on the target cable (X1 & X2 are open-circuited from ALE). An exceptional case will be the SD98ALE3880 and SD00ALE3800, Users are allowed to input sub-clock.

Clock: The frequency of the input clock to the ALE300L must be twice the actual MCU operating clock. The OSC2 pin is open-circuited from ALE. If user intends to input target clock, a proper generated clock are required. The crystal connection will not be able to cause oscillation to generate the clock.

Stack: when emulation is halted with users' stack pointer (SP) pointing between H'FD80 and H'FF7E, the emulator for internal processing uses 4 bytes of internal RAM. The address of this area is SP-1 to SP-4, these data must not be altered by the user. (This is applicable for devices supported by SD97ALE3734 only).

Auto-Power Switch: An auto-power switch circuitry is implemented on the SD00ALE3800 only.
This design detects for userVCC input from the target system. If there is no presence of userVCC, it will automatically switch to supply 5V, emulator power supply and input to the emulated MCU. However, if userVCC is present, it would switch to userVCC supply and input the emulated MCU.

3.1.2 Items Requiring Special Attention under Actual MCU Operation

Initialization of Stack Pointer (SP): The ALE300L sets the SP to the end of the RAM area (H'FF7E) implicitly. In the MCU, however the contents of the SP are undetermined until set by instruction. In addition, the general registers are set to H'0000 by the ALE300L.

Input pins Left Open: The RES, TEST, and OSC1 input pins are pulled up using the 470-KiloOhm resistor in the ALE300L emulator. If these pins are left opened in the target system, they are internally fixed at high in the ALE300L. Check the above inputs before using the ALE300L. Note that the TEST pin must be grounded if it is left open in the target system.

3.2 Target System Interface

All signals are connected to the evaluation chip in the ALE300L except for the following signals.
OSC2, X1 (except SD98ALE3880 and SD00ALE3800), X2, TEST, CVCC, …

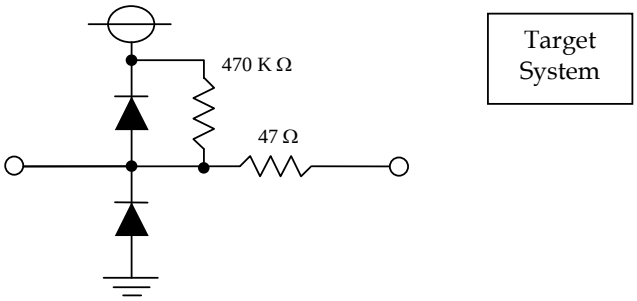
3.2.1 Signal Protection

All user system interface signals are protected from over- or under- voltage by the use of diode array.

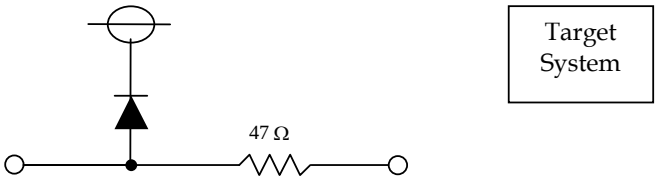
3.2.2 User Interface Circuits

The circuits that interface the evaluation chip in the ALE300L to the user system include buffers and resistors that cause signal delays. Note that when an input terminal is in the high-impedance state, the pull-up resistor forces the terminal to be at high level. Adjust the user system to compensate for these effects.
The following show the user interface signal circuits.

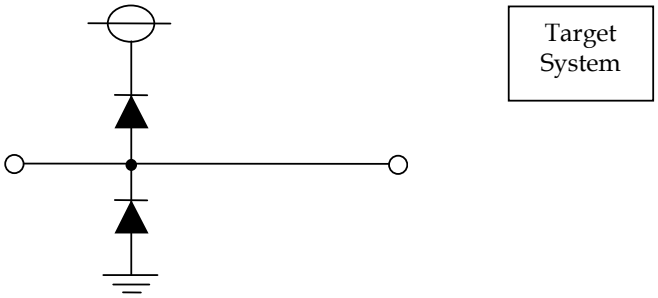
- Normal Input / Output ports - for SD97ALE3724 SD98ALE3880, SD00ALE3880R supported devices only & All /NMI signal



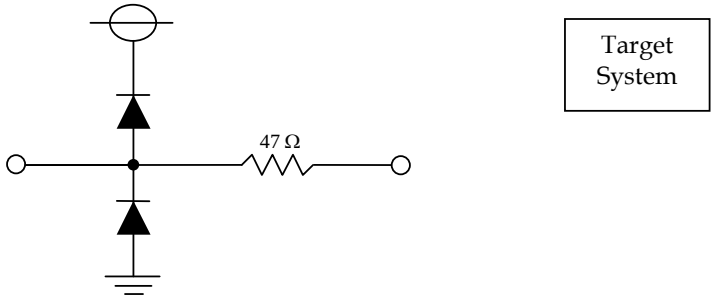
- High Voltage Input / Output ports - for SD97ALE3724 supported devices only



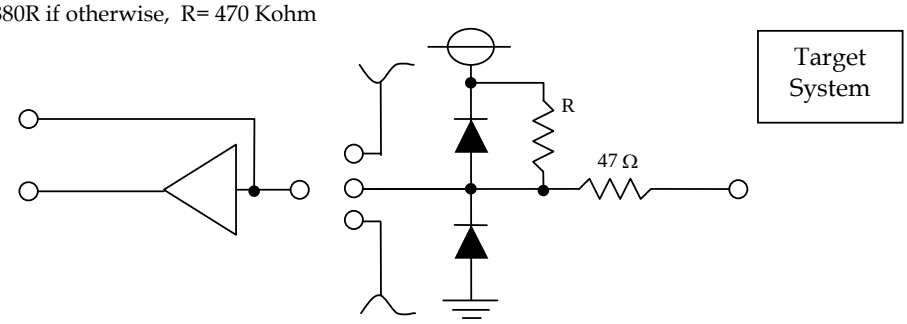
- For all Analog signals (AN?, V?, AVCC, TONED, Vtref…) in all devices except for SD97ALE3724 & SD98ALE3880, SD00ALE3880R supported devices.



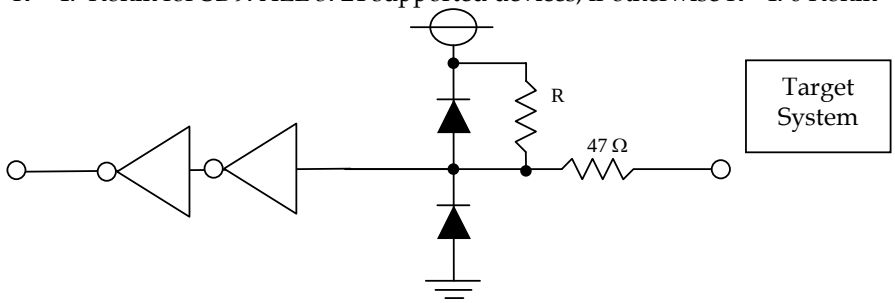
- General Interface for all signals and V? & AN? of SD98ALE3880
  except SD97ALE3724, other SD98ALE3880, SD00ALE3880R signals and SD00ALE3800.



- For OSC1 –> R = 47 Kohm for SD97ALE3724 supported devices, R=10 Kohm for SD98ALE3880, SD00ALE3880R if otherwise, R= 470 Kohm



- For RES –> R = 47 Kohm for SD97ALE 3724 supported devices, if otherwise R= 470 Kohm



- Vcc (except SD00ALE3800)

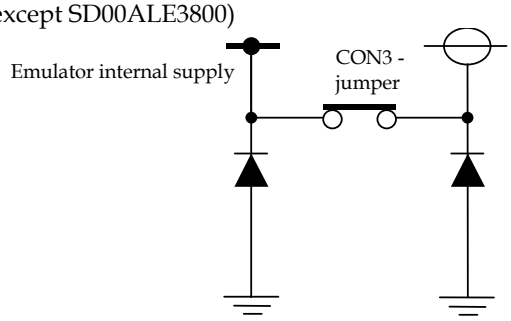


Figure 3-1 Target interface circuitry

Notes:

1. The voltage input to AN0 to AN7 must be within the range between AVss and AVcc.
2. The frequency of the target system clock must be double that of the MCU operation clock.
3. ⊖ indicates ALE300L or target system UserVcc connection.

3.2.3 Special Attention for SD97ALE3834, SD98ALE3880, SD00ALE3880R & SD00ALE3800

3.2.3.1 LCD Function

Devices supported by SD97ALE3834, SD98ALE3880, SD00ALE3880R & SD00ALE3800 have on-chip LCD controller. Thus the default setting allows the pins to be used for the LCD controls signal V1, V2 & V3. If users intend to use the pins as I/O ports PA_4, PA_5 & PA_6, the DIP-switches 1(way 2) has to be changed.

DIP SW 1 Way 2	SD97ALE3834,SD98ALE3880, SD00ALE3880R, SD00ALE3800 Emulated Function
On<default>	V1, V2, V3
Off	PA_4, PA_5, PA_6

Table 3-1 LCD function for SD97ALE3834, SD98ALE3880, SD00ALE3880R & SD00ALE3800

Notes:

1. No built-in power supply split resistance is added to boost the LCD drive power supply in the SD97ALE3834, SD00ALE3880, SD00ALE3880R & SD00ALE3800 emulator. Pls. refer to the microcomputer's hardware manual.

3.2.3.2 User Connector Pin Configuration

The user connector pin configuration attached with the appendix A in this user manual indicates all the resources supported by the evaluation chip. For actual pin configuration, please refer to the hardware manual of each targeted microcomputers.

Section 4. Emulation Functions

4.1 Emulation by Go Command

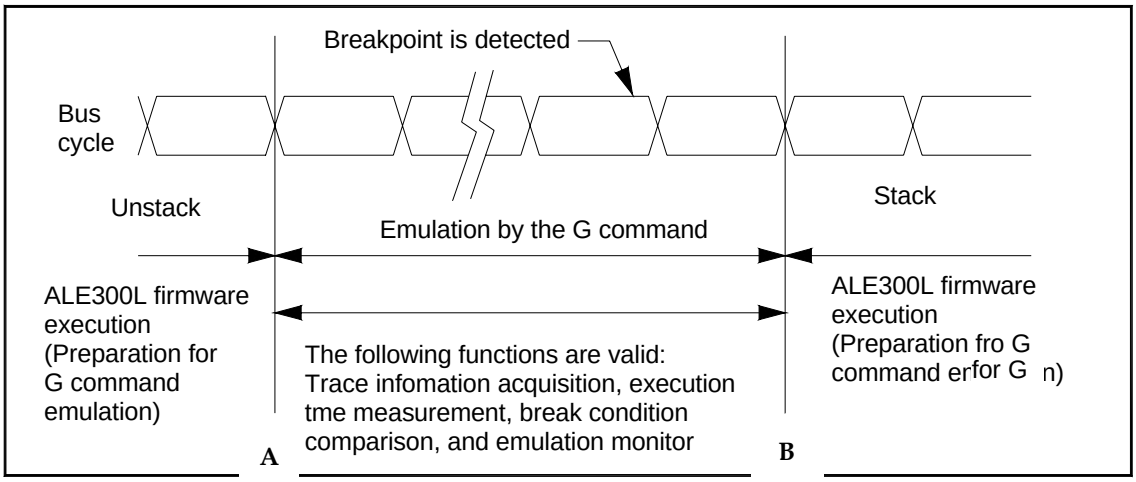


Figure 4-1 Go Command Emulation Timing

The ALE300L executes a firmware program before and after executing the target program. The execution from A to B shown in the above figure without ALE300L firmware is called real-time emulation, which is performed by Go command.

4.2 Single-Step Emulation by Step Command

Bus timing is the same as in the figure above, but the target program is executed only one step at a time. When completing the Step command, ALE300L update all windows. The above sequence is repeated for every single-step execution, which means that no real time emulation is executed with the Step command. No interrupt will be serviced during single stepping.

4.3 Break Function

There are 5 types of break. These break functions are classified into 2 classes, mainly hardware and software break. When the function is implemented in hardware, the condition will cause the break to occur after the instruction is executed, whereas the software implementation will not cause the instruction to be executed at the break condition. For ALE300L, software implementation is done for the PC breakpoint only, the other breaks are implemented using hardware method.

Table 4-1 Lists of types of break that may be encountered during emulation.

	Types of Break	Description
1	Combination break	A break occurs when the CPU agrees with a condition specified by the Combination Breakpoint Window, or when the pre-fetch cycle of the CPU agrees with the specified states.
2	PC Break	A break occurs at the program address specified by PC Breakpoint Window. The instruction at this address is replaced with an illegal instruction beforehand, therefore, the specified address must correspond to the beginning of the instruction. If a PC breakpoint is detected, the emulation stops at the specified address before executing the corresponding instruction.
3	User break	Pressing the ESC key of the PC generates a break..
4	Write protect break	When ROM in the MCU is specified, a write protect break occurs when attempting to write to the ROM area.
5	Invalid area access	A break occurs when an invalid area (outside of internal ROM, RAM, and I/O area) is accessed. A break also occurs if a pre-fetch for an address, which cover into the invalid area.

Notes: 1. Breaks 1 to 5 above are valid during Go command emulation.
2. Breaks 3 to 5 are valid during Step command emulation.

4.3.1 Combination Breakpoint

In ALE300L, there is 1 combination breakpoint, which has a break condition determined by 5 factors.

1. Address
2. Data
3. Access (Read/Write)
4. External Probe 1 (Hi/Lo) and
5. External Probe 2 (Hi/Lo)

The break condition will occurs in a "AND" condition. If either factor is not set, the particular condition will be ignored.

4.4 Run-Time Measurement

There is a built-in 32-bits counter based on a 3.125 MHz clock. This produces a 0.32 micro-second time measurement resolution, which can measure up to 22.9 minutes. Whenever User activates the Run mode, the counter will start counting until a break condition appears.

4.5 Trace

Whenever User program is executed, the instructions and all relevant control signals are latched in the trace buffer. In each Trace cycle, the available displayed data are

1. 24 bits Address
2. 16 bits Data
3. External Probe 1 & 2
4. Read / Write
5. MCU states

The trace buffer contains 32 K Bus cycles. The traced data will be overwritten if 16K(approximately) of instructions are executed.

4.6 During Run state

During run state, useful information, such as current PC address, MCU states and run-time are available visually to the users.

PC address will be displayed to the user at a time interval of 0.5 sec (approximately). Thus it will serve as an indicative measure for the user, to track the position of the program counter. This PC display will not be updated when the MCU is put into low power modes.

MCU states display is

1. Sub-sleep *
2. Watch
3. Sub-active
4. Standby
5. Sleep
6. Active (medium speed mode) *
7. Other / Normal (including reset)

Note: *chip states differ for each product family.

Run-Time will be displayed to the resolution of microseconds.

4.7 Reset CPU

When "RESET CPU" command is send, the following actions happen,

PC = reset vector
CCR = H'80
R0 ~ R7 = no change

Section 5. Tutorial

The following describes a sample debugging session, designed to introduce the main features of the ALE300L emulator used in conjunction with the HEW (Pure Debugger) for ALE300L software.
The tutorial is designed to run in the ALE300L emulator's resident memory so that it can be used without connecting the ALE300L emulator to an external user system. **<Note: This tutorial is meant for H8/3800 device. Please run it with the ALE300L-H8/3800 Low-cost Emulator and please choose device : H8/3800 in the device selection box to run this tutorial properly >**

5.1 Introduction

The 300L_3800_Tut is based on a simple Assembler / C program. It is located in the following directory:
<\\(Your Working Directory)\Tools\Renesas\DebugComp\Platform\Emulator\ALE300L\300L_3800_Tut>

The tutorial is based on a simple C program.
Before reading this chapter:

- Set up the ALE300L emulator and verify that it is working in conjunction with the HEW (Pure Debugger) software. You do not need to connect to the ALE300L emulator to a user system to use this tutorial.
- Make sure you are familiar with the architecture and instruction set of the MCU. For more information refer to the H8/300L Series Programming Manual and H8/300L Series Hardware Manual.

Note: On a first time loading of the tutorial, a dialog box prompting the move of workspace from previous installed directory is displayed. Please click [YES] and the workspace would be configured to the current installed directory permanently.

5.1.1 Overview

This program is an infinite loop that sort elements based on NAME in the alphabetical order, and AGE and ID in the ascending order.
The tutorial is provided on the installation CD. A compiled version of the 300L_3800_Tut is provided in Motorola S-Record in the file 300L_3800_Tut.mot.

5.2 How the Tutorial Program Works

The first part of the program includes a series of header files:

```
#include <machine.h>
#include "\CH38\INCLUDE\string.h"
```

The program then gives prototypes for the constants, structures, and function initial values:

```
#define NAME    (short)0
#define AGE     (short)1
#define ID      (short)2
#define LENGTH  8

struct namelist
{
        char    name[LENGTH];
        short   age;
        long    idcode;
};

struct namelist section1[] =
{
        "Naoko",  17, 1234,
        "Midori", 22, 8888,
        "Rie",    19, 7777,
        "Eri",    20, 9999,
        "Kyoko",  26, 3333,
        "",        0,    0
};

int count;

void sort();
```

Now the main program.

```
main( )
{
        count = 0;
        for ( ; ; )
        {
              sort(section1, NAME);
              count++;
              sort(section1, AGE);
              count++;
              sort(section1, ID);
              count++;
        }
}
```

The remainder of the program defines the functions called from main:

```
void sort(list, key)
struct namelist list[];
short key;
{
        short i,j,k;
        long min;
        char *name;
        struct namelist worklist;

        switch(key){
              case NAME :
                    for (i = 0 ; *list[i].name != 0 ; i++){
                          name = list[i].name;
                          k = i;
                          for (j = i+1 ; *list[j].name != 0 ; j++){
                                if (strcmp(list[j].name , name) < 0){
                                      name = list[j].name;
                                      k = j;
                                }
                          }
                          worklist = list[i];
                          list[i] = list[k];
                          list[k] = worklist;
                    }
                    break;
              case AGE  :
                    for (i = 0 ; list[i].age != 0 ; i++){
                          min = list[i].age;
                          k = i;
                          for (j = i+1 ; list[j].age != 0 ; j++){
                                if (list[j].age < min){
                                      min = list[j].age;
                                      k = j;
                                }
                          }
                          worklist = list[i];
                          list[i] = list[k];
                          list[k] = worklist;
                    }     break;
```

```
        case ID   :
        for (i = 0 ; list[i].idcode != 0 ; i++){
                min = list[i].idcode;
                k = i;
                for (j = i+1 ; list[j].idcode != 0 ; j++){
                    if (list[j].idcode < min){
                        min = list[j].idcode;
                        k = j;
                    }
                }
                worklist = list[i];
                list[i] = list[k];
                list[k] = worklist;
            }
            break;
    }
```

5.3 Running HEW (Pure Debugger) for ALE300L

- Execute HEW (Pure Debugger) for ALE300L by selecting High-Performance Embedded Workshop.

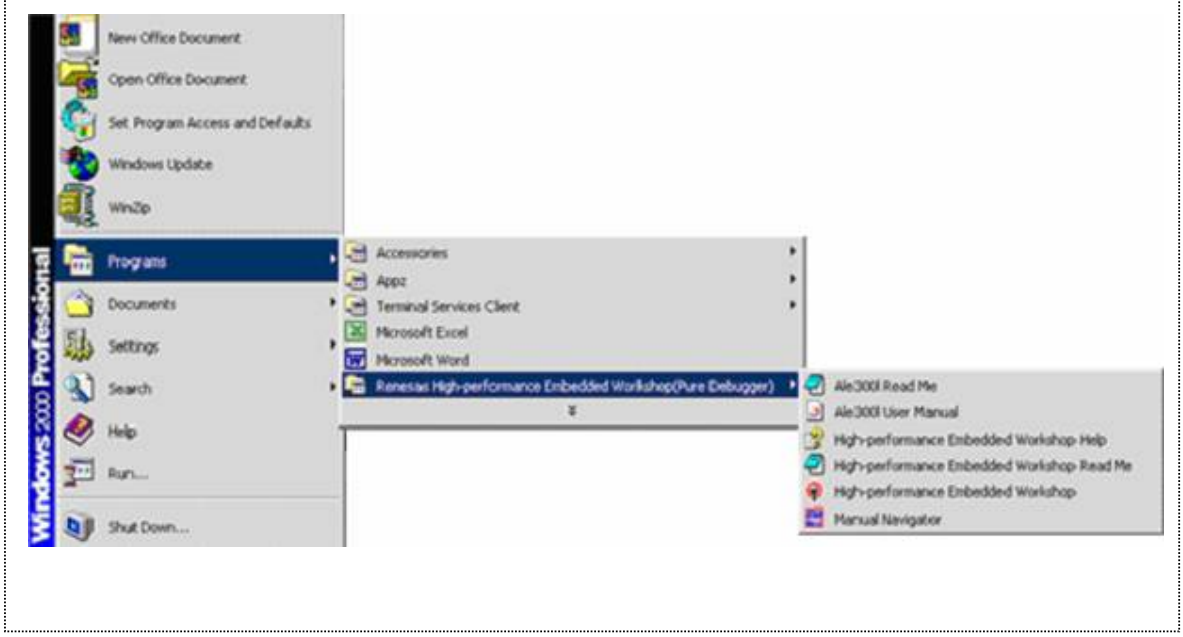


Figure 5-1 HEW (Pure Debugger) for ALE300L Icon

5.3.1 Selecting the Target

HEW (Pure Debugger) for ALE300L can be extended to support multiple target emulators or platforms (if the system is setup for more than one platform), user will have to choose a platform for the session from *Debug Settings…* in the *Options* menu.

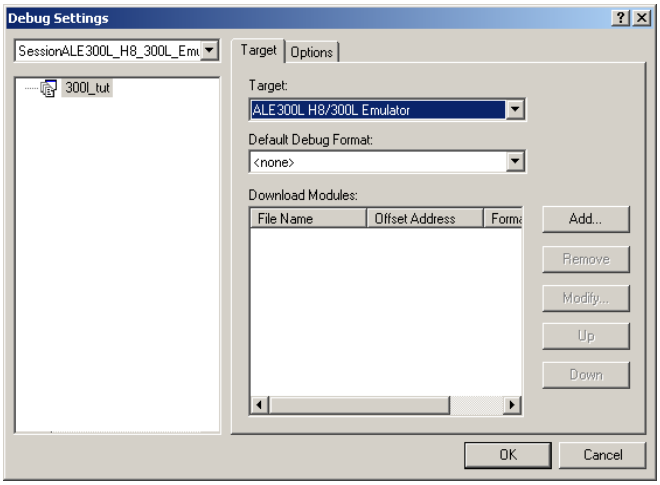


Figure 5-2 Select Session Dialogue Box

- For this 300L_3800_Tut, select '*ALE300L Emulator*' and then click OK to continue.

Note: User can change the target platform at any time by choosing *Debug Setting…* from the *Tools* menu. In there User can also define the Download Module/s for Debugging.

When the emulator has been successfully setup, the HEW (Pure Debugger) for ALE300L desktop window will be displayed. , with the message Ready in the status bar and message Connected Debug tab window . Figure 5.2 shows the key features of the window.

Note that you can change the target platform at any time by choosing *Debug Settings…* in the *Options* menu. If you have only one Target platform installed this menu option will not be available.

The key features of the window are:

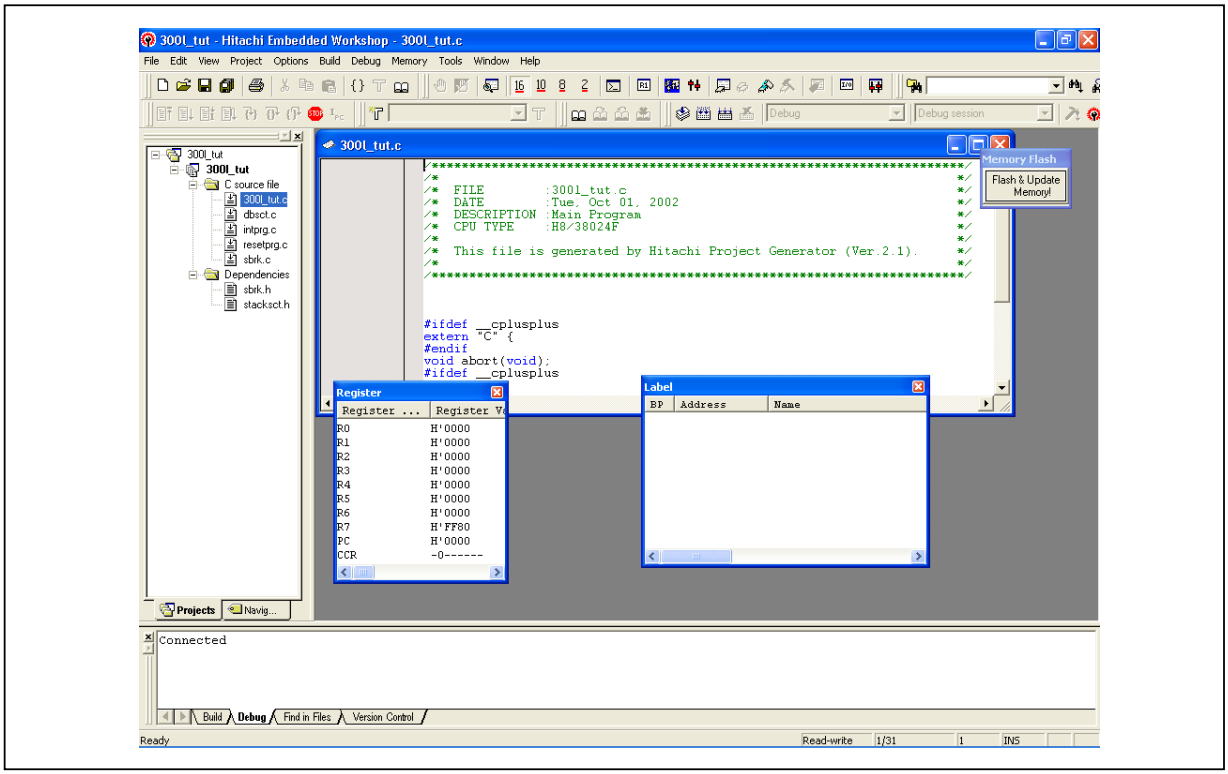


Figure 5-3 High-Performance Embedded Workshop Window

The key features of HEW (Pure Debugger) for ALE300L are described in the following sections:

Menus : Give you access to the HEW (Pure Debugger) for ALE300L debugging commands for controlling ALE300L.

Toolbar : Provides convenient buttons as shortcuts for the most frequently used menu commands.

Program Window : Displays the source of the program being debugged.

Workspace : Project's or Projects' files

Status Bar : Displays the status of the ALE300L. For example, progress information about downloads.

Help Button : Activates context sensitive help about any feature of the HEW (Pure Debugger) for ALE300L software.

5.4 Setting up the ALE300L HEW (Pure Debugger)

Before downloading a program to the ALE300L emulator you need to set up the user system for your application. The following items need to be configured:

- The device type
- The memory map

The following sections describe how to set up the ALE300L emulator as appropriate for the tutorial program.

5.4.1 Configuring the Platform

To set up the target configuration

- Choose Configure Platform… by clicking on the following icon on the Toolbar:

- The following Configure Platform dialog will appear:

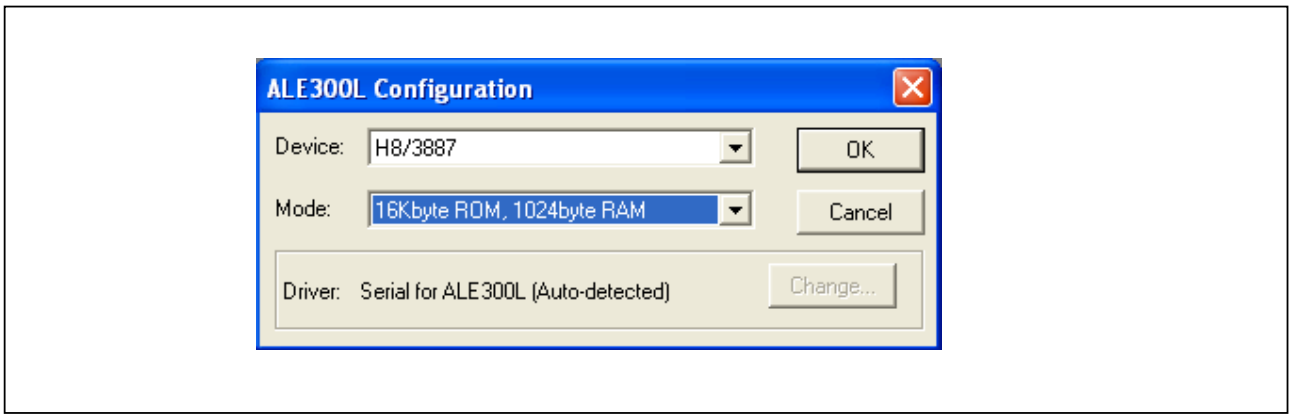


Figure 5-4 Target Configuration Dialogue Box

Ensure the setting is similar to above figure, denoted by *Figure 5-4*.

- There are various devices for selection that is supported by ALE300L. To Select another Device type or Memory mode, change the Device by clicking on the combo-box. This will automatically update the Mode combo-box, and vice-verse.
- Proceed to clicking OK to change the target configuration.

5.4.2 Mapping the Memory

The next step is to map the ALE300L emulator memory for the application you are developing.

- Choose Configure Map… from the Memory menu, or click the Open memory mapping button in the toolbar:

The **Memory Mapping** window shown in the following figure is displayed.

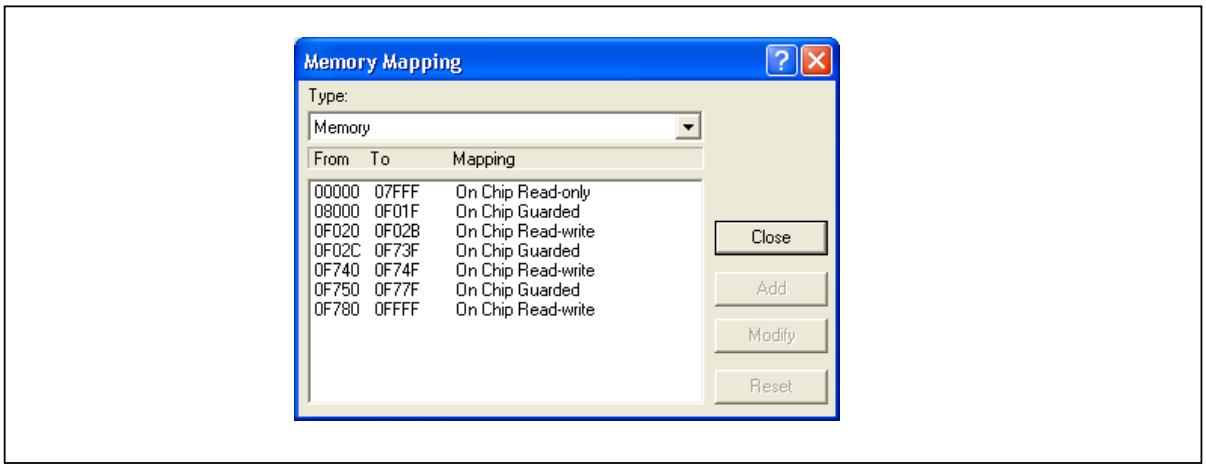


Figure 5-5 Target Memory Configuration Dialogue Box

For **TUTORIAL** we can use the default mapping, but you can view the mapping as follows:

- Click on Add or Modify button to view the edit memory mapping dialog. User can change or add desired address in this dialog.

The **Edit Memory Mapping** dialog box is displayed.

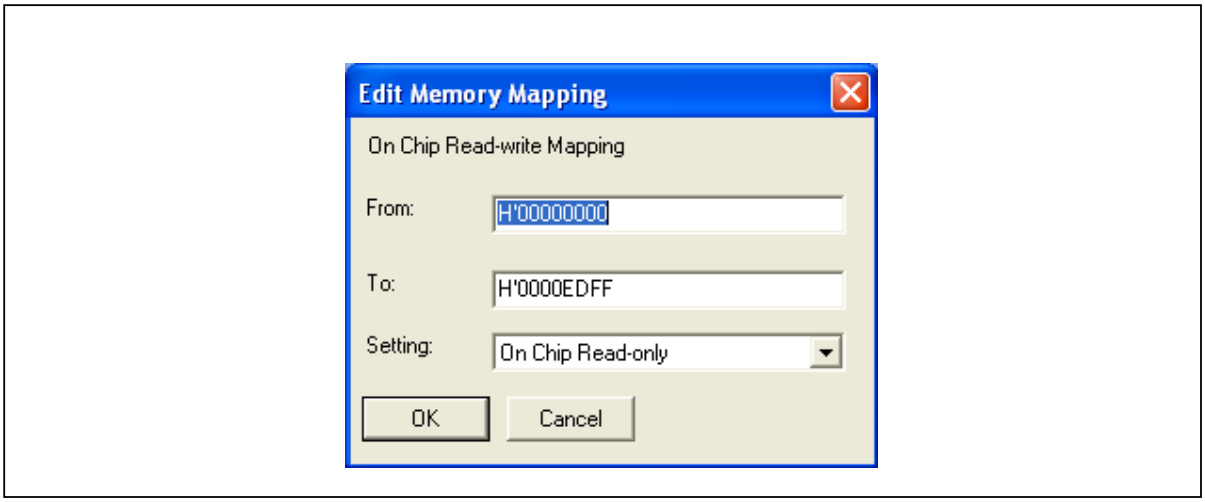


Figure 5-6 Edit Memory Mapping Dialog Box

The ALE300L emulator allows you to map any area of memory as described in the following table.

Table 5-2 Memory Types

Memory	Description
Reserved Area	MCU unused & reserved area
Internal IO	MCU on-chip Input/Output area
Internal RAM	MCU on-chip random access memory
Internal ROM	MCU on-chip read only memory

For each of these options you can specify one of the three access types listed in the following table.

Table 5-3 Access Types

Access Type	Description
On-Chip Read-write	RAM memory
On-Chip Read-only	ROM memory
On-Chip Guarded	No access allowed

- Click on the Close button to end the Memory Mapping configuration and open Status window under the View menu.
- Select the Memory tab in Status window to show the Memory Mapping configured:

Note: Memory maps differ depending on the MCU. For details, refer to the H8/300L series Hardware Manual.

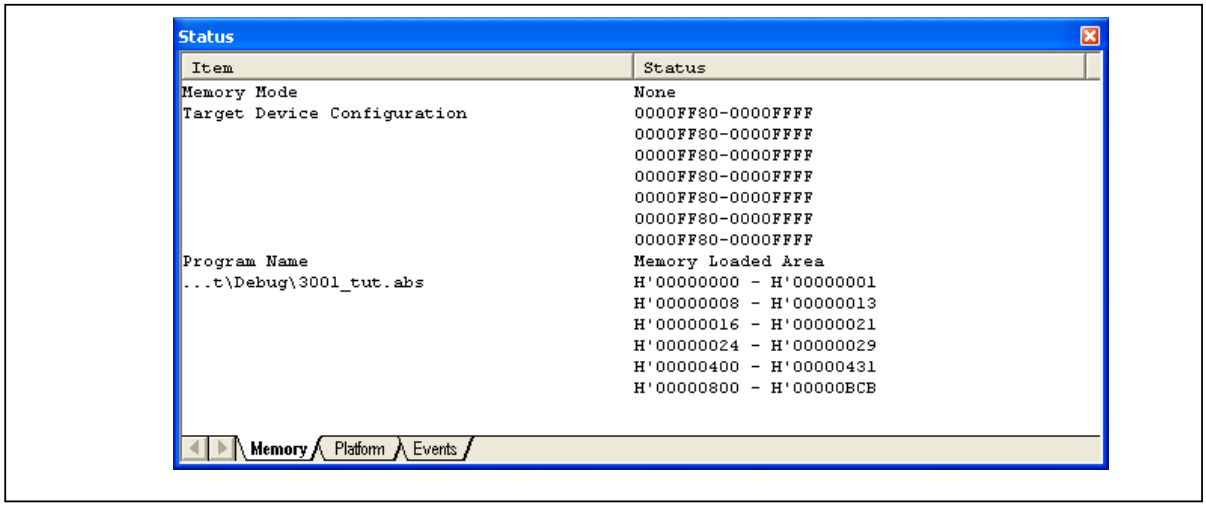


Figure 5-7 Target Memory Configuration Dialog

The following explains the target memory configuration dialog:

Target Device Configuration : Display the memory configuration of the specific target device selected.

Program Name : Downloaded Module (User Target Program)

Loaded Memory Areas : Display the memory space that the loaded program has occupied

5.5 Downloading the Tutorial Program

Once the ALE300L emulator is set up you can download the object program you want to debug.

5.5.1 Loading the Object File

- First load the ELF/DWARF2 file, as follows:
- Choose Options menu and Debug Setting…
- Select the file '*300L_3800_Tut.abs*' by clicking on Add… button.

Note: Select Elf/Dwarf2 for format (this is the default file-format generated after link phase)

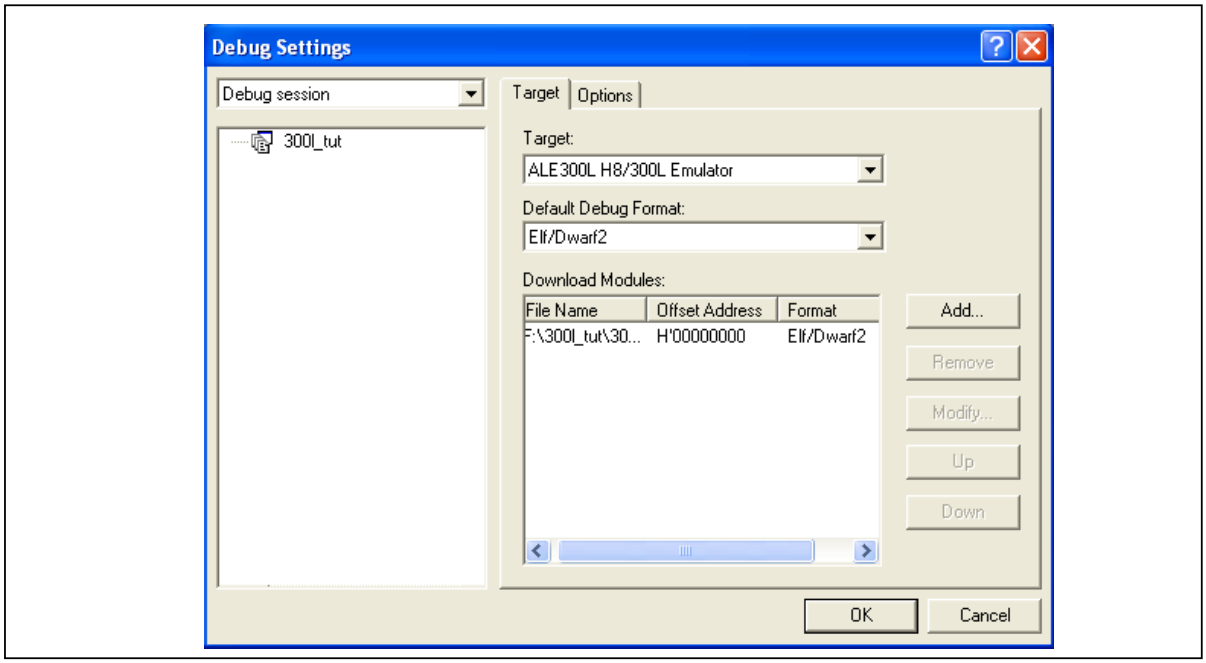


Figure 5-8 Load Object File Dialogue Box

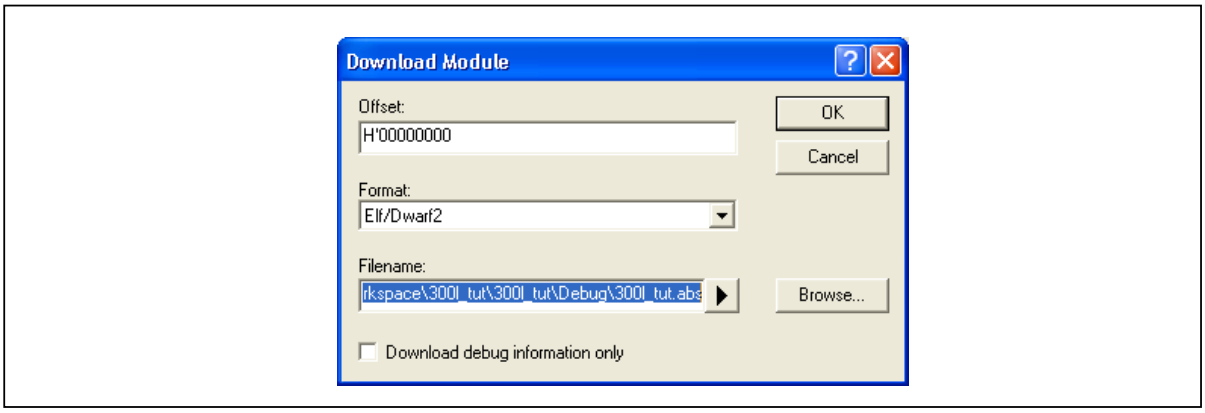


Figure 5-9 Configure Load Object File Dialog

When the file has been loaded, the Status-window Memory Tab will show the downloaded Memory Address (*Figure 5-7*)

Note: All the code should lie within the on-chip ROM.

5.5.2 Displaying the Program Listing

HEW allows you to debug a program at source level, so that you can see a listing of the program alongside the disassembled code as you debug. To do this you need to read in a copy of the source program from which the object file was compiled.

- Choose Reset CPU from the Run menu.

User will be prompted for the '*Resetprg.c*' source file corresponding to the loaded object file if HEW could not automatically locate the required file

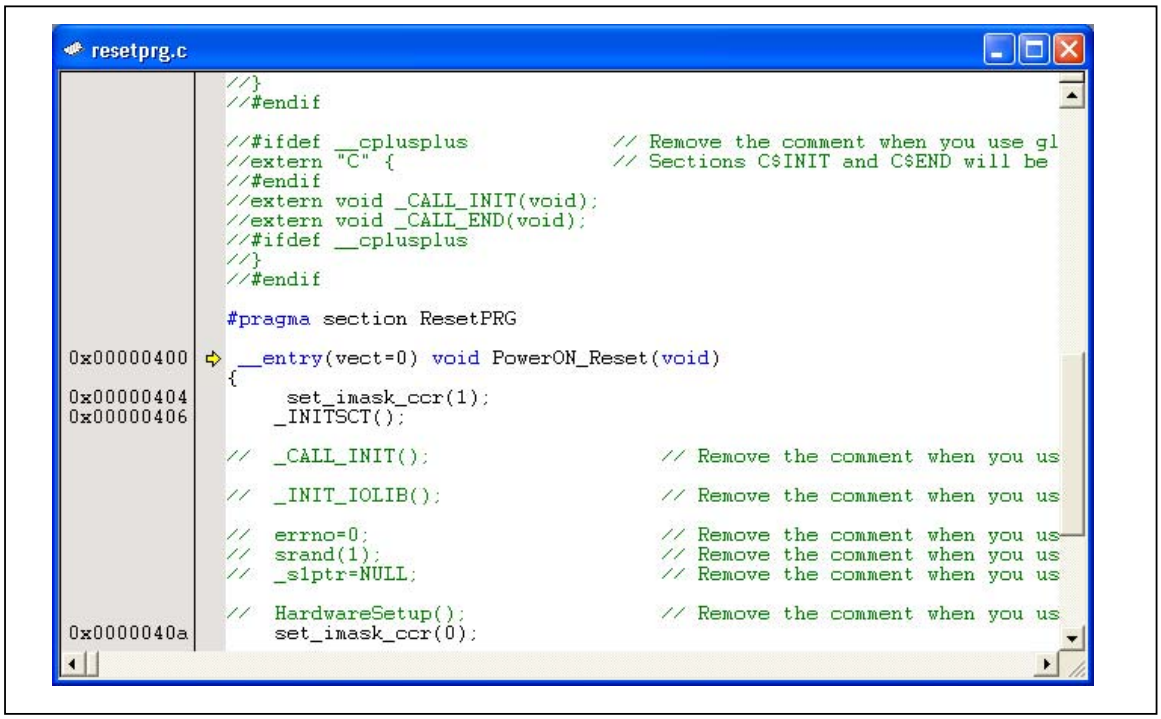


Figure 5-10 Source-window "Resetprg.c"

- Run the program until Address H'0000040c.

- Single step again to Jump into the 300L_3800_Tut .c main program window

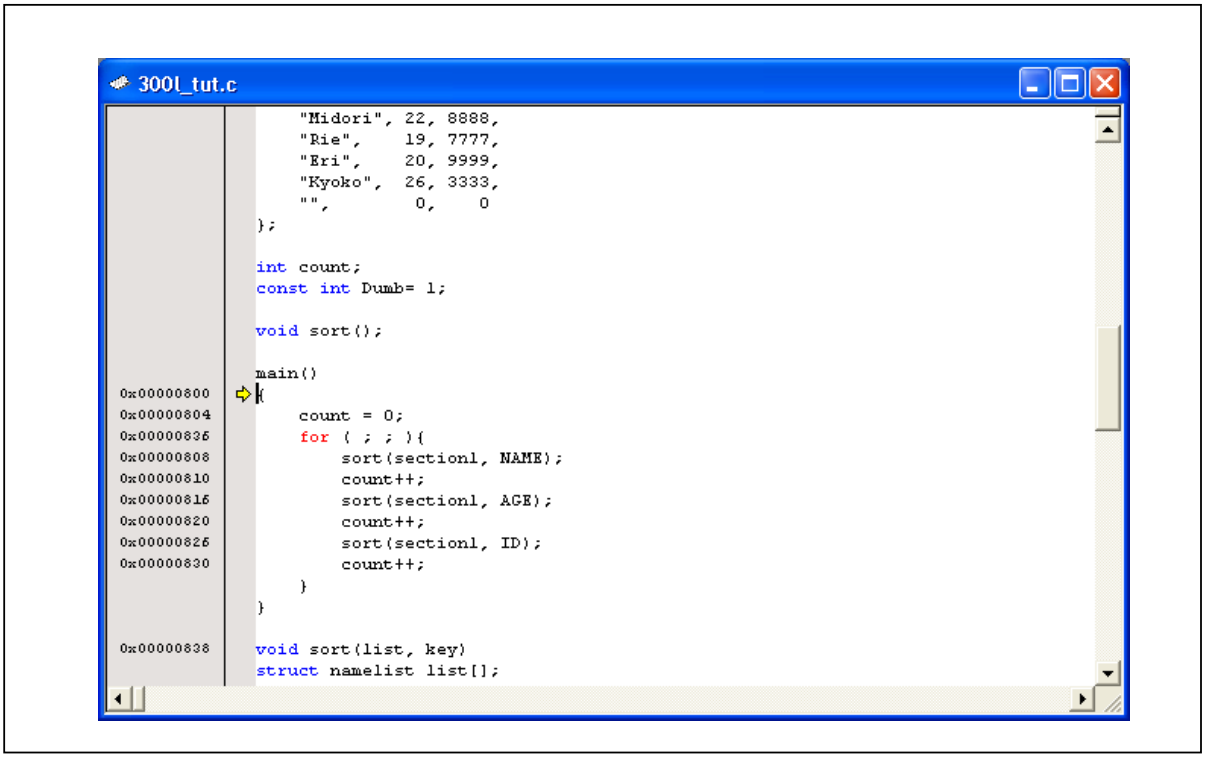


Figure 5-11 Source-window"300L_3800_Tut.c"

- If necessary, choose Tools: Format View… from the menu to select a font and size suitable for your computer.

The above source-window has it font change to Courier New, 8 point font.

Note: If change of font or size did not take place in the window, close the window and re-open the file again.

5.6 Using Breakpoints

The simplest debugging aid is the program breakpoint, which lets you halt execution when a particular point in the program is reached. You can then examine the state of the MCU and memory at that point in the program.

5.6.1 Setting a Program Breakpoint

The program window provides a very simple way of setting a program breakpoint.
For example, set a breakpoint at address H'00000808 as follows:

- Click once on the line containing address H'00000808 and right-click for the pop-up menu and select *Toggle Breakpoint*

OR

- Click once on the line containing address H'00000808 and press F9.

A red dot will be displayed there to indicate that a program breakpoint is set at that address.

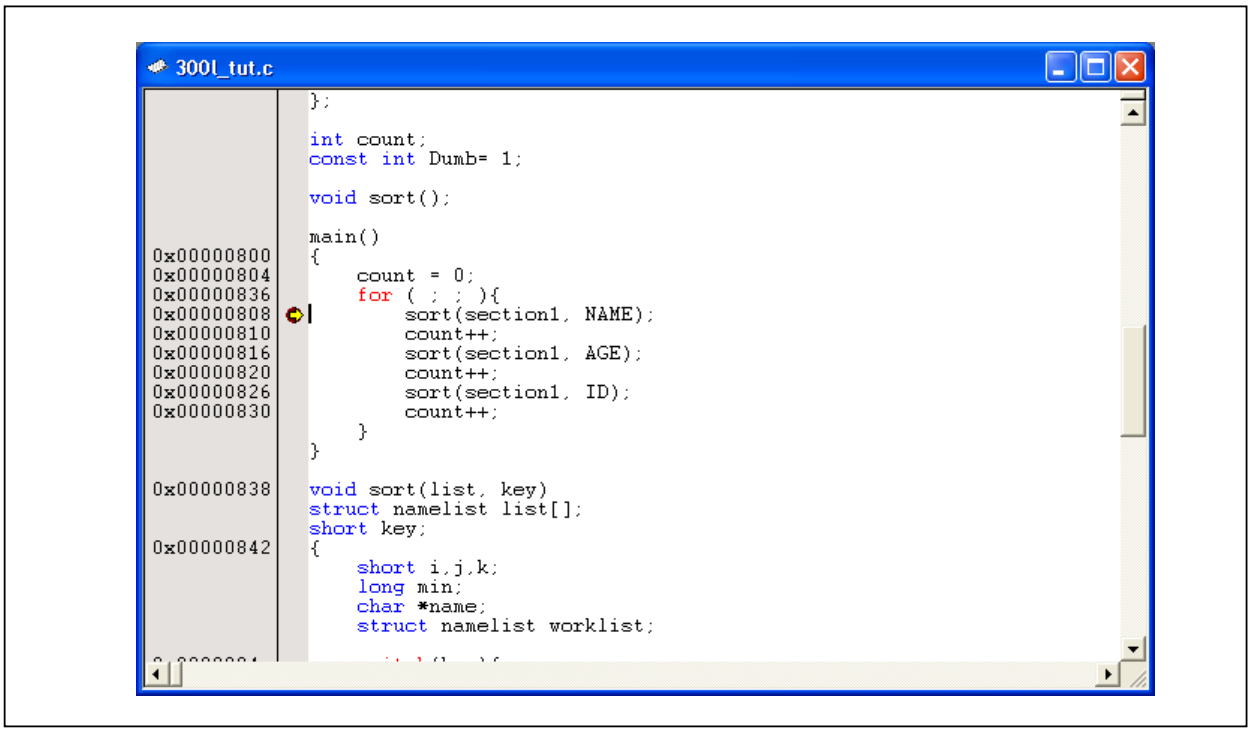


Figure 5-12 Setting a Breakpoint

5.6.2 Executing the Program

To run the program from reset:

- Choose **Go Reset** from the **Run** menu, or click Go Reset button at the toolbar icon.

The program will be executed up to the breakpoint you inserted, and the statement will be highlighted in the program window to show that the program has halted.

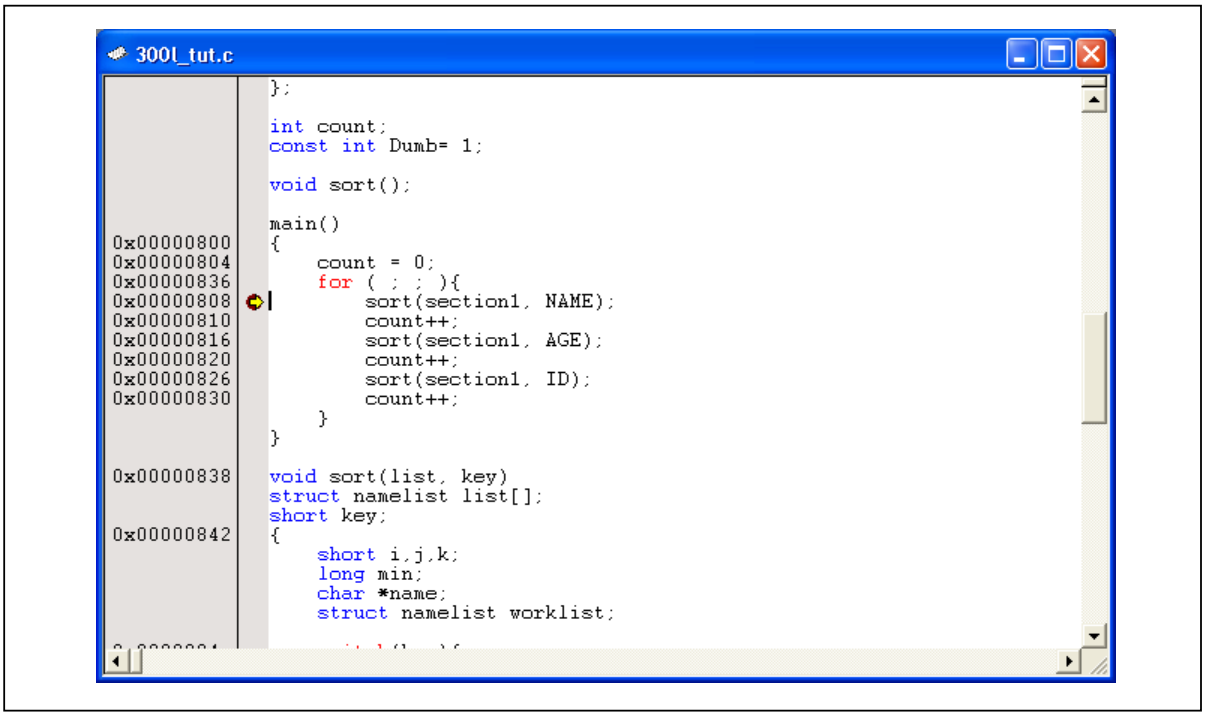


Figure 5-13 Program Break

This can be viewed under cause of the last break in the System Status window.

- Choose **Status Window** from the **View** menu, or click the Status window button in the toolbar:

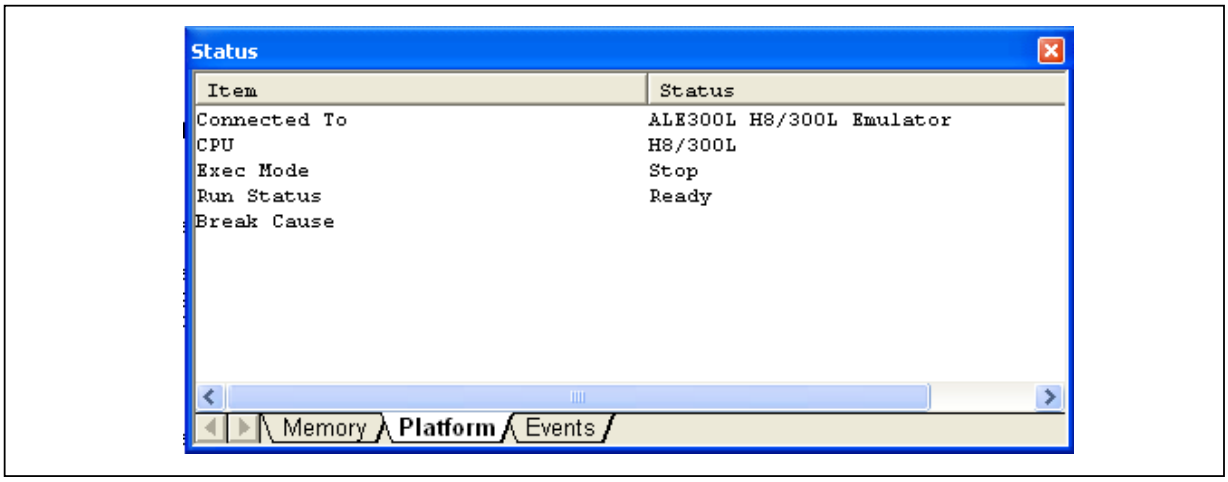


Figure 5-14 System Status Window

The **Cause of last break** line shows that the break was a program break.

5.6.3 Reviewing the Breakpoints

The list of all the breakpoints set in the program can be viewed in the Breakpoints window.

- Choose Breakpoints from the View menu, or click the Breakpoint Window button in the toolbar:

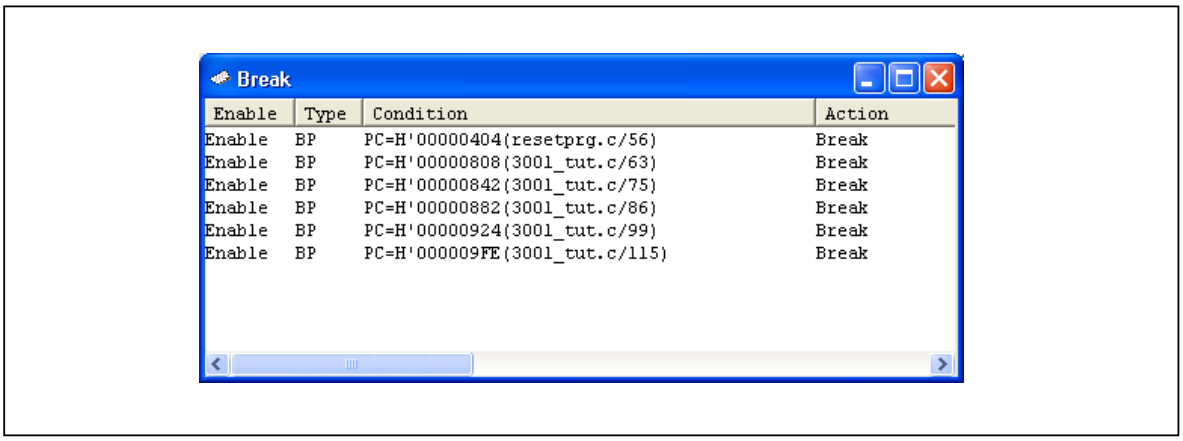


Figure 5-15 Breakpoint Window

The Breakpoints window also allows user to perform the following:

1. Define new breakpoints
2. Delete existing breakpoints
3. Disable existing breakpoints

- Right-mouse click within the Breakpoint-window to show the following pop-up:

Figure 5-16 Popup in Breakpoint Window

5.6.4 Examining Registers

While the program is halted, you can examine the contents of the ALE300L series MCU registers. These are displayed in the Registers Window..

- Choose Registers from the View menu, or click the Registers Window button in the toolbar:

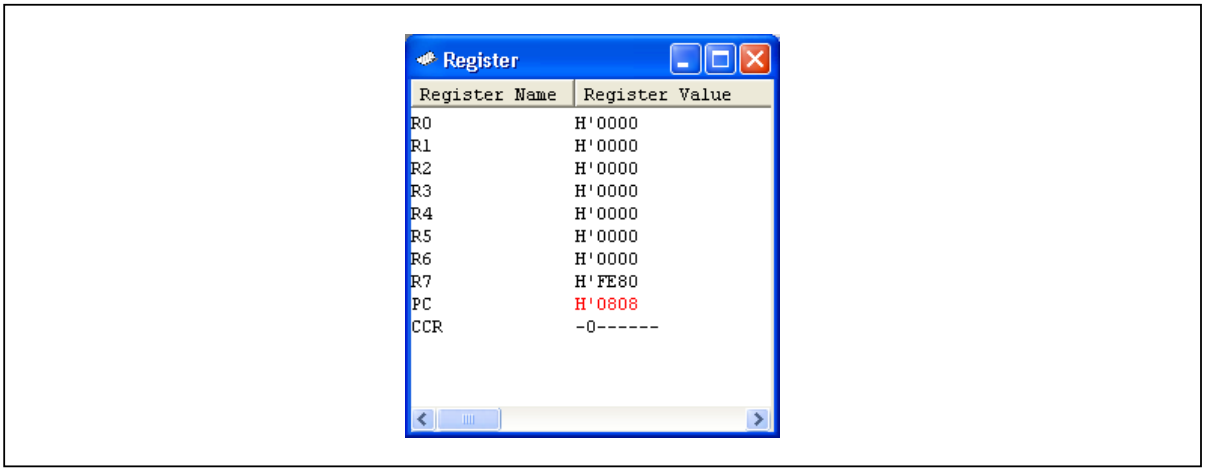


Figure 5-17 CPU Register Window

As expected, the value of the program counter (PC) is the same as the highlighted statement, H'00000808.
The registers' values can be changed from the Registers window by double-clicking on respective registers in the Registers window.

The Register-PC dialogue box allows you to edit the value.

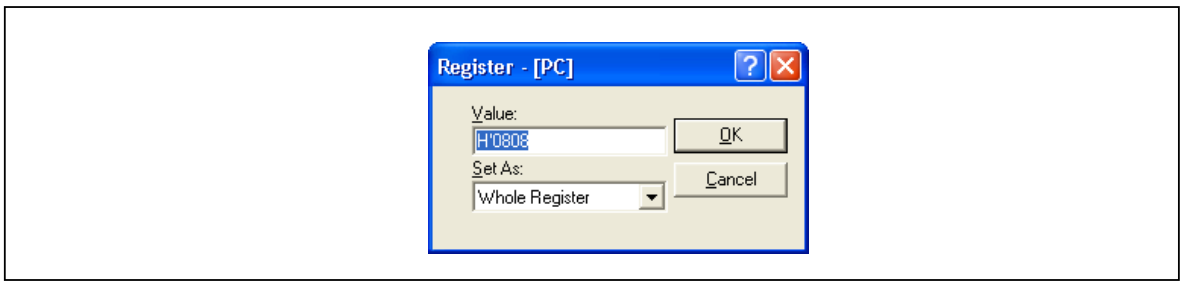


Figure 5-18 Changing Register Value

5.7 Stepping Through a Program

The ALE300L emulator provides a range of options for single stepping through a program, executing an instruction or statement at a time. The alternative step commands listed in table 5.4 are provided.

Table 5-4 Step Commands

Command	Description
Step In	Executes every statement, including statements within functions.
Step…	Allows you to step repeatedly at a specified rate.

5.7.1 Single Stepping

- Execute up to the breakpoint from the current position by choosing Go from the Run menu, or clicking the Go button in the toolbar.

- Issue one Step In from the Run menu, or click on the Step In button in the toolbar command to execute into the function sort(section1, NAME).

The first instruction in the function sort(section1, ID) will be highlighted.

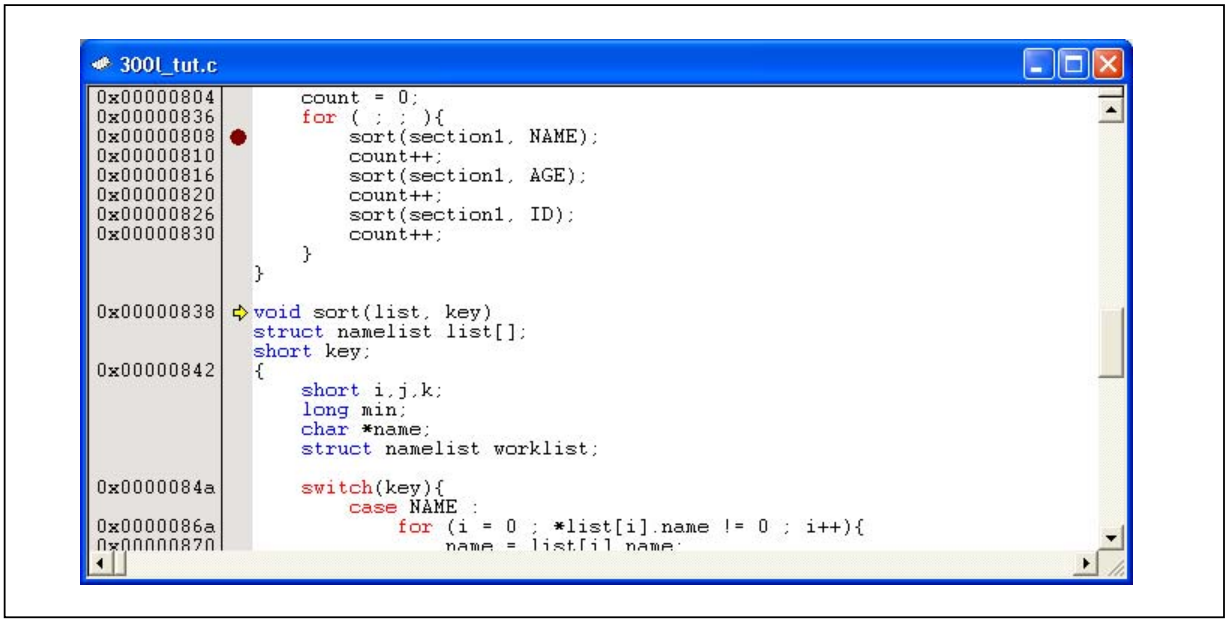


Figure 5-19 Executing Up to a Function Call

- Issue another Step In command to execute the next instruction.

Note: After performing several Step In, there will be a time when the Code window will be displayed showing the assembled codes. These codes are included into the user target program to handle certain tasks such as saving or restoring CPU registers etc. C Compiler generates these codes automatically.

- **Now choose** Step… **from the** Run **menu and set the selection as such:**

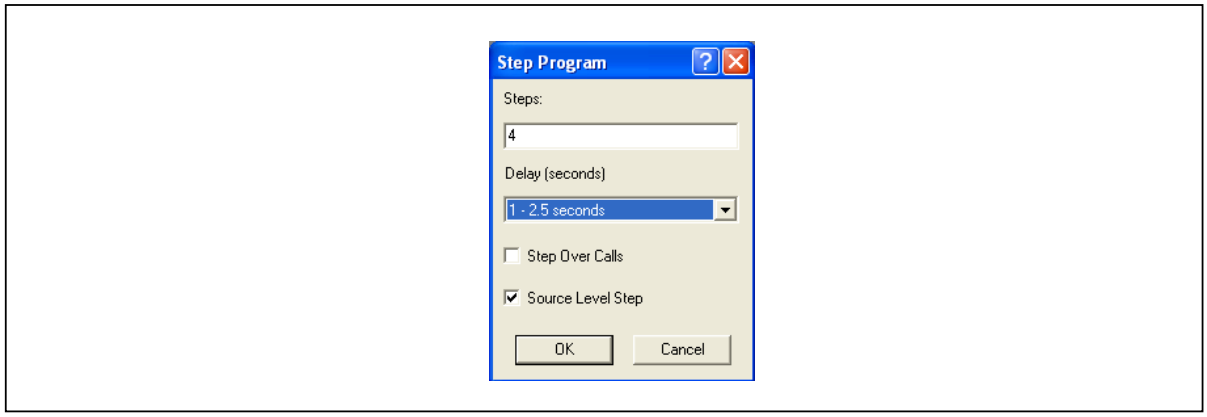


Figure 5-20 Step… window (Animated single-stepping window)

- This will activate multiple stepping (4 single-stepping, one after the other).
- The Highlight should now be at function **sort(section1, NAME);**
- Do TWO **Step In** under the **Run** menu, or click on the **Step In** button TWICE.

Disassembly window should be displayed at this point. It may be noticeable that certain assembly code are generated automatically by C Compiler and does not necessarily originate from the users.

Now step several steps to bring the Highlight back **300L_3800_Tut.c** Program window.

5.8 Examining Memory and Variables

You can monitor the behavior of a program by examining the contents of an area of memory, or by displaying the values of variables used in the program.

5.8.1 Viewing Memory

The contents of a block of memory can be viewed in the Memory Window.

For example, to view the memory corresponding to the array section1 in ASCII:

- Choose Memory… from the View menu, or click the Memory Window button in the toolbar.
- Enter "_section1" (a label valid only after downloading of Download Module- .abs file) in the Begin Address field and "ffff" in the End field, and keep the Format as Byte (x1).

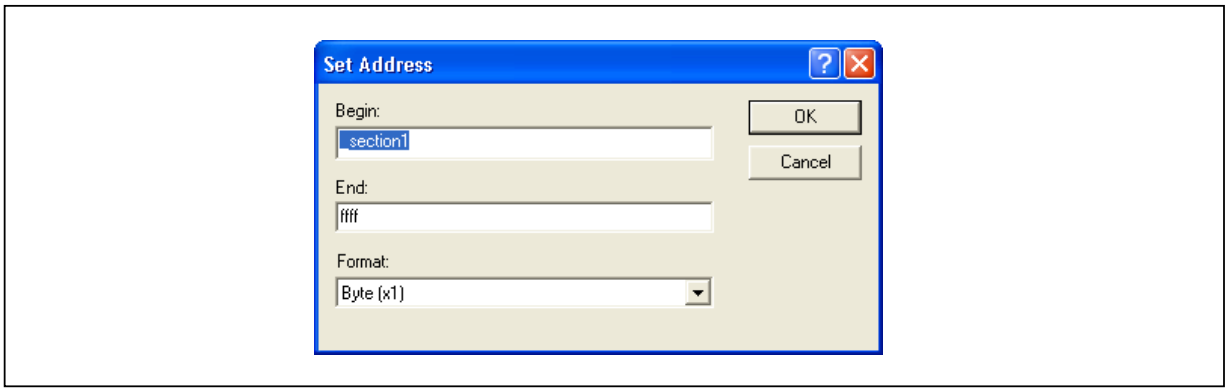


Figure 5-21 Open Memory Window

- Click OK to open the **Memory** window showing the specified area of memory.

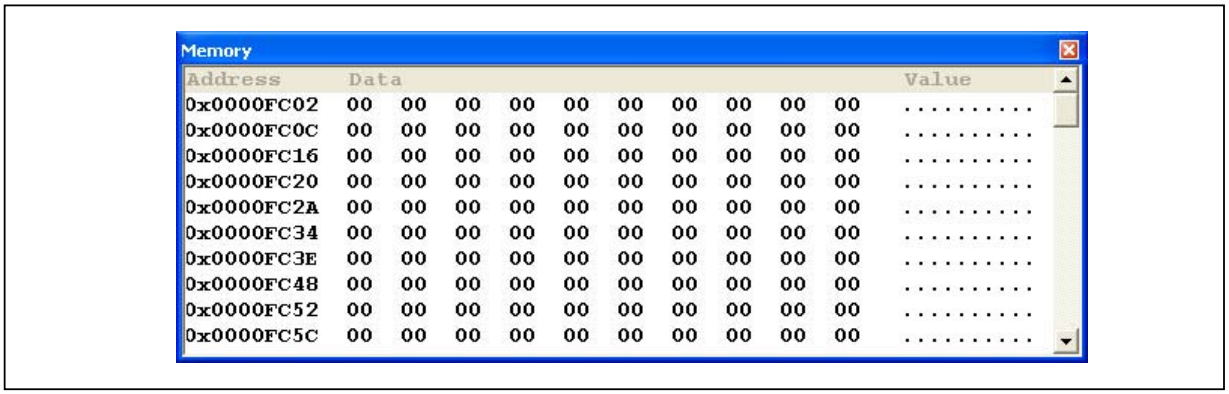


Figure 5-22 Memory Window

- Leave the **Memory** window open so that you can monitor the contents of the array **section1**.

You can monitor the behaviour of a program by examining the contents of an area of memory, or by displaying the values of variables used in the program.

5.8.2 Watching Variables

It is useful to be able to watch the values of variables as the program is being stepped.

For example, set a watch on the structure (STRUCT) variable section1, which is declared at the beginning of the program, using the following procedure:

- Scroll up in the program window until you see the line:

sort(section1, ID);

- In the Program windows, click once at the word section1 to position the blinking cursor on the word section1.
- Within the Program Window (300L_3800_Tut.c) perform a right mouse button click to display a pop-up menu, and choose Instant Watch.

The Instant Watch window will be displayed:

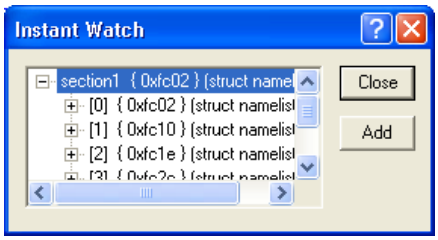


Figure 5-23 Instant Watch Window

- Click Add button to add the variable to the Watch Window

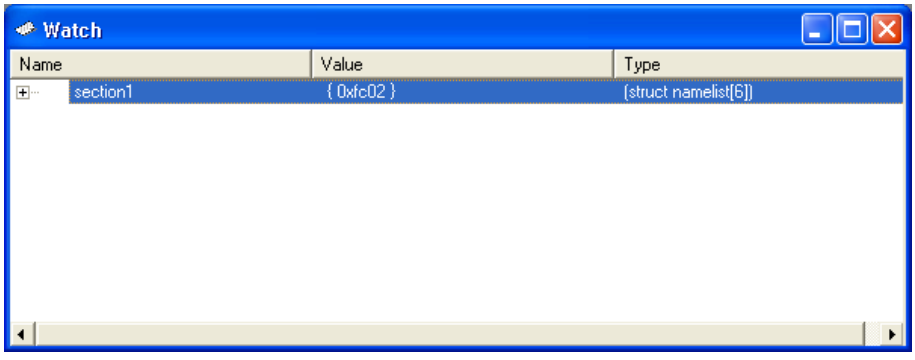


Figure 5-24 Watch Window

A variable watch can be added to the Watch Window by specifying its name. Use this method to add a Watch on the variable 'count' as follows:

- Click with the right mouse button within the Watch window and choose Add Watch… from the pop-up menu.

The Add Watch… dialogue box appears.

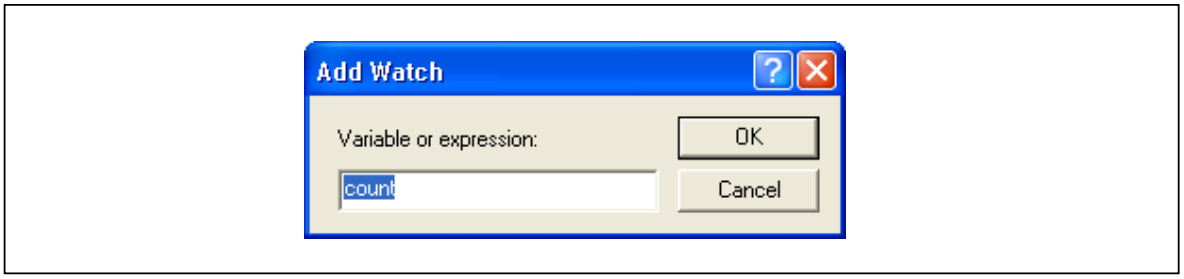


Figure 5-25 Add Watch Dialog Window

- Type the variable count and click OK.

The Watch Window will show the content of the variable label 'count'.

Note: You might be getting different result of count.

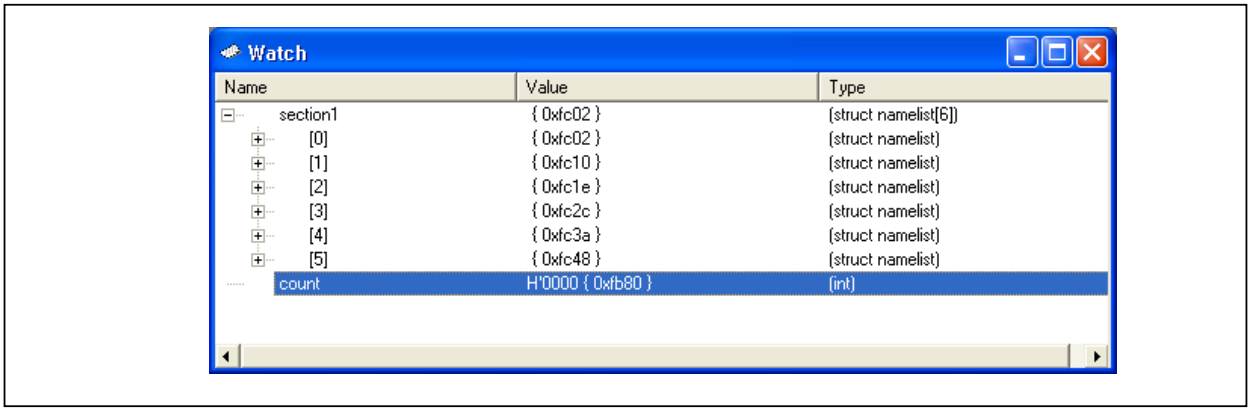


Figure 5-26 Watch Window

You can double-click the **+** symbol to the left of any symbol (i.e. **section1** in this case) in the **Watch window** to expand it and display the individual elements in the array.

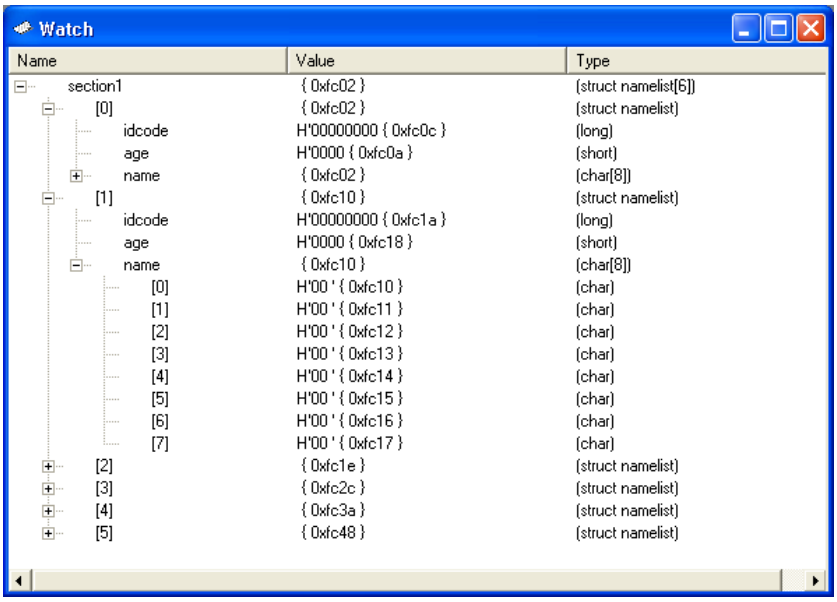


Figure 5-27 Displaying Individual Elements in an Array

5.9 Using the Combination Breakpoint

The combination breakpoint allows user to halt the program based on several conditions (such as address, data, access type & external probes).

5.9.1 Defining a Combination Breakpoint

Now define a combination breakpoint to monitor this port as follows:

- Choose **Breakpoint Window** from the **View** menu to display the **Breakpoints** window, or click the Breakpoint Window button in the toolbar.

- Click Add to define a new breakpoint.

The **Breakpoint/Event Properties** dialogue box allows you to define the breakpoint's properties.

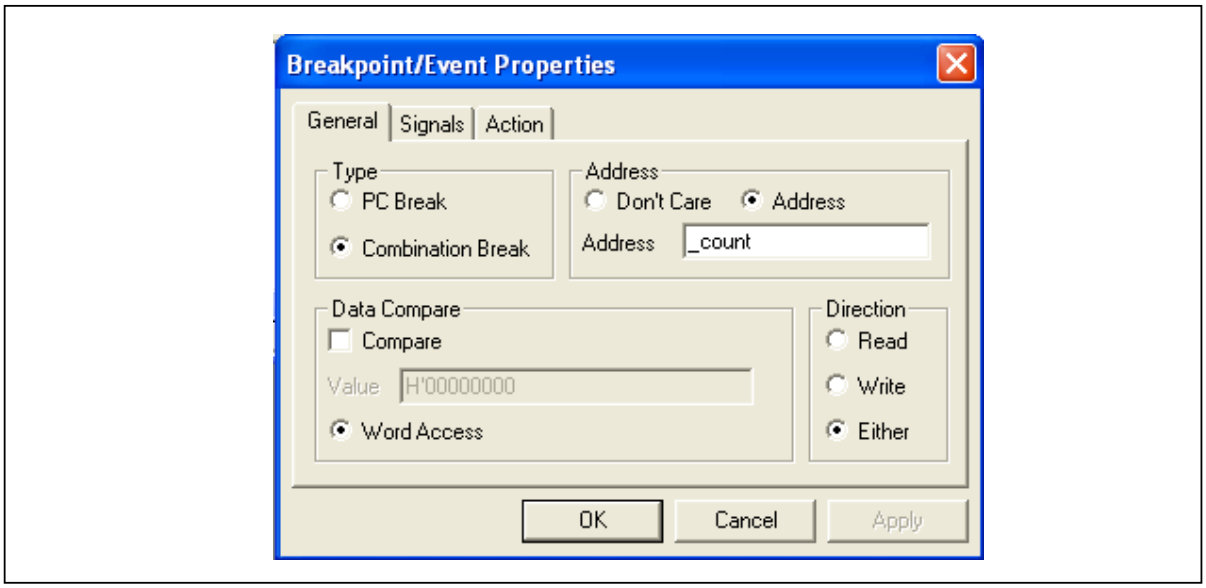


Figure 5-28 Breakpoint/Event Properties Dialogue Box

- Set the Type to Combination Break, to specify that you are defining a complex event, and enter the _count, into the Address Value: box.
- Click OK to define the breakpoint.

This will cause a break whenever the address _count (address of variable _count) is accessed, either for a read or a write.

The **Breakpoints** window shows the new complex breakpoint you have defined.

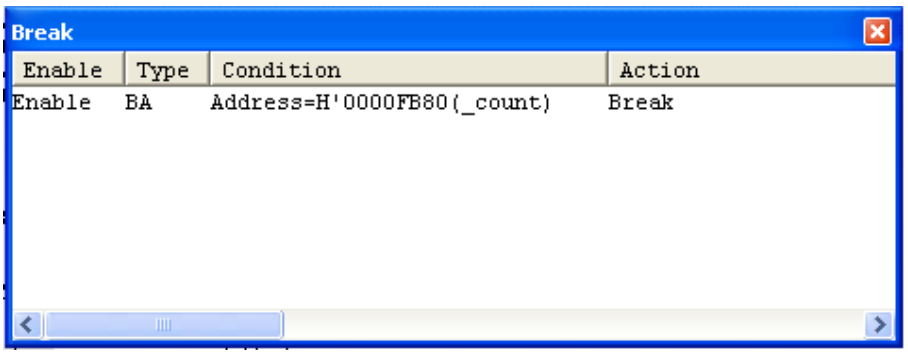


Figure 5-29 Breakpoints Window

- Run the program from the current position, by choosing **Go** from the **Run** menu, or click the Go button in the toolbar.

Execution will stop at address H'108C.

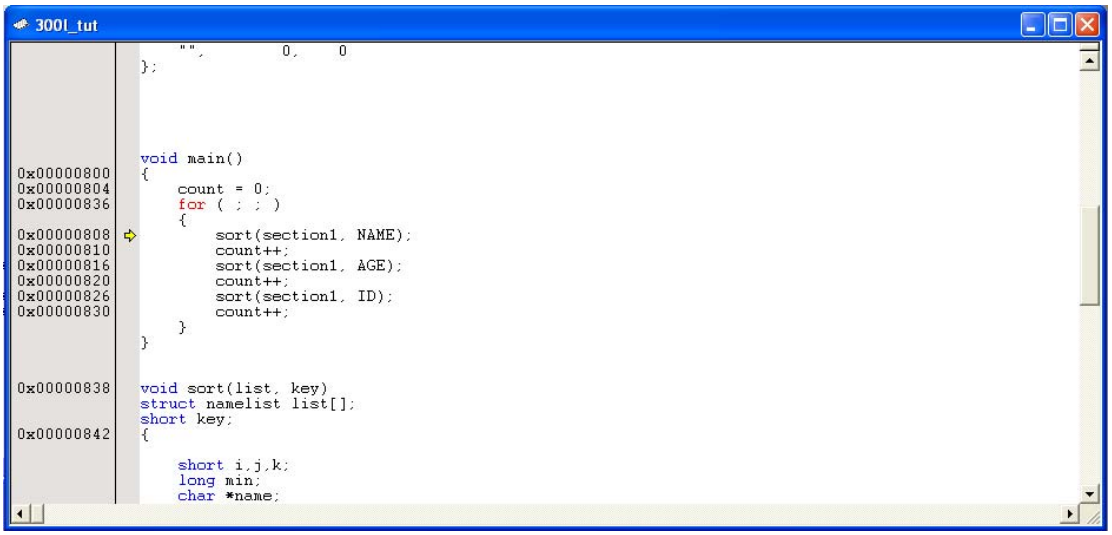


Figure 5-30 Combination Break

The system status window will display Cause of last break = Combination Break to indicate the type of break that has occurred.

5.10 Watching Local Variables

The localised variables within a function can be viewed using the Local Variables Window.

For example, in order to examine the local variables in the function sort, performs the following:

- Open the Local window by choosing Local… from the View menu.

Note: The Local Window will be empty if there is no local variable declared or local variables have not yet been entered. In another words, user target program execution should halt within a function with local variables to show any variables within Locals Window.

In this 300L_3800_Tut, once when the execution halts within the function sort(), the local variables within function sort() will be shown in Locals Window:

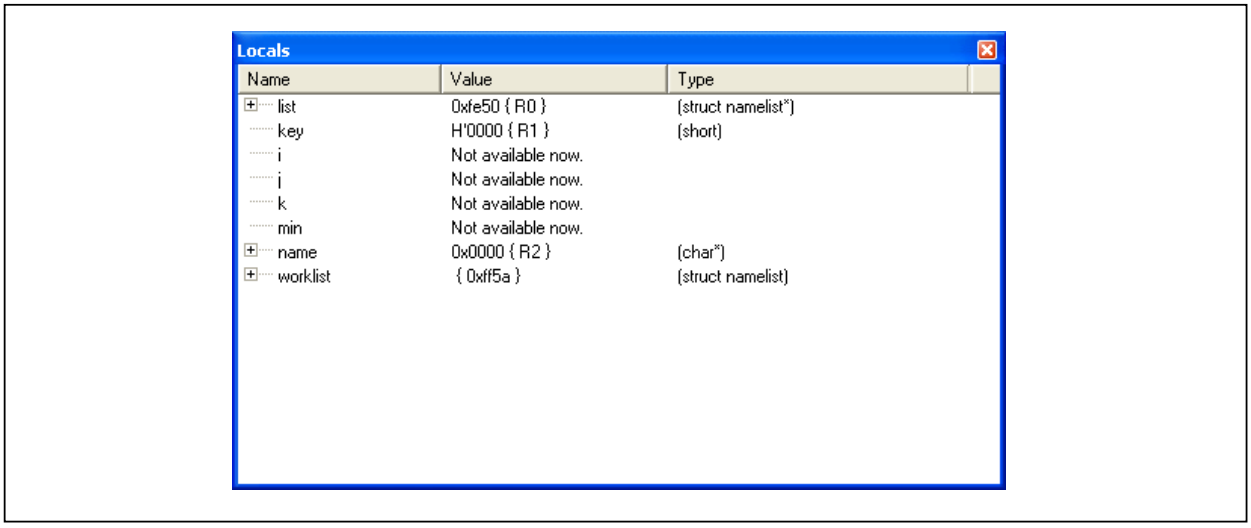


Figure 5-31 Locals Window

- Double-click the + symbol in front of the variable worklist in the **Locals** window to display the individual elements of the array worklist.

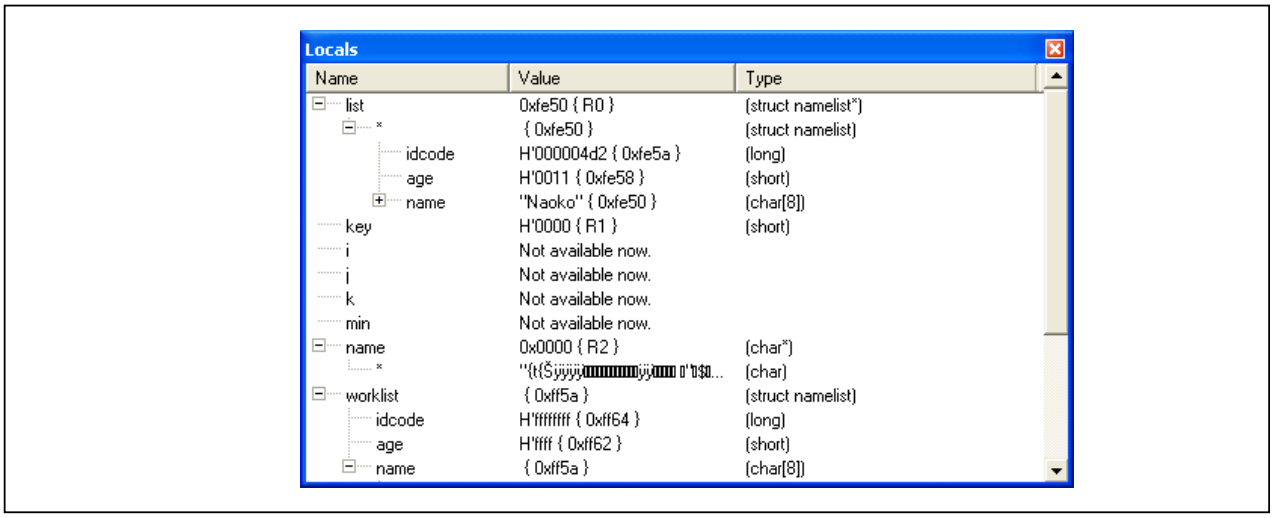


Figure 5-32 Displaying Individual Elements in an Array

5.11 Using the Trace Buffer

The trace buffer allows us to look back over previous MCU cycles to see exactly what the MCU was doing prior to a specified event, such as a complex breakpoint.

5.11.1 Displaying the Trace Buffer

Having identified the point in the program where the program accesses we can use the trace buffer to look back to see what accesses took place, and to verify that the correct data was transferred to the port.

- Open the **Trace** window by choosing **Trace Window** from the **View** menu, or click the Trace Window button in the toolbar.

If necessary scroll the window down so that you can see the last few cycles. The **Trace** window is displayed, as shown in the following figure.

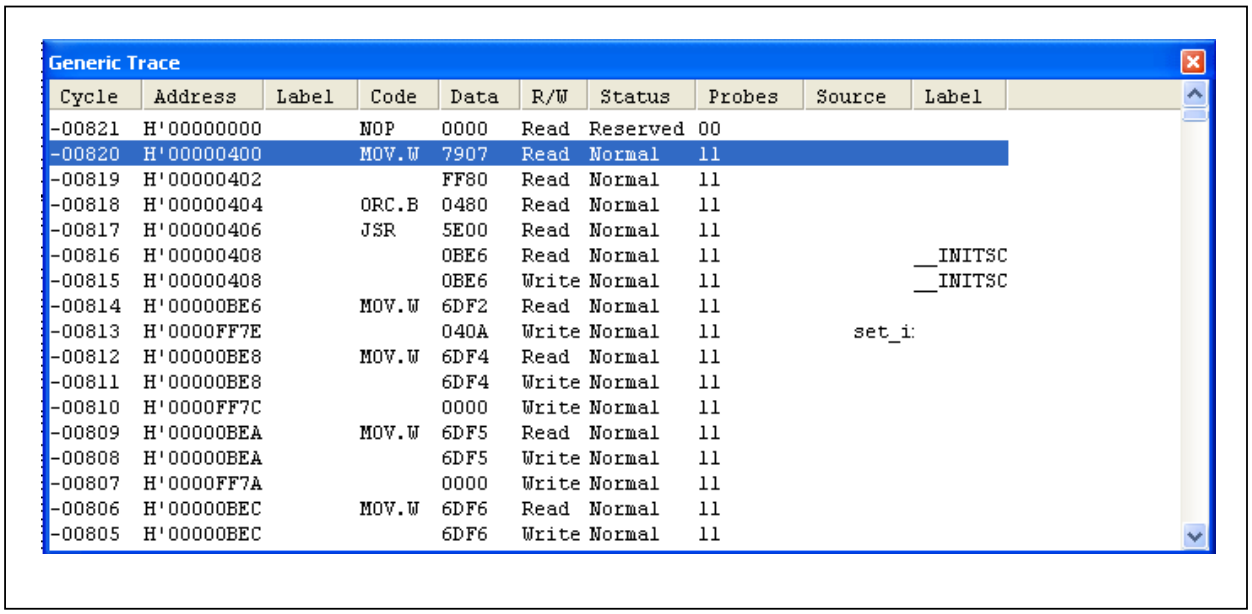


Figure 5-33 Trace Window

- If necessary adjust the width of each column by dragging the column dividers on either side of the labels just below the title bar.

5.12 Save the Session

Before exiting it is good practice to save your session, so that you can resume with the same ALE300L emulator and HEW configuration at your next debugging session.

- Choose Save Session from the File menu.
- Choose Exit from the File menu to exit from HEW (Pure Debugger) for ALE300L.

5.12.1 What Next?

This tutorial has introduced you to some of the key features of the ALE300L emulator, and their use in conjunction with the HEW (Pure Debugger). By combining the emulation tools provided in the ALE300L emulator you can perform extremely sophisticated debugging, allowing you to track down hardware and software problems very efficiently by precisely isolating and identifying the conditions under which they occur.

Section 6. Troubleshooting

Communication Problems	
1	Verify the communication channel
	Is the Communication Port used by another device? Right Communication Port (COM1/COM2) selected? Right Communication Baud Rate (9600~115200bps) selected on PC & ALE? Is UserVcc supplied?
2	Are you using 115200bps on a Window 3.x environment?
	Window 3.x Communication Driver does not support high Baud Rate link. However there are 3rd party communication driver that can support this communication. These shareware can be downloaded from the internet. But users are reminded to check the availability of the 16550 UART on their PC, if otherwise high speed communication is not possible too. Another Suggestion is to change the DIP SW setting (located on the main board of ALE300L) to maintain a communication link at a lower Baud Rate.
3	Power supply not switched on, or not connected, or connected loosely to the ALE300L emulator. Check the power LED on the ALE300L emulator.
4	The PC interface cable is not correctly connected between the PC interface board and the ALE300L emulator.
5	Wrong PC interface/serial cable used? PC Interface /serial cable break down? Verify that pins 2 and 3 of each end of cable are connected to each other respectively.
6	Is the target system drawing too much current? Reconnect the user supply as stated in section 2.4.2.B2.
No Power	
1	The fuse may have been blown due to mishandling such as shorting of UserVCC and VCC, or drawing of too much current from target system, … Simply replace the 1A fuse located beside the main power switch. **Note:** Please investigates the cause before replacing the fuse.
Target system not working	
1	Check User voltage
	• Running at Emulator supply (5 Volt) Target may be drawing too much current from the emulator • Running at User supply Is the jumper, CON3 removed?
2	Check User clock
	• Using emulator internal clock • Using target clock • Using external clock
3	Driving current
	ALE300L do not support high current drive unless Japan (E6000) user cables are used.

Section 7. Diagnostic

Although ALE300L is designed to have all possible protective measures, it is still pronged to damage by user system. Within HEW, users can activate the diagnostic tests by choosing **diagnostic window** from the **View** menu. Users can select the tests to verify the functionality of ALE300L.

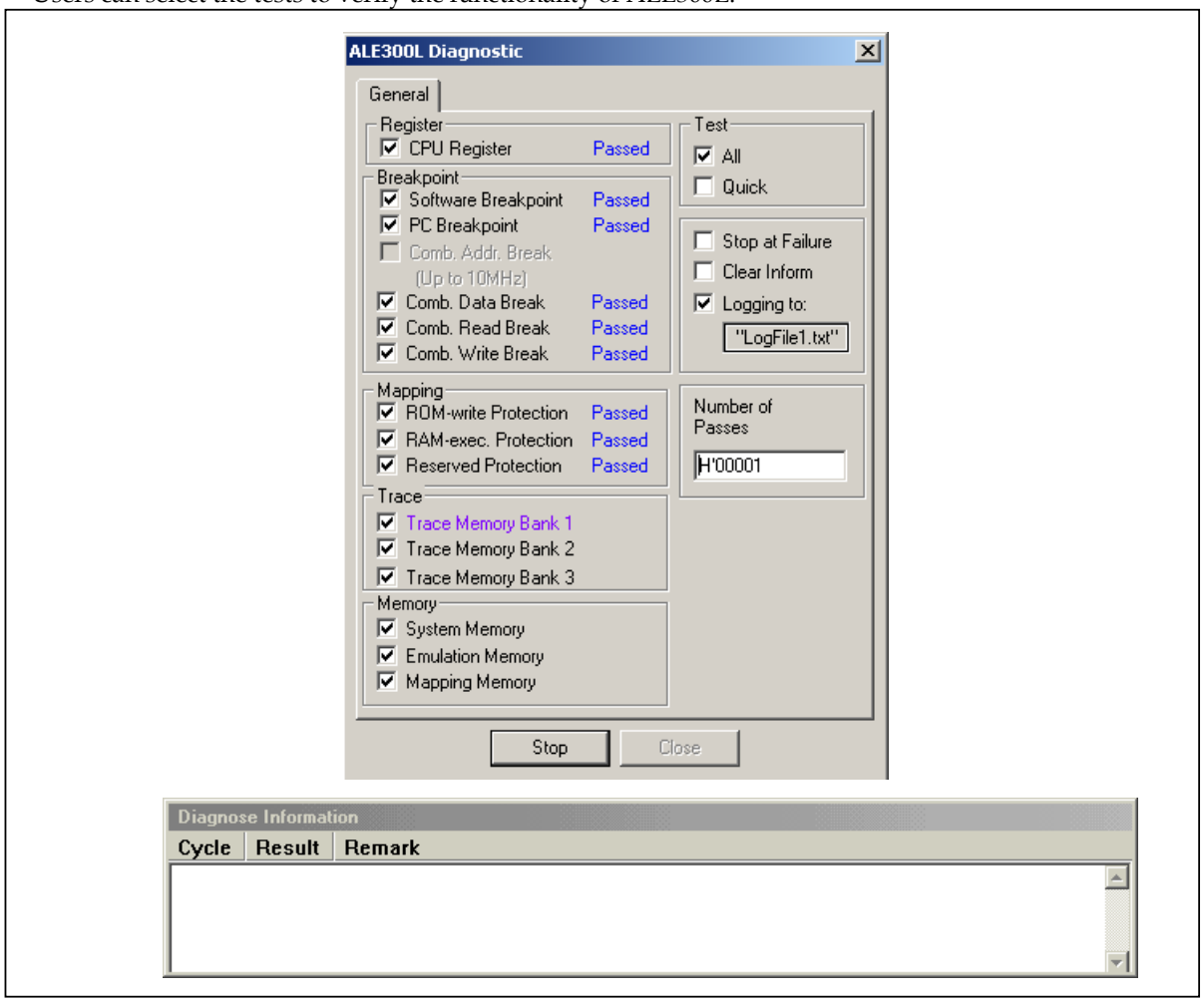


Figure 7-1 Diagnostic window

Self-Test performed has covered the majority features & functions of ALE300L.

1. System Memory
2. Emulation Memory
3. Mapping Memory
4. Trace Memory Bank 1 - 3
5. CPU Registers
6. Software Break
7. PC Break
8. Combination Break Address
9. Combination Break Data
10. Combination Break Read Access
11. Combination Break Write Access
12. Write ROM Break
13. Execute From Internal Area Break
14. Access Reserved Area Break

Appendix A: ALE300L User Connector Pin Configuration

ALE-60P (USERCON 1) – Signal name					
Pin No.	**SD97ALE3834**	**SD97ALE3927**	**SD97ALE3876 SD98ALE3877R**	**SD97ALE3644**	**SD97ALE3947**
1	P40/SCK3	AVcc	PE0/SEG49/M	Avcc	P40/SCK3
2	P41/RXD	`	PE1/SEG50/DO	PA3	P41/RXD
3	P42/TXD	`	PE2/SEG51/CL2	PA2	P42/TXD
4	GND	GND	GND	GND	GND
5	P43//IRQ0	`	PE3/SEG52/CL1	PA1	P43//IRQ0
6	AVcc	`	AVcc	PA0	Avcc
7	GND	GND	GND	GND	GND
8	PB0/AN0	PC3/DA3	TONEM	PC3	PB0/AN0
9	PB1/AN1	PC2/DA2	TONED	PC2	PB1/AN1
10	GND	GND	GND	GND	GND
11	PB2/AN2	PC1/DA1	VTref	PC1	PB2/AN2
12	PB3/AN3	PC0/DA0	AVref	PC0	PB3/AN3
13	GND	GND	GND	GND	GND
14	PB4/AN4	PB7/AN7	PB7/AN7	PB7[A]/AN7	PB4/AN4
15	PB5/AN5	PB6/AN6	PB6/AN6	PB6[A]/AN6	PB5/AN5
16	GND	GND	GND	GND	GND
17	PB6/AN6	PB5/AN5	PB5/AN5	PB5[A]/AN5	PB6/AN6
18	PB7/AN7	PB4/AN4	PB4/AN4	PB4[A]/AN4	PB7/AN7
19	GND	GND	GND	GND	GND
20	PC0/AN8	PB3/AN3	PB3/AN3	PB3[A]/AN3	PC0/AN8
21	PC1/AN9	PB2/AN2	PB2/AN2	PB2[A]/AN2	PC1/AN9
22	GND	GND	GND	GND	GND
23	PC2/AN10	PB1/AN1	PB1/AN1	PB1[A]/AN1	PC2/AN10
24	PC3/AN11	PB0/AN0	PB0/AN0	PB0[A]/AN0	PC3/AN11
25	AVss	AVss	AVss	Avss	Avss
26	`	`	`	`	`
27	Vss	Vss	Vss	GND	Vss
28	OSC1	OSC1	OSC1	OSC1	OSC1
29	GND	GND	GND	GND	GND
30	/RES	/RES	/RES	/RES	/RES
31	MD0	/NMI	/NMI	/NMI	/NMI
32	GND	GND	GND	GND	GND
33	P20//IRQ4//ADTRG	P40	P20//IRQ4//ADTRG	P40	P20//IRQ4//ADTRG
34	P21/UD	P41	P21/SCK1	P41	P21/UD
35	P22	P42	P22/SI1	P42	P22
36	P23	P43	P23/SO1	P43	P23
37	P24	P44	P24/SCK3	P44	P24
38	P25	P45	P25/RXD	P45	P25
39	P26	P46	P26/TXD	P46	P26/TMIB2A
40	P27	P47	P27//IRQ0	P47	P27/TMIB2B
41	GND	GND	GND	GND	GND
42	`	/IRQ0	`	/IRQ0	PA6
43	P50//WKP0//SEG1	P50//INT0	P50/WKP0/SEG1	P50//INT0	P50//WKP0
44	P51//WKP1//SEG2	P51//INT1	P51/WKP1/SEG2	P51//INT1	P51//WKP1
45	P52//WKP2//SEG3	P52//INT2	P52/WKP2/SEG3	P52//INT2	P52//WKP2
46	P53//WKP3//SEG4	P53//INT3	P53/WKP3/SEG4	P53//INT3	P53//WKP3
47	P54//WKP4//SEG5	P54//INT4	P54/WKP4/SEG5	P54//INT4	P54//WKP4
48	P55//WKP5//SEG6	P55//ADTRG//INT5	P55/WKP5/SEG6	P55//INT5//ADTRG	P55//WKP5
49	P56//WKP6//SEG7	P56/TMIB//INT6	P56/WKP6/SEG7	P56//INT6/TMIB	P56//WKP6
50	P57//WKP7//SEG8	P57/TMIY//INT7	P57/WKP7/SEG8	P57//INT7	P57//WKP7
51	GND	GND	GND	GND	GND
52	P60//SEG9	P60/RP0	P60/SEG9	P60[LC]	P60
53	P61//SEG10	P61/RP1	P61/SEG10	P61[LC]	P61
54	P62//SEG11	P62/RP2	P62/SEG11	P62[LC]	P62
55	P63//SEG12	P63/RP3	P63/SEG12	P63[LC]	P63
56	P64//SEG13	P64/RP4	P64/SEG13	P64[LC]	P64
57	P65//SEG14	P65/RP5	P65/SEG14	P65[LC]	P65
58	P66//SEG15	P66/RP6	P66/SEG15	P66[LC]	P66
59	P67//SEG16	P67/RP7	P67/SEG16	P67[LC]	P67
60	GND	GND	GND	GND	GND

Note: SD97ALE3834 H8/3834(U)(S), H8/3814 (U)(S)
SD97ALE3927 H8/3927
SD97ALE3947 H8/3947
SD97ALE3644 H8/3644 (R), H8/3657
SD97ALE 3876 H8/3877(U)(N)
SD97ALE 3877R H8/3877(U)(N), H8/3627, H8/3637

ALE-60P (USERCON 2)					
Pin No.	**SD97ALE3834**	**SD97ALE3927**	**SD97ALE3876 SD98ALE3877R**	**SD97ALE3644**	**SD97ALE3947**
1	P70/SEG17	P70	P70/SEG17	P70	P70
2	P71/SEG18	P71	P71/SEG18	P71	P71
3	P72/SEG19	P72	P72/SEG19	P72	P72
4	P73/SEG20	P73	P73/SEG20	P73	P73
5	P74/SEG21	P74/TMRIV	P74/SEG21	P74/TMRIV	P74
6	P75/SEG22	P75/TMCIV	P75/SEG22	P75/TMCIV	P75
7	P76/SEG23	P76/TMOV	P76/SEG23	P76/TMOV	P76
8	P77/SEG24	P77	P77/SEG24	P77	P77
9	GND	GND	GND	GND	GND
10	P80/SEG25	P80/FTCI	P80/SEG25	P80/FTCI	P80
11	P81/SEG26	P81/FTOA	P81/SEG26	P81/FTOA	P81
12	P82/SEG27	P82/FTOB	P82/SEG27	P82/FTOB	P82
13	P83/SEG28	P83/FTIA	P83/SEG28	P83/FTIA	P83
14	P84/SEG29	P84/FTIB	P84/SEG29	P84/FTIB	P84
15	P85/SEG30	P85/FTIC	P85/SEG30	P85/FTIC	P85
16	P86/SEG31	P86/FTID	P86/SEG31	P86/FTID	P86
17	P87/SEG32	P87	P87/SEG32	P87	P87
18	GND	GND	GND	GND	GND
19	P90/SEG33	`	P90/SEG33	P37	`
20	P91/SEG34	`	P91/SEG34	P36	P91/SCL1
21	P92/SEG35	`	P92/SEG35	P35	P92/SDA1
22	P93/SEG36	`	P93/SEG36	P34	P93/SCL2
23	P94/SEG37/M	`	P94/SEG37	P33	P94/SDA2
24	P95/SEG38/DO	`	P95/SEG38	P32/SO1	P95/TMRIH
25	P96/SEG39/CL2	`	P96/SEG39	P31/SI1	P96/TMCIH
26	P97/SEG40/CL1	`	P97/SEG40	P30/SCK1	P97/TMOH
27	GND	GND	GND	GND	GND
28	VCC	VCC	VCC	VCC	VCC
29	P10/TMOW	P10/TMOW	PD0/SEG41	P10/TMOW	P10/TMOW
30	P11/TMOFL	P11/TMOE	PD1/SEG42	P11	P11/TMOFL
31	P12/TMOFH	P12/UD	PD2/SEG43	P12	P12/TMOFH
32	P13/TMIG	P13/TMIC	PD3/SEG44	P13	P13/TMIG
33	P14/PWM	P14/PWM	PD4/SEG45	P14/PWM	P14
34	P15/TMIB//IRQ1	P15//IRQ1	PD5/SEG46	P15//IRQ1	P15/TMIB1
35	P16/TMIC//IRQ2	P16//IRQ2	PD6/SEG47	P16//IRQ2	P16/TMIC
36	P17/TMIF//IRQ3	P17//TRGV//IRQ3	PD7/SEG48	P17//IRQ3/TRGV	P17/TMIF
37	GND	GND	GND	GND	GND
38	`	`	`	`	`
39	P37//CS	`	P10/TMOW	P20/SCK3	P37/PWM7
40	P36/STRB	`	P11/TMOFL	P21/RXD	P36/PWM6
41	P35/SO2	`	P12/TMOFH	P22/TXD	P35/PWM5
42	P34/SI2	`	P13/TMIG	P23	P34/PWM4
43	P33/SCK2	`	P14	P24	P33/PWM3
44	P32/SO1	`	P15//IRQ1	P25	P32/PWM2
45	P31/SI1	`	P16//IRQ2	P26	P31/PWM1
46	P30/SCK1	`	P17//IRQ3/TMIF	P27	P30/PWM0
47	GND	GND	GND	GND	GND
48	`	`	`	P90	PA5
49	`	`	`	P91	PA4
50	PA3/COM4	`	PA3/COM4	P92	PA3
51	PA2/COM3	`	PA2/COM3	P93	PA2
52	PA1/COM2	`	PA1/COM2	P94	PA1
53	PA0/COM1	`	PA0/COM1	`	PA0
54	GND	GND	GND	GND	GND
55	V1	`	V1	NC	`
56	GND	GND	GND	GND	GND
57	V2	`	V2	NC	`
58	GND	GND	GND	GND	GND
59	V3	`	V3	NC	`
60	GND	GND	GND	GND	GND

Note: SD97ALE3834 H8/3834(U)(S), H8/3814 (U)(S)
SD97ALE3927 H8/3927
SD97ALE3947 H8/3947
SD97ALE3644 H8/3644 (R), H8/3657
SD97ALE 3876 H8/3877(U)(N)
SD97ALE 3877R H8/3877(U)(N), H8/3627, H8/3637

ALE-50P(USERCON 1)	
Pin No	**SD97ALE3734**
1	GND
2	P03/AN3
3	P02/AN2
4	P01/AN1
5	P00/AN0
6	AVCC
7	`
8	`
9	OSC1
10	`
11	/RES
12	P10//IRQ0
13	P11//IRQ1
14	P12//IRQ2
15	P13//IRQ3
16	P14//IRQ4
17	P15//IRQ5/TMOE
18	P16//EVENT
19	`
20	`
21	`
22	`
23	`
24	`
25	GND
26	`
27	`
28	`
29	`
30	`
31	`
32	P33/FS27
33	P32/FS26
34	P31/FS25
35	P30/FS24
36	P47/FS23
37	P46/FS22
38	P45/FS21
39	P44/FS20
40	P43/FS19
41	P42/FS18
42	P41/FS17
43	P40/FS16
44	P50/FS15
45	P51/FS14
46	P52/FS13
47	P53/FS12
48	P54/FS11
49	P55/FS10
50	P56/FS9

ALE-50P(USERCON 2)	
Pin No	**SD97ALE3734**
1	GND
2	P57/FS8
3	P17/VDISP
4	P60/FS7/FD0
5	P61/FS6/FD1
6	P62/FS5/FD2
7	P63/FS4/FD3
8	P64/FS3/FD4
9	P65/FS2/FD5
10	P66/FS1/FD6
11	P67/FS0/FD7
12	P70/FD8
13	P71/FD9
14	P72/FD10
15	P73/FD11
16	P74/FD12
17	P75/FD13
18	P76/FD14
19	P77/FD15
20	VCC
21	P80
22	P81
23	P82
24	P83
25	GND
26	P84
27	P85
28	P86
29	P87
30	P90/PWM
31	P91/SCK1
32	P92/SI1
33	P93/SO1
34	P94/SCK2
35	P95/SI2//CS
36	P96/SO2
37	P97/UD
38	PA0
39	PA1
40	`
41	`
42	`
43	`
44	`
45	`
46	AVSS
47	P07/AN7
48	P06/AN6
49	P05/AN5
50	P04/AN4

Note: SD97ALE3734 H8/3724, H8/3714, H8/3614, H8/3754

ALE-60P(USERCON 1)	
Pin No	**SD98ALE3880 SD00ALE3880R**
1	GND
2	OSC1
3	`
4	GND
5	EMVCC
6	`
7	AVSS
8	PB7/AN7
9	UVCCOUT
10	P97/SEG40
11	AVCC
12	GND
13	P96/SEG39
14	P95/SEG38
15	P42/TXD32
16	P43/~IRQ0
17	GND
18	P94/SEG37
19	PB0/AN0
20	PB1/AN1
21	P93/SEG36
22	AVSS
23	GND
24	AVSS
25	PB2/AN2
26	PB3/AN3
27	PB4/AN4
28	PB5/AN5
29	PC0/AN8
30	PC1/AN9
31	PC2/AN10
32	PC3/AN11
33	AVSS
34	PB6/AN6
35	P16/~IRQ2
36	P15/~IRQ1/TMIC
37	P14/~IRQ4/~ADTRG
38	P13/TMIG
39	P12/TMOFH
40	P11/TMOFL
41	P10/TMOW
42	/RES
43	GND
44	P17/~IRQ3/TMIF
45	P30/PWM
46	P31/UD
47	P32/~RESO
48	P33/SCK31
49	P34/RXD31
50	P35/TXD31
51	P36/AEVH
52	GND
53	PA7
54	PA6
55	PA5
56	PA4
57	P92/SEG35
58	P91/SEG34
59	P90/SEG33
60	GND

ALE-60P(USERCON 2)	
Pin No	**SD98ALE3880 SD00ALE3880R**
1	V1
2	V2
3	V3
4	GND
5	VCC
6	P51/~WKP1/SEG2
7	P50/~WKP0/SEG1
8	PA0/COM1
9	PA1/COM2
10	PA2/COM3
11	PA3/COM4
12	V0
13	GND
14	P61/SEG10
15	P60/SEG9
16	P57/~WKP7/SEG8
17	P56/~WKP6/SEG7
18	P55/~WKP5/SEG6
19	P54/~WKP4/SEG5
20	P53/~WKP3/SEG4
21	P52/~WKP2/SEG3
22	GND
23	P71/SEG18
24	P70/SEG17
25	P67/SEG16
26	P66/SEG15
27	P65/SEG14
28	P64/SEG13
29	P63/SEG12
30	P62/SEG11
31	GND
32	P81/SEG26
33	P80/SEG25
34	P77/SEG24
35	P76/SEG23
36	P75/SEG22
37	P74/SEG21
38	P73/SEG20
39	P72/SEG19
40	GND
41	P82/SEG27
42	P83/SEG28
43	P84/SEG29
44	P85/SEG30
45	P86/SEG31
46	P87/SEG32
47	P40/SCK32
48	P41/RXD32
49	GND
50	P27
51	P26
52	P25
53	P24
54	P23
55	P22/SO1
56	P21/SI1
57	P20/SCK1
58	P37/AEVL
59	X1
60	GND

Note: SD98ALE3880 H8/3827, H8/3847, H8/3867, H8/3887
SD00ALE3880R H8/3827(R)(S), H8/3847(R)(S), H8/3867, H8/3887, H8/3937R, H8/3827S, H8/3847S.

ALE-60P(USERCON 1)	
Pin No	**SD00ALE3800**
1	OSC1
2	AVCC
3	GND
4	AVSS
5	X1
6	UVCC
7	GND
8	GND
9	P11
10	P10
11	P13
12	P12
13	P15
14	P14
15	P17
16	P16
17	P21
18	P20
19	P23
20	P22
21	P25
22	P24
23	P27
24	P26
25	P31/TMOFL
26	P30
27	P33
28	P32/TMOFH
29	P35
30	P34
31	P37/AEVL
32	P36/AEVH
33	/RES
34	GND
35	IRQAEC
36	NC
37	P81
38	P80/SEG25
39	P83
40	P82
41	P85
42	P84
43	P87
44	P86
45	P91/PWM2
46	P90/PWM1
47	P93
48	P92
49	P95
50	P94
51	P96
52	NC
53	PA1/COM2
54	PA0COM1
55	PA3/COM4
56	PA2/COM3
57	PA5
58	PA4
59	PA7
60	PA6

ALE-60P(USERCON 2)	
Pin No	**SD00ALE3800**
1	GND
2	GND
3	V1
4	V0
5	V3
6	V2
7	GND
8	UVCC
9	P41/RXD32
10	P40/SCK32
11	P43/~IRQ0
12	P42/TXD32
13	P51/SEG2
14	P50/SEG1
15	P53/SEG4
16	P52/SEG3
17	P55/SEG6
18	P54/SEG5
19	P57/SEG8
20	P56/SEG7
21	P61/SEG10
22	P60/SEG9
23	P63/SEG12
24	P62/SEG11
25	P65/SEG14
26	P64/SEG13
27	P67/SEG16
28	P66/SEG15
29	P71/SEG18
30	P70/SEG17
31	P73/SEG20
32	P72/SEG19
33	P75/SEG22
34	P74/SEG21
35	P77/SEG24
36	P76/SEG23
37	NC
38	NC
39	PB1/AN1
40	PB0AN0
41	PB3/AN3
42	PB2/AN2
43	PB5
44	PB4
45	PB7
46	PB6
47	PC1
48	PC0
49	PC3
50	PC2
51	OPTION1
52	OPTION2
53	OPTION3
54	OPTION4
55	OPTION5
56	TONED
57	VTREG
58	VCREF
59	GND
60	GND

Note: SD00ALE3800
H8/3802, H8/3801, H8/3800, H8/38024, H8/38023, H8/38022, H8/38021, H8/38020.

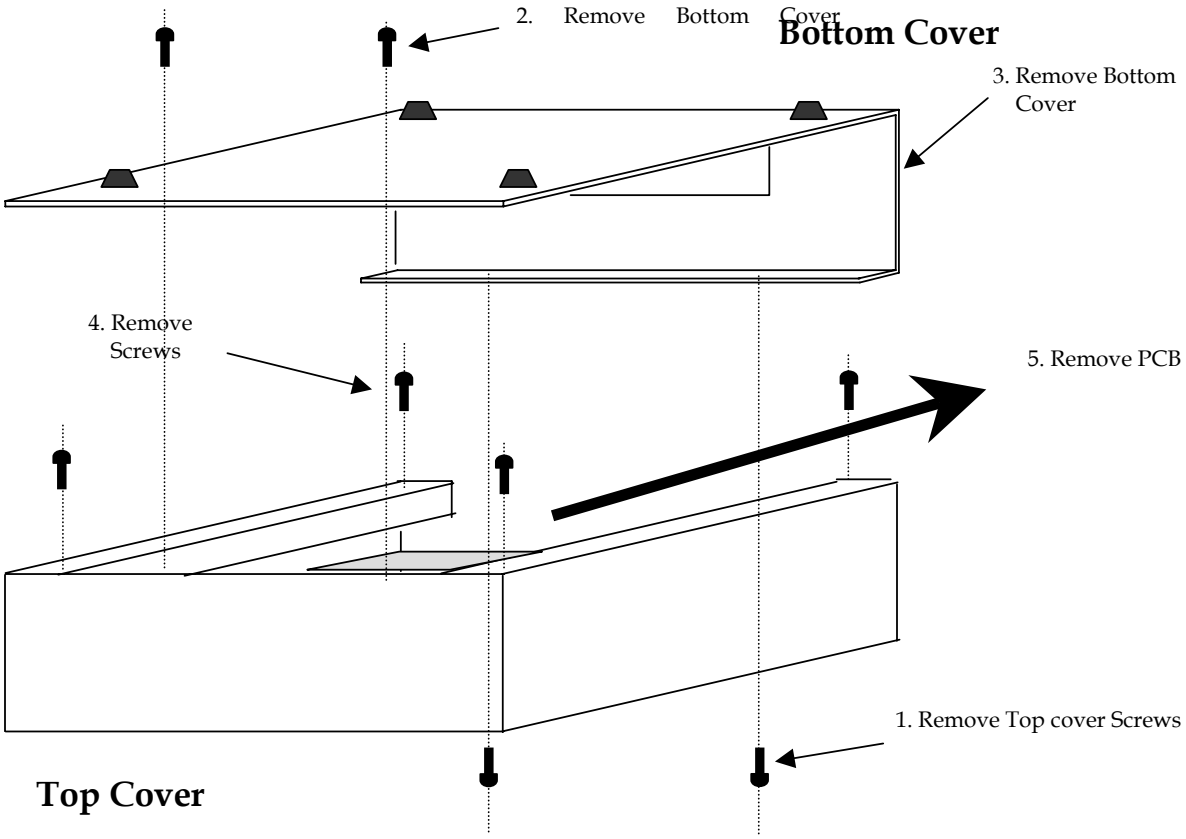
APPENDIX B: ALE300L Part Number Look-up Table

MCU Series	ALE300L Package	ALE Standard User Cable	ALE User Cable Adapter
H8/3614 H8/3714 H8/3724	SD97ALE3734	ALE –50P	ALE-100A
H8/3814(U)(S) H8/3834(U)(S)	SD97ALE3834	ALE –60P	ALE-130A ALE-160A
H8/3877(U)(N)	SD97ALE3876		
H8/3877(U)(N) H8/3637 H8/3627	SD98ALE3877R		
H8/3927	SD97ALE3927		
H8/3644(R) H8/3657	SD97ALE3644		ALE-160A
H8/3947	SD97ALE3947		ALE-130A
H8/3827 H8/3847 H8/3867 H8/3887	SD98ALE3880		-
H8/3837R H8/3847R H8/3937R	SD00ALE3880R		-
H8/3802 H8/38024	SD00ALE3800		-

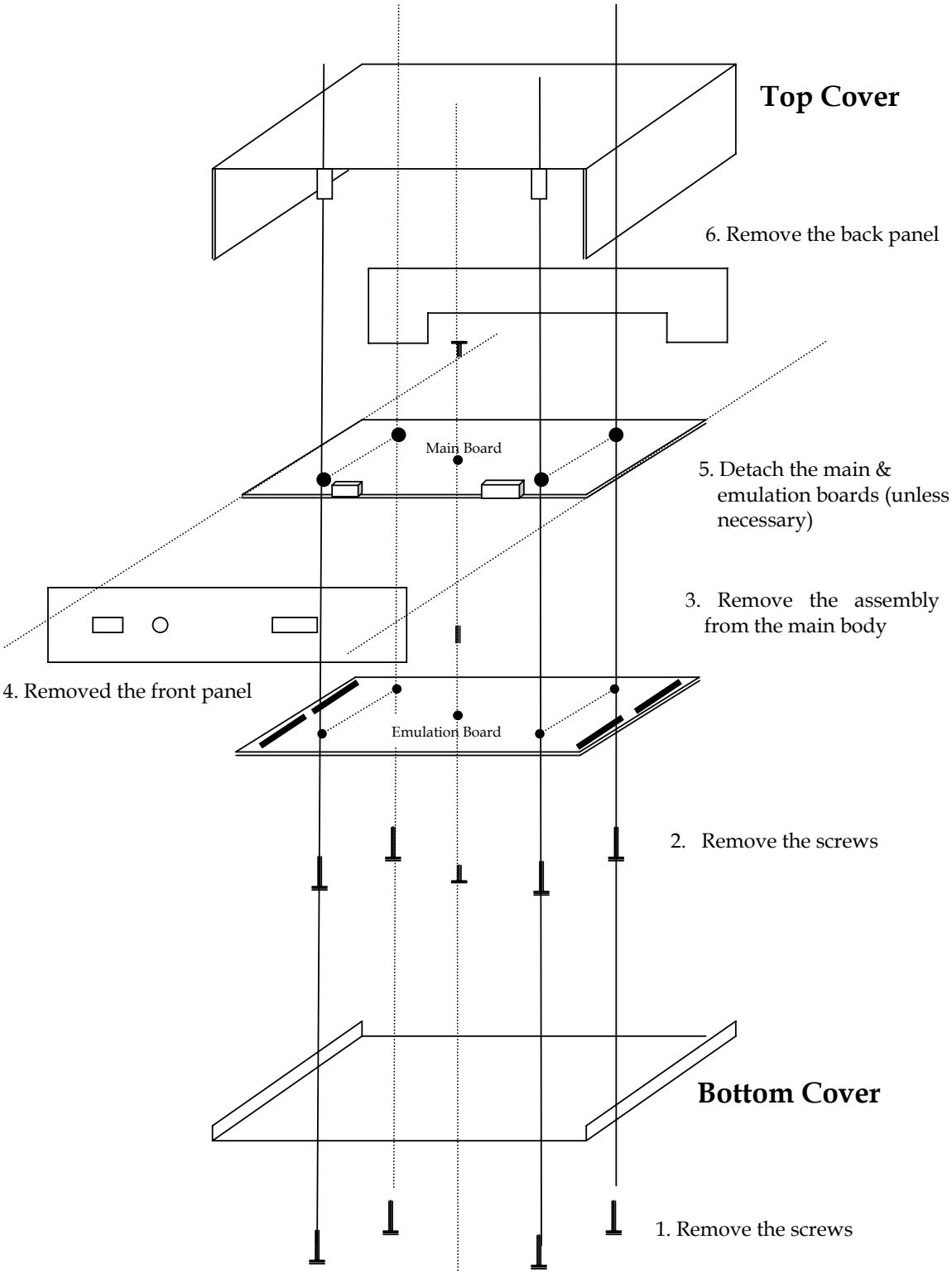
Note: Please contacts yours nearest sales office for latest updates.
The Standard User Cable & Adapter

APPENDIX C: ALE300L Assembly

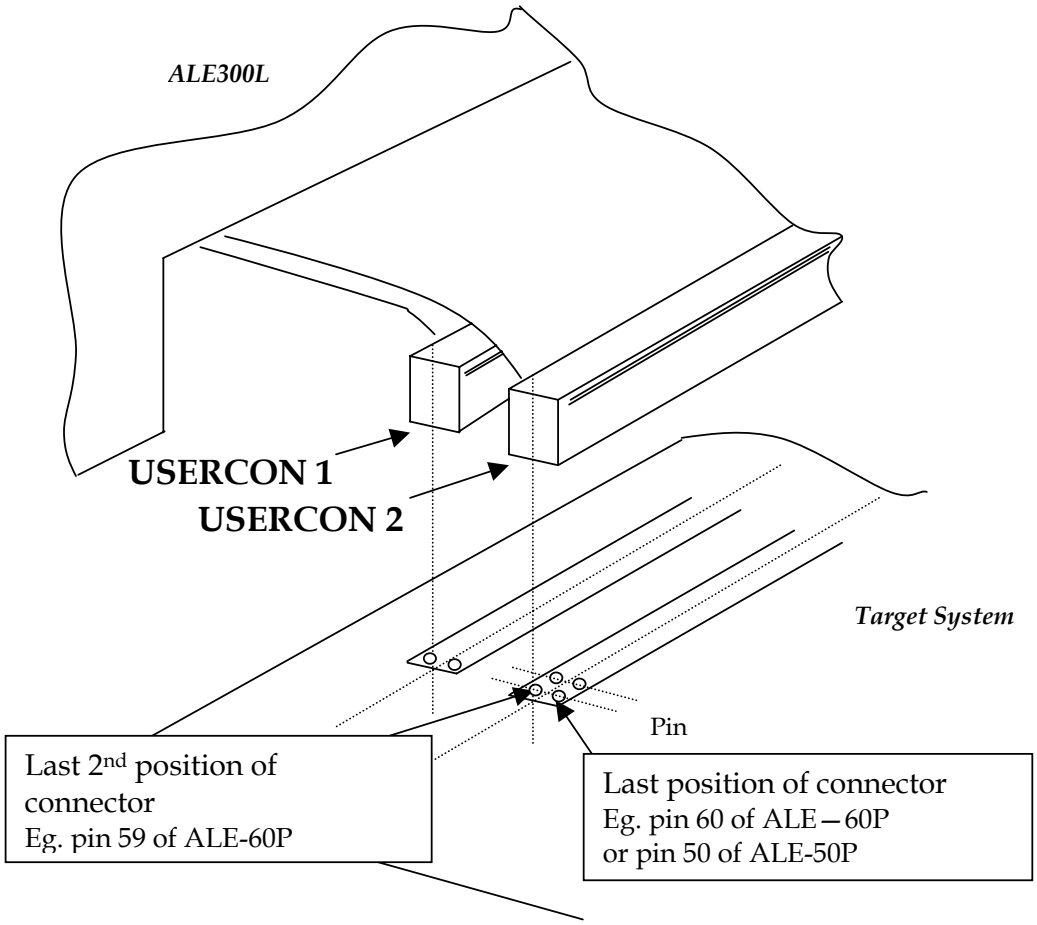
Assembly for Type 2



Assembly for Type 1



APPENDIX D: ALE300L User Connector Layout



APPENDIX E: Frequently Asked Questions Ver1.2

1. How to startup ALE300L Emulator?

2. Repetitive prompt by HEW for Serial Communication Setting?

3. How to communicate using baudrate 115,200 in Microsoft Windows 3.1?

4. Cannot communicate when using ALE300L with Win NT?

5. How to connect Vdisp pin(available for mask-rom version only) for 3724 series MCU?

6. How to use MOS pull-up Ports(available for mask-rom version only) for 3724 series MCU?

7. 3834 / 3880 series' Port A malfunction?

8. Why is the reading of ADC not accurate?

9. Signal degradation?

10. Can the Sub-Clock be accessed by the target system?

11. Can users acces these user pins; X1, X2, OSC2 & TEST?

12. Select Target or External Clock?

13. No input Power?

14. Not enough drive for Port 6 CMOS large current ports (3644 series)?

15. Flash memory function not working?

16. How to connect User/Target Power supply?

17. How is the user/target system connected the ALE300L

18. What type of code can be downloaded to the ALE300L?

19. What is the different between the actual MCU operation and the ALE300L Emualtion of the MCU?

20. Can users download another device Mapping Information for the development of a new device?

21. Does ALE300L have the built-in spilt resistors for the embedded LCD controllers?

22. Does peripherals continue to operate in break mode?

23. Failure encounters in Diagnostic window for test item 'Combination Address Break'.

1 How to startup ALE300L Emulator?

Ensure that the Serial Cable is plugged into ALE300L Emulator and the other end is connected to any avaliable Serial Communication Port at the Host PC side. Execute High-Performance Embedded Workshop (HEW2.EXE) at Host PC. Everything should startup smoothly with HEW showing “Linkup” at its status bar once communication between ALE300L and Host PC is established..

2 Repetitive prompt by HEW for Serial Communication Setting?

Both hardware and software Serial Communication setting should be identitical for communication between ALE300L and Host PC to be successful. During shippment, ALE300L is configured to communicate at 115,200 baudrate (max.). Thus when HEW prompt for Serial Communication Setting, select the correct Port & Baudrate, and click OK. HEW will display “Linkup” once it is able to detect ALE300L hardware. If HEW prompt for the Serial Communication Setting again, Users will have to verify the hardware setting of the Serial Communication inside ALE300L Emulator.

3 How to communicate using baudrate 115,200 in Microsoft Windows 3.1?

To communicate at 115200 bps, Users have to download the updated communication driver from the internet. Users can obtain this source from most search engine by using the keyword ”serial communication driver”.

4 Cannot communicate when using ALE300L with Win NT?

There is two possibilities;
1. Only HEW version 1.1 and above can support Win NT.
2. Currently Win NT can only support upto 57600bps. User has to changed the ALE300L BaudRate Dip Switch setting, located on the ALE main board.

5 How to connect Vdisp pin(available for mask-rom version only) for 3724 series MCU?

ALE300L do not have built-in pull-down resistor connected to -Vdisp. Therefore users have to connect pull down resistors (200Kohm) in between the VFD output and the Vdisp pins.

6 How to use MOS pull-up Ports(available for mask-rom version only) for 3724 series MCU?

If MOS pull-up ports are enabled, Users have to connect pull-up resistors (470Kohm) in between the User Vcc and the ports (port 0, 1, 8, 9 & A) pins.

7 3834 / 3880 series’ Port A malfunction?

User have to change the Dip SW 1 Way 2 to off. (refer to section 3.2.3 of the user manual)

8 Why is the reading of ADC not accurate?

The analog signal from the target board has to travel across the user cable and connector in order to be converted by the ADC. Moreover ADC has an absolute accuracy of ±2.5 LSB. Users are advised to calculate and compensate for the losses.

9 Signal degradation?

If users used a long user cable, the signal degradation in the user cable & connector, may cause problems to the target system. Users are advised to take the signal delay & signal drive capblilties into consideration during their troubleshooting.

10 Can the Sub-Clock be accessed by the target system?

Users are reminded that the sub-clock is supplied internally by ALE300L. User has no access to this feature (except to tap the signal from TP16). In the case of ALE H8/3880 series, user are allowed to input sub clock through the user connector.

11 Can users acces these user pins; X1, X2, OSC2 & TEST?

These pins have no connection to the user target system. Users have no access to these pins. (ALE H8/3880 is an exception, X1 is connected to user target suystem.).

12 Select Target or External Clock?

The Target Clock (or External Clock) is used if only the clock selector jumper is set to 'TGT' (or EXT). Moreover the input target clock (external clock) from the user system (or function generaator) must be a proper 50% duty cycle, 15ns(max) rise/fall time 'CLOCK' signal.

13 No input Power?

Excessive current drawn or wrong polarity input may blow the protective device of ALE300L, causing a cut-off of supply. User may verify the cause by checking the fuse (F1) and diode (D3) located near to the power switch.

14 Not enough drive for Port 6 CMOS large current ports (3644 series)?

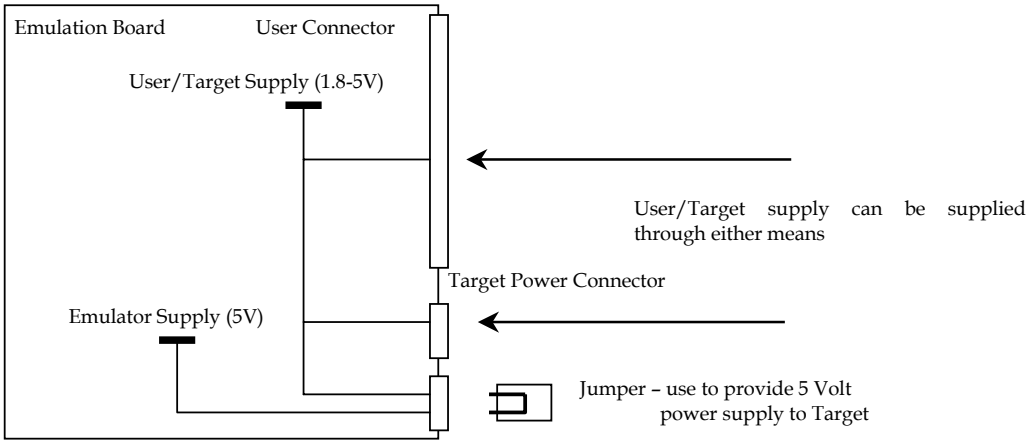
This can be solved if Japan user cable is used. The Japan user cable contains an AC244 driver to drive the user system.

15 Flash memory function not working?

ALE300L do not have flash support.

16 How to connect User/Target Power supply?

Unlike E1000 and E6000, ALE300L target power supply can be supplied by the emulator (5 Volt only). This is to ease the users to have a quick startup. If otherwise users can use power supply from the target system. In ALE300L, if users are to use their target power supply, about 10mA of current will be consumed by the emulator.

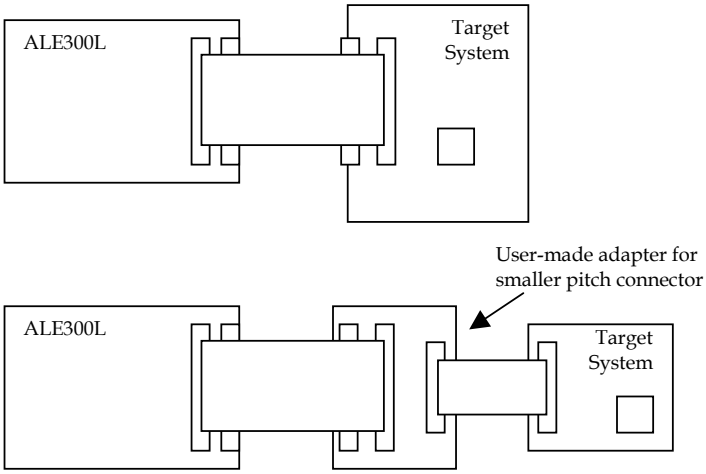


17 How is the user/target system connected the ALE300L

There are three methods to connect target system to the ALE300L.

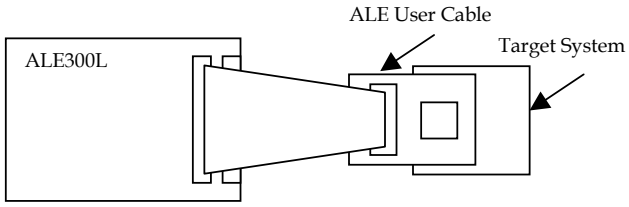
17.1 Direct

ALE300L uses the readily available standard cable (0.05” pitch). This provides a simple mean for user to connect this to their target. Eg.



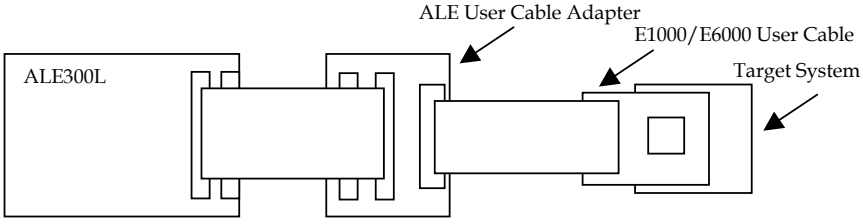
17.2 ALE User Cable

ALE provides several types of user cable of higher demand. This cable will be able to plug onto the actual selected MCU footprint (An IC socket must be avaiable)



17.3 E1000/E6000 User Cable – through ALE User Cable Adaptor

Since the whole MCU series are supported by E1000/E6000 user cables, ALE300L will support these user cables. (eg E1000 & E6000 user cable).



18 What type of code can be downloaded to the ALE300L?

ALE300L supports Motorola S-type record and SYSROF Format.

19 What is the different between the actual MCU operation and the ALE300L Emulation of the MCU?

The emulator provides a control enviroment for the users to manipulate the MCU operation. Thus the users have to take note of the following points;
Initiazation of stack pointer.
Introduction of capacitive effect due to the ALE cable and connectors.
Introduction of ALE user interface protective circuitries.
Comsumption of target/user current by the emulator.
Pin connection of X1, X2, OSC1, OSC2, TEST, CVCC…

20 Can users download another device Mapping Information for the development of a new device?

This is possible if the two devices are almost identical and the users know the actual new device specification. These Mapping Information contain the ROM/RAM/RESERVED address range. They will limit the execution of user program in reserved area, writing of data into ROM area,…Moreover the Mapping Information will provide the HEW with all the registers details. These will help the users to debug their program more efficiently.
In general, the Mapping Information provides a guide to help users to debug their program. It will not affect the users' final product, in term of timing, current consumption, and etc.

21 Does ALE300L have the built-in split resistors for the embedded LCD controller?

No. The user need to build an external LCD circuitry to boost for the LCD signals, as recommended by the Hardware Manual.

22 Does peripherals continues to operate in the Break mode?

Yes, peripherals such as timers continue to operate in Break mode, this is the limitation of the H8/300L evalution chip in the emulator. Therefore, pls. take note that the operation of the peripherals will be affected.

23 Failure encounters in Diagnostic window for test item 'Combination Address Break'.

The emulator could only support this diagnostic test at operating frequency 10MHz and below. Please perform this test under the specified condition as mentioned.

24 Failure encounters in Register window on CCR (I-bit) during Single –Step/Run.

This problem is resolved with the latest HDI and HEW.

Renesas Technology (Asia Sales Offices)

URL: http://www.renesas.com
URL: http://www.sg.renesas.com/sales

ASIA HEADQUARTERS & TECHNICAL SUPPORT :	
South Asia Headquarters : Singapore Renesas Technology Singapore Pte. Ltd. 1, HarbourFront Avenue, #06-10, Keppel Bay Tower, Singapore 098632. Tel : (65)-6213-0200 Fax : (65)-6278-8001 Email : contact.singapore@renesas.com	**Technical Support** Renesas System Solutions Asia Pte. Ltd. 1, Harbourfront Avenue, #06-06, Keppel Bay Tower, Singapore 098632. Tel : (65)-6213-0333 and 6387-2839 Fax : (65)-6278-1226
North Asia Headquarters : Hong Kong Renesas Technology Hong Kong Ltd. 7/F., North Tower, World Finance Centre, Harbour City, Canton Road, Hong Kong. Tel: (852) 2265-6688 Fax: (852) 2375-6836 Email : contact.hongkong@renesas.com	
ASIA SALES OFFICES :	
China	
Renesas Technology Hong Kong Ltd Shenzhen Representative Office Unit 1511-12, Shun Hing Square Di Wang Commercial Centre, 5002 Shennan Road East, Shenzhen City 518008, China Tel : (86) (755) 8246-1711 Fax : (86) (755) 8246-1728 Email : contact.china@renesas.com URL : http://www.cn.renesas.com	
Renesas Technology (Shanghai) Co., Ltd. 26/F., Ruijin Building, No. 205 Maoming Road (S), Shanghai 200020, China Tel : (86) (21) 6472-1001 Fax : (86) (21) 6415-2952 Email : contact.china@renesas.com URL : http://www.cn.renesas.com	
Renesas Technology (Shanghai) Co., Ltd. Beijing Office Room 1654, Office Building, New Century Hotel, No. 6 Southern Rd. Capital GYM., Beijing 100044, China Telex : 210509 HTCBJ CN Tel : (86) (10) 6849-2430 Fax : (86) (10) 6849-2819 Email : contact.china@renesas.com URL : http://www.cn.renesas.com	
Taipei	
Renesas Technology Taiwan Co., Ltd. (effective July 1, 2003) FL. 10, #99, Fu-Hsing N. Rd., Taipei, Taiwan Tel : (886)(2) 2715-2888 Fax : (886)(2) 2713-2999 Email : contact.taiwan@renesas.com URL : http://www.tw.renesas.com	

ALE300L

Renesas System Solutions Asia Pte. Ltd.