

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.



用户手册

ID78K0-QB Ver. 3.00

集成调试器

操作

目标设备

78K0 微控制器

文件编号: U18492CA1V0UM00 (第一版)

发布日期: 2007 年 8 月 CP(K)

© 日本电气电子株式会社 2007

日本印制

[备忘录]

IECUBE 是 NEC Electronic Corporation 在日本和德国的注册商标。

MINICUBE 是 NEC Electronic Corporation 在日本和德国的注册商标或在美国的商标。

Windows 是微软公司在美国和/或其他国家的注册商标或商标。

Pentium 是因特尔公司的商标。

- 本文档所刊登的内容有效期截至 2007 年 8 月。将来可能未经预先通知而更改。在实际进行生产设计时，请参阅各产品最新的数据表或数据手册等相关资料以获取本公司产品的最新规格。
- 并非所有的产品和/或型号都向每个国家供应。请向本公司销售代表查询产品供应及其他信息。
- 未经本公司事先书面许可，禁止复制或转载本文件中的内容。否则因本文档所登载内容引发的错误，本公司概不负责。
- 本公司对于因使用本文件中列明的本公司产品而引起的，对第三者的专利、版权以及其它知识产权的侵权行为概不负责。本文件登载的内容不应视为本公司对本公司或其他人所有的专利、版权以及其它知识产权作出任何明示或默示的许可及授权。
- 本文件中的电路、软件以及相关信息仅用以说明半导体产品的运作和应用实例。用户如在设备设计中应用本文件中的电路、软件以及相关信息，应自行负责。对于用户或其他人因使用了上述电路、软件以及相关信息而引起的任何损失，本公司概不负责。
- 虽然本公司致力于提高半导体产品的质量及可靠性，但用户应同意并知晓，我们仍然无法完全消除出现产品缺陷的可能。为了最大限度地减少因本公司半导体产品故障而引起的对人身、财产造成损害（包括死亡）的危险，用户务必在其设计中采用必要的安全措施，如冗余度、防火和防故障等安全设计。
- 本公司产品质量分为：

“标准等级”、“专业等级”以及“特殊等级”三种质量等级。

“特殊等级”仅适用于为特定用途而根据用户指定的质量保证程序所开发的日电电子产品。另外，各种日电电子产品的推荐用途取决于其质量等级，详见如下。用户在选用本公司的产品时，请事先确认产品的质量等级。

“标准等级”： 计算机，办公自动化设备，通信设备，测试和测量设备，音频·视频设备，家电，加工机械以及产业用机器人。

“专业等级”： 运输设备（汽车、火车、船舶等），交通用信号控制设备，防灾装置，防止犯罪装置，各种安全装置以及医疗设备（不包括专门为维持生命而设计的设备）。

“特殊等级”： 航空器械，宇航设备，海底中继设备，原子能控制系统，为了维持生命的医疗设备、用于维持生命的装置或系统等。

除在本公司半导体产品的数据表或数据手册等资料中另有特别规定以外，本公司半导体产品的质量等级均为“标准等级”。如果用户希望在本公司设计意图以外使用本公司半导体产品，务必事先与本公司销售代表联系以确认本公司是否同意为该项应用提供支持。

（注）

- （1）本声明中的“本公司”是指日本电气电子株式会社（NEC Electronics Corporation）及其控股公司。
- （2）本声明中的“本公司产品”是指所有由日本电气电子株式会社所开发或制造，或为日本电气电子株式会社（定义如上）开发或制造的产品。

[备忘录]

前言

目标读者	本手册适用于那些希望使用 78K0 微控制器进行设计和开发的工程师。		
目的	本手册的目的是让用户了解 ID78K0-QB 如下组件的功能。		
组件	本手册包括如下章节： • 概述 • 安装 • 启动和终止 • 与 PM+的联系 • 调试功能 • 窗口参考 • 命令参考		
如何使用本手册	本手册的读者应具备电气工程、逻辑电路、微控制器、C 语言和汇编程序的一般知识。 要了解 78K0 微控制器的功能： → 请参阅每个产品的《硬件用户手册》。 要了解 78K0 微控制器的指令功能： → 请参阅《78K/0 系列指令用户手册》（U12326E）。		
约定：	数字有效性：	高数位在左边、低数位在右边。	
	注：	文中标有“ 注 ”的项目的脚注。	
	注意：	需要特别注意的信息。	
	备注：	补充信息。	
	数值表示：	二进制 ……XXXX 或 XXXXB	
		十进制 ……XXXX	
		十六进制…… 0xXXXX	
	表示 2 次方的前缀（地址空间、内存容量）：	K（千） $2^{10} = 1024$	
		M（百万） $2^{20} = 1024^2$	
		G（千兆） $2^{30} = 1024^3$	

相关文件： 在使用本手册时请参阅以下文件。本出版物中显示的相关文件可能包括初期的版本，但初期的版本并未特别标注。

与开发工具相关的文件（用户手册）

文件名称		文件编号
QB-78K0KX1H 在线仿真器		U17081E
QB-78K0KX2 在线仿真器		U17341E
QB-780714 在线仿真器		U17366E
QB-78K0LX2 在线仿真器		U17468E
QB-78K0MINI 片上调试仿真器		U17029E
RA78K0 汇编程序包 V3.80	操作	U17199E
	汇编语言	U17198E
	结构化汇编语言	U17197E
CC78K0 C 编译器包 V3.70	操作	U17201E
	语言	U17200E
ID78K0-QB V3.00 集成调试器	操作	本手册
PM+ V5.20		U16934E

[备忘录]

目录

第1章	总览	18
1.1	特性	19
1.1.1	新功能，改进功能	19
1.1.2	其他	21
1.2	系统配置	22
1.3	操作环境	24
1.3.1	硬件环境	24
1.3.2	软件环境	24
1.4	调试期间注意事项	25
1.4.1	当执行源代码级调试时	25
1.4.2	安全 ID	25
第2章	安装	26
2.1	安装	26
2.2	卸载	26
第3章	启动和终止	27
3.1	启动前注意事项（当连接 MINICUBE 时）	27
3.2	启动选项和参数规格说明	27
3.2.1	规格说明方法	28
3.2.2	规格说明格式和选项	29
3.3	启动	30
3.4	终止	31
3.5	启动时产生的错误信息	32
3.5.1	当连接 IECUBE 时	32
3.5.2	当连接 MINICUBE 时	32
第4章	与 PM+ 相关联	33
4.1	设置 Build 模式	33
4.2	将调试器注册到 PM+ 工程中	34
4.2.1	选择调试器	34
4.3	从 PM+ 中启动 ID78K0-QB	35
4.3.1	恢复调试环境	35
4.4	自动加载	36
4.4.1	通过纠正源代码实现自动加载	36
4.4.2	通过启动调试器实现自动加载	37
第5章	调试功能	38
5.1	设置调试环境	39
5.1.1	设置操作环境	39
5.1.2	设置选项	39
5.1.3	设置映射	40
5.2	下载功能，上传功能	41
5.2.1	下载	40
5.2.2	上传	44
5.3	源代码显示，反汇编显示功能	46
5.3.1	源代码显示	46
5.3.2	反汇编显示	47
5.3.3	混合显示模式（源窗口）	47
5.3.4	转换符号（符号转地址）	48
5.4	中断功能	49
5.4.1	中断类型	49
5.4.2	断点设置	50
5.4.3	将中断设置为变量	51
5.4.4	硬件中断和软件中断	51
5.4.5	失效保护断点	52
5.5	程序执行功能	53

5.6 监视功能	55
5.6.1 显示和改变数据指	55
5.6.2 显示和改变局部变量值	56
5.6.3 注册恶化删除监视数据	56
5.6.4 改变监视数据	57
5.6.5 临时显示和改变数据值	57
5.6.6 回调查看功能	58
5.6.7 堆栈堆堆栈跟踪显示功能	58
5.7 存储器操控功能	59
5.7.1 显示和改变存储器内容	59
5.7.2 填充，复制和比较存储内容	59
5.7.3 访问监控器功能（当连接 IECUBE 时）	60
5.7.4 闪存写入功能（当连接 MINICUBE 时）	60
5.8 寄存器操控功能	61
5.8.1 显示和改变寄存器内容	61
5.8.2 显示和改变 SFR 内容	62
5.8.3 显示和改变 I/O 端口内容	62
5.9 计时器功能（当连接 IECUBE 时）	63
5.9.1 计时器事件功能	63
5.9.2 运行-中断事件	64
5.10 跟踪功能（当连接 IECUBE 时）	65
5.10.1 跟踪存储器	65
5.10.2 检查跟踪数据	65
5.10.3 混合显示模式（跟踪查看窗口）	66
5.10.4 跟踪器操作	66
5.10.5 设置条件跟踪	68
5.11 覆盖测试功能（当连接 IECUBE 时）	69
5.11.1 覆盖测试结果显示	69
5.11.2 覆盖测试范围	70
5.11.3 被执行的覆盖测试位置的显示	70
5.12 事件功能	72
5.12.1 使用事件功能	72
5.12.2 创建事件	73
5.12.3 设置事件状态	73
5.12.4 每个事件状态中有效事件数	75
5.12.5 管理事件	76
5.13 截图功能（当连接 IECUBE 时）	77
5.13.1 截图事件状态	77
5.13.2 捕获数据	77
5.14 存根功能（当连接 IECUBE 时）	78
5.14.1 设置存根事件状态	78
5.14.2 存根流功能	79
5.15 RRM 功能	80
5.15.1 实时 RAM 监控功能	80
5.15.2 伪实时 RAM 监控功能（当读取时中断）	80
5.16 DMM 功能	81
5.16.1 事件 DMM 状态（当连接 IECUBE 时）	82
5.17 加载/保持功能	83
5.17.1 调试环境（工程文件）	83
5.17.2 窗口显示信息（查看文件）	85
5.17.3 窗口设置信息（设置文件）	86
5.18 每个窗口共有的功能	87
5.18.1 活动状态和静止状态	87
5.18.2 跳转函数	87
5.18.3 使用链接窗口跟踪结果（当连接 IECUBE 时）	89
5.18.4 拖放功能	90
5.18.5 注意事项	92
第6章 窗口参考	93
6.1 窗口列表	94
6.2 窗口的解释	96
主窗口	97
配置对话框	110

- 扩展选项对话框...119
- 失效保护中断对话框...122
- RRM对话框...124
- 屏蔽选项对话框...127
- 闪存选项对话框...129
- 调试器选项对话框...132
- 伪仿真对话框...137
- 项目文件保存对话框...138
- 项目文件加载对话框...139
- 下载对话框...141
- 上传对话框...143
- 源窗口...145
- 源搜索对话框...150
- 源文本移动对话框...152
- 汇编窗口...154
- 汇编搜索对话框...159
- 地址移动对话框...161
- 符号转地址对话框...169
- 监视窗口...164
- 快速监视对话框...169
- 添加监视对话框...171
- 修改查看对话框...174
- 局部变量窗口...176
- 堆栈窗口...178
- 内存窗口...181
- 内存搜索对话框...185
- 内存填充对话框...187
- 内存复制对话框...189
- 内存比较对话框...190
- 内存比较结果对话框...192
- DMM对话框...193
- 寄存器窗口...196
- SFR窗口...199
- SFR选择对话框...203
- 添加I/O端口对话框...205
- 计时器对话框...207
- 计时器结果对话框...210
- 跟踪视图窗口对话框...211
- 跟踪搜索对话框...216
- 跟踪移动对话框...221
- 跟踪数据选择对话框...223
- 跟踪对话框...225
- 延迟计数对话框...227
- 代码覆盖窗口...228
- 软件中断管理器...231
- 事件管理器...233
- 事件对话框...239
- 事件联系对话框...244
- 中断对话框...247
- 截取对话框...249
- 存根对话框...253
- 事件DMM对话框...255
- 视图文件保存对话框...258
- 视图文件加载对话框...260
- 环境设置文件保存对话框...262
- 环境设置加载对话框...263
- 复位调试器对话框...264
- 退出调试器对话框...265
- 关于对话框...266
- 控制台窗口...268
- 浏览对话框...269
- 第7章 参考命令...271

7.1	命令行规则	272
7.2	命令目录	272
7.3	别名目录	274
7.4	变量目录	274
7.5	包目录	275
7.6	键绑定	275
7.7	扩展窗口	276
7.7.7	示例（计数器脚本）	276
7.8	调回程序	277
7.9	钩程序	278
7.10	相关文件	279
7.11	注意事项	279
7.12	命令解释	279
	地址	281
	集合	282
	批	283
	断点	284
	dbgexit	286
	下载	287
	擦除	288
	Extwin	289
	完成	290
	转到	291
	帮助	292
	钩	293
	检查	294
	跳转	295
	映射	296
	多文档界面	297
	存储器	298
	模块	299
	下一步	300
	刷新	301
	寄存器	302
	复位	303
	运行	304
	步骤	305
	停止	306
	上传	307
	译本	308
	监视	309
	位置	310
	请求	311
	X覆盖	312
	X计时	313
	X跟踪	314
	tkcon	315
附录A	扩展窗口	316
A.1	概述	316
A.2	扩展窗口目录样例	316
A.3	激活	316
A.4	每个窗口样例的解释	316
	列表窗口	317
	寻找字目录窗口	318
	钩窗口	319
	符号检查窗口	320
附录B	输入惯例	321
B.1	可用字符设置	321
B.2	符号	323

- B.3数值…323
- B.4表达式和操作符…323
- B.5文件名称…326

附录C 键功能列表…327

附录D 消息…330

- D.1显示格式…330
- D.2消息类型…330
- D.3消息目录…331

附录E 索引…353

图列表

图代号	标题	页面
1-1	ID78K0-QB…18	
1-2	ID78K0-QB系统配置实例 (IECUBE) …22	
1-3	ID78K0-QB系统配置实例 (MINICUBE) …23	
1-4	ID78K0-QB系统配置实例 (MINICUBE2) …23	
3-1	启动选项 (例子) …28	
3-2	配置对话框…30	
3-3	主窗口 (启动时) …31	
3-4	退出调试器对话框…31	
5-1	断点设置…50	
5-2	设置中断为变量…51	
5-3	软件中断管理…52	
5-4	失效保护断点设置…52	
5-5	执行按钮…53	
5-6	[Run]按钮…53	
5-7	查看窗口…55	
5-8	显示格式规格说明 (调试器选项对话框) …58	
5-9	局部变量窗口…56	
5-10	改变查看对话框…57	
5-11	快速查看对话框…57	
5-12	回调查看功能…58	
5-13	堆栈窗口…58	
5-14	访问监控功能 (存储器窗口) …60	
5-15	全名/函数名转换…61	
5-16	显示SFR内容…62	
5-17	注册I/O端口…62	
5-18	设置和显示计时器事件 (计时器对话框) …63	
5-19	检查跟踪数据…65	
5-20	覆盖测试结果显示…69	
5-21	被执行的覆盖测试位置的查看…71	
5-22	设置可变数据状态…73	
5-23	管理事件 (事件管理器) …76	
5-24	截图对话框…77	
5-25	设置存根功能状态…78	
5-26	存根流功能…79	
5-27	伪实时RAM监控功能的规格说明…80	
5-28	DMM对话框…81	
5-29	事件DMM对话框…82	
6-1	主窗口…97	
6-2	工具栏 (仅图片) …106	
6-3	工具栏 (图片和文本) …106	
6-4	状态栏…107	
6-5	配置对话框…110	
6-6	当使用内部ROM Bank (Bank ROM大小是40KB) 时地址空间图标. . 113	
6-7	扩展选项对话框…119	
6-8	失效保护断点对话框…122	
6-9	RRM对话框…124	
6-10	记号选项对话框…127	
6-11	闪存选项对话框…129	
6-12	调试器选项对话框…132	
6-13	[Add Source path] 对话框…133	
6-14	[Font] 对话框…134	
6-15	伪仿真对话框…137	
6-16	工程文件保存对话框…138	
6-17	工程文件加载对话框…139	
6-18	下载对话框…141	

6-19	上传对话框…143
6-20	源窗口…145
6-21	源搜索窗口…150
6-22	源文本移动对话框…152
6-23	汇编窗口…154
6-24	汇编搜索窗口…159
6-25	地址移动对话框（例子：当存储器窗口打开时）…161
6-26	符号转地址对话框…162
6-27	查看窗口…164
6-28	快速查看对话框…169
6-29	增加查看对话框…171
6-30	改变查看对话框…174
6-31	局部变量窗口…176
6-32	堆栈窗口…178
6-33	存储器窗口…181
6-34	存储器搜索对话框…185
6-35	存储器填充对话框…187
6-36	存储器复制对话框…189
6-37	存储器比较对话框…190
6-38	存储器比较结果对话框…192
6-39	DMM对话框…193
6-40	寄存器窗口…196
6-41	SRF窗口…199
6-42	SFR选择对话框…203
6-43	添加I/O端口对话框…205
6-44	计时器对话框…207
6-45	计时器结果对话框…210
6-46	跟踪查看窗口…211
6-47	跟踪搜索对话框…216
6-48	跟踪移动对话框…221
6-49	跟踪数据选择对话框…223
6-50	跟踪对话框…225
6-51	延迟计数对话框…227
6-52	代码覆盖窗口…228
6-53	软件中断管理器…231
6-54	事件管理器（在详细显示模式）…233
6-55	选择显示信息对话框…238
6-56	事件对话框…239
6-57	事件链接对话框…244
6-58	中断对话框…247
6-59	截图对话框（当选择“Register”时）…249
6-60	存根对话框…253
6-61	事件DMM对话框（当选择“Memory”时）…255
6-62	视图文件保存对话框…258
6-63	查看文件加载对话框…260
6-64	环境设置文件保存对话框…262
6-65	环境设置文件加载对话框…263
6-66	复位调试器复位调试器对话框…264
6-67	退出调试器对话框…265
6-68	关于关于对话框…266
6-69	控制台窗口…268
6-70	浏览对话框…269
7-1	执行屏幕…276
A-1	列表窗口…317
A-2	Grep窗口…318
A-3	钩窗口…319
A-4	SymInspect窗口…320
D-1	错误/警告消息…330

表列表

表代号 标题 页面

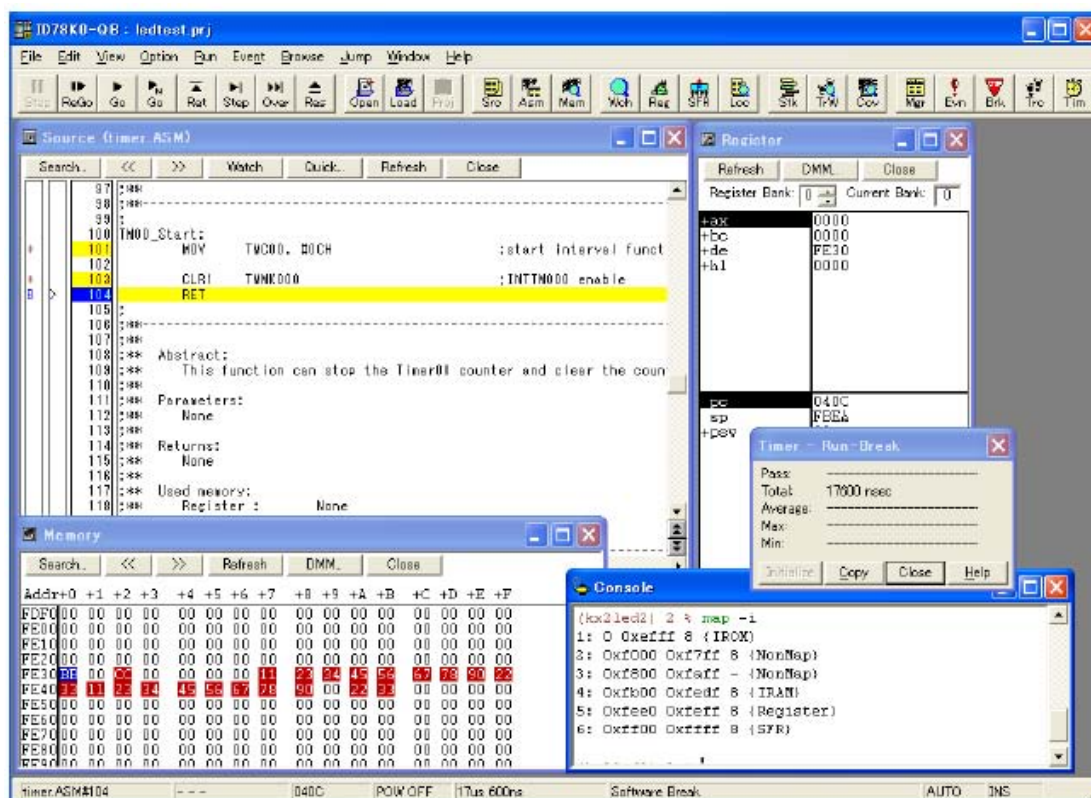
2-1	安装…26
3-1	启动选项…29
3-2	错误消息输出模式…32
5-1	调试功能列表（调试操作流）…38
5-2	映射属性…40
5-3	能被下载的文件类型…41
5-4	能被上传的文件类型…44
5-5	能显示的文件类型…46
5-6	指定符号…48
5-7	中断类型…49
5-8	有效软件中断数…51
5-9	执行类型…54
5-10	全面和函数名对应…61
5-11	跟踪存储大小…65
5-12	跟踪模式类型…66
5-13	跟踪控制模式类型…67
5-14	条件跟踪类型…68
5-15	代码覆盖测试范围…70
5-16	被执行的覆盖测试位置的显示格式…70
5-17	各种事件状态…72
5-18	每个事件状态有效事件数…75
5-19	事件图标…76
5-20	要被执行的函数的开始地址（存根功能）…78
5-21	实时RAM监控功能（采用范围）…80
5-22	保存到工程文件中的内容…83
5-23	查看文件类型…85
5-24	设置文件类型…86
5-25	跳转源地址详细信息…88
5-26	拖放功能的详细信息（行/地址）…90
5-27	拖放功能的详细信息（字符串）…91
6-1	窗口列表…94
6-2	CPU状态…107
6-3	IE状态…108
6-4	中断原因…108
6-5	内部ROM/RAM设置的范围和单位（当连接IECUBE时）…112
6-6	中断期间操作时钟（当连接MINICUBE时）…115
6-7	事件设置状态（事件记号）…146
6-8	查看窗口显示格式（符号）…165
6-9	查看窗口显示格式（数据）…166
6-10	查看窗口输出格式…172
6-11	当指定范围时怎样处理一个变量…172
6-12	可测量值…208
6-13	当跟踪器停止时中断原因…212
6-14	跟踪操作期间重新设置原因…213
6-15	地址状态可设置的范围（跟踪）…218
6-16	帧数规格说明格式…222
6-17	可设置事件数…226
6-18	显示事件细节信息的分隔符…234
6-19	状态条件…240
6-20	地址状态可设置的范围（事件）…241
6-21	事件状态可设置范围…242
6-22	通过计数…243
6-23	在事件链接对话框中事件状态数…245
6-24	在状态设置区域中可设置事件数…248
6-25	地址可设置范围（截图对话框）…251

6-26	捕获数据显示格式…252
6-27	事件DMM数据显示格式…257
7-1	调试器控制命令列表…272
7-2	控制台/Tcl命令列表…273
7-3	.tcl别名文件的内容…274
7-4	变量列表…274
7-5	包列表…275
7-6	消息ID…277
7-7	相关文件列表…279
A-1	扩展窗口列表（例子）…316
B-1	字符集列表…321
B-2	特别字符列表…321
B-3	数值输入格式…323
B-4	操作符列表…324
B-5	操作符优先级…325
B-6	基数范围…325
C-1	主要功能列表…327
D-1	消息类型…330

第1章 总览

用于78K0微控制器而设置的集成调试器ID78K0-QB（此后简称为ID78K0-QB）是一种为NEC Electronics 78K0微控制器开发的软件工具（嵌入式控制）。此软件开发工具目的是为了提高用户调试程序的效率。

图 1-1 ID78K0-QB



本章将介绍关于ID78K0-QB的：

- 特性
- 系统配置
- 操作环境
- 调试时注意事项

1.1 特性

ID78K0-QB有以下几个特性:

- 新功能, 改进功能
- 其他

1.1.1 新功能, 改进功能

(1) 支持IECUBE™、MINICUBE™ 和 MINICUBE2

三种仿真器(IECUBE, MINICUBE和MINICUBE2)都可以连接到1调试器上(ID78K0-QB) (参见“1.2 系统配置”)支持USB2.0。本文档出现的专业词“MINICUBE”包括“MINICUBE2”。

(2) 改进RRM功能 (当连接MINICUBE时)

现在可以为实时RAM监控区域的采样范围设置区域(参见“5.15 RRM功能”)。可以设置16字节RAM区域 (在1字节单位内可以有8个位置)。

(3) 改进定时功能 (当连接IECUBE时)

除了在Run-Break期间, 在其他任何时候都可以显示最大时间、最小时间、通过次数和平均时间 (参见“5.9 定时功能 (当连接 IECUBE 时)”)。支持在执行用户程序和暂停时显示测定时间。

(4) 失效保护断点保护支持

除了传统的保护区域和SFR区域, 也支持内部ROM和内部RAM的保护区域的失效保护断点 (参见“5.4.5 失效保护断点”)。

(5) 改进命令功能

在ID启动时的脚本文件规范说明 (参见“3.2 启动选项和参数规范说明”)。

通过指定工程文件 (和脚本文件同时) 点击1来完成测试。

将核心程序更新到最新V8.4。

(6) 程序执行时的设置 (当连接IECUBE时)

当执行用户程序时, 可以对定时器事件状态和跟踪事件状态进行设置。

(7) 显示硬件版本细节信息

硬件版本细节信息显示在关于 对话框中。

可以在启动前对配置对话框中版本信息进行确认。同时您可以对显示的信息进行拷贝和粘贴。

(8) 支持单字节空间

单字节空间用于记录文件名。

(9) 支持代码覆盖测试（当连接IECUBE时）

执行代码覆盖测量（C0覆盖）。

代码覆盖将显示在代码覆盖窗口中，对于被执行的覆盖测量部分将显示在源窗口和汇编窗口中（参见“[5.11 覆盖测试功能（当连接IECUBE时）](#)”）。

(10) 附加截图功能（当连接IECUBE时）

执行用户程序时，寄存器和SFR中的内容可以作为捕获的数据存储在跟踪存储器中（参见“[5.13 截图解图功能（当连接IECUBE时）](#)”）。

(11) 附加存根功能（当连接IECUBE时）

对于在建立事件时通过线性装配下载或书写到内存空间（未使用）中的用户程序（或子程序）可以直接执行（参见“[5.14 存根功能（当连接IECUBE时）](#)”）。

(12) 改进DMM功能

DMM(动态存储器修改)可以是存储器、寄存器或指定的SFR。

在执行用户程序期间，可以利用DMM功能实时的将数据书写到存储器当中（参见“[5.16 DMM功能](#)”）。

(13) 支持闪存自我编程错误仿真器（当连接IECUBE时）

闪存自我编程现在可以被调试（参见“[闪存选项对话框](#)”）。

(14) 支持多种版本

可以将产品的多个版本安装在同一个机器当中。

1.1.2 其他

(1) 使用内电路仿真器功能

利用内电路仿真器的事件设置功能，可以对断开事件进行设置、跟踪用户程序和测定时间等（参见“5.12事件功能”）。

(2) 支持片上调试(当连接 MINICUBE时)

由片上芯片调试组件来实现调试功能。

(3) 闪存写入功能(当连接 MINICUBE时)

将信息写入内部闪存当中，下载负载模块，他们的存取方法和一般存储器操作类似（参见“5.7.4闪存书写功能(当连接MINICUBE时)”）。

(4) 安全功能(当连接 MINICUBE时)

可以对存储在产品内部闪存中的ID代码进行安全鉴定（参见“配置对话框”，“(5) ID 代码”）。

(5) 通过Tcl进行功能扩展

使用Tcl/Tk命令行（工具命令语言）可以进行批处理、钩处理和初始客户窗口的创建（参见“第7章 参考命令”，“附录 A 扩展窗口”）。

1.2 系统配置

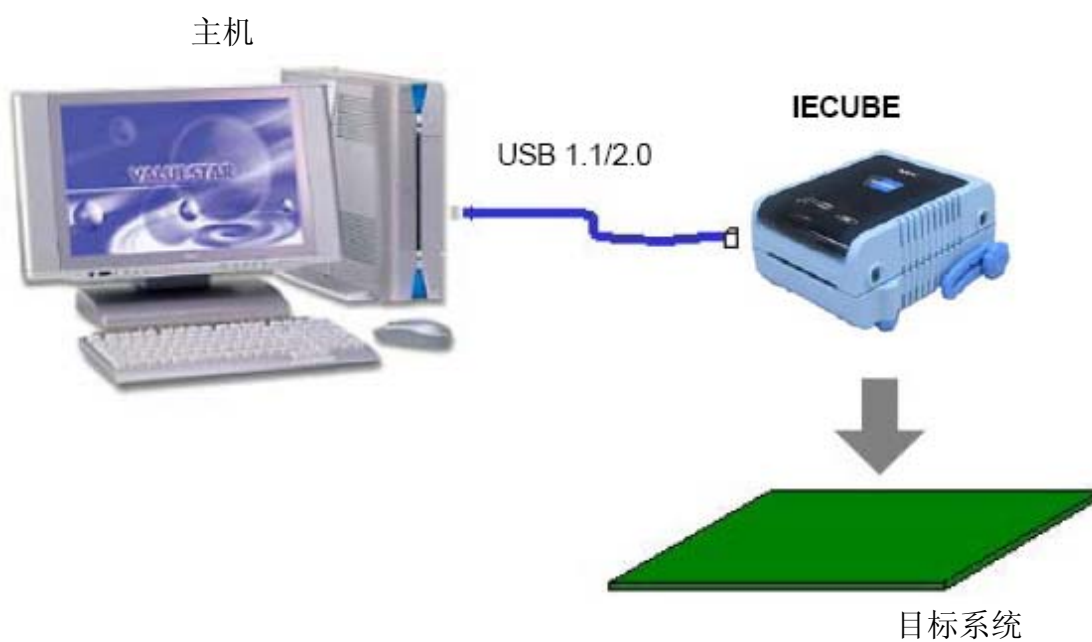
可以将ID78K0-QB连接到以下两种仿真器上。

提供开发用于78K0微控制器的用户程序和用于目标系统的友好调试环境。

(1) IECUBE（内电路仿真器）

由ID78K0-QB操控的IECUBE可以通过USB电缆将其连接到ID78K0-QB上。

图 1-2 ID78K0-QB 系统配置实例（IECUBE）



(2) MINICUBE, MINICUBE2 (片上调试仿真器)

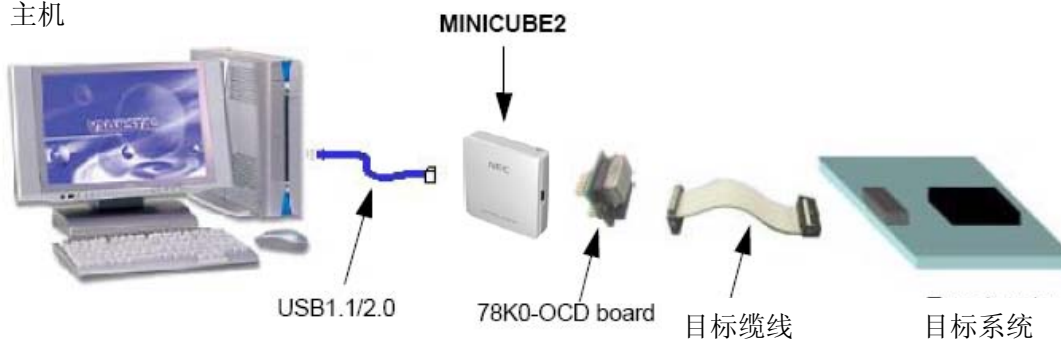
当利用USB线将MINICUBE、MINICUBE2和主机连接时，那么对他们进行的操作是通过ID78K0-QB来实现的。

图 1-3 ID78K0-QB 系统配置实例 (MINICUBE)
主机



图 1-4 ID78K0-QB 系统配置实例 (MINICUBE2)

主机



1.3 操作环境

本部分介绍以下几个操作环境：

-硬件环境

-软件环境

1.3.1 硬件环境

(1) 主机（目标操作系统操作的机器）

CPU	至少是奔腾 II™ 400MHz
主存	至少是256MB

(2) 仿真器

- IECUBE (QB-78K0KX2, 其他) (内电路仿真器)
- MINICUBE(QB-78K0MIN) (片上调试仿真器)
- MINICUBE2(QB-MINI2) (片上调试仿真器)

1.3.2 软件环境

(1) 操作系统（下面中的任何一种）

Windows® 2000, Windows XP(家庭版和专业版)

注意：不论您使用以上那种操作系统，我们都建议您要安装最新Service Pack。

(2) 设备文件(个人获取)

-使用目标设备的设备文件

记住：此文件可以通过下面NEC Electronics网站中获取（版本服务）。

<http://www.necel.com/micro/ods/eng/>

(3) 支持的软件开发工具（由公司NEC Electronics开发）

-汇编程序包RA78K0（V4.00以上）

-C编译器包CC78K0（V4.00以上）

-工程管理器PM+（V6.30以上）

1.4 调试期间注意事项

在调试期间需要注意：

- [执行源代码级调试时](#)
- [安全ID](#)

1.4.1 执行源代码级调试时

执行源代码级的调试时，目标文件必须包括调试中所用到的符号信息或其他信息（调试信息）。

在编译源文件期间进行下面的处理方式。

(1) 当使用PM+时

当选择Build模式时指定[Debug Build]。

(2) 以独立基础使用LK78K0

增加-g选项。

1.4.2 安全 ID

当连接MINICUBE时，使用的目标文件必须包括安全ID信息。

对于安全ID 78K0-QB设置，参见“RA78K0汇编程序包操控”内容。

对于安全ID细节方面的信息，参见MINICUBE用户手册。

在[配置对话框](#)中对ID78K0-QB中的安全ID（ID代码）进行设置。

第2章 安装

本章将介绍关于ID78K0-QB安装的内容：

-安装

-卸载

2.1 安装

当使用ID78K0-QB时，必须要安装下面几项。

表2-1安装

条目	过程
ID78K0-QB	当使用ID78K0-QB磁盘中的安装工具时： 根据自动执行的安装工具来安装此磁盘中内容。 当使用从Version-up Service网页中下载的安装工具时： 根据安装工具的指导说明来运行下载的可执行文件。
设备文件	选择[start]菜单→[All Programs]→[NEC Electronics Tools]→[DeviceFile Installer]，根据DFINST.exe专门启动安装工具来安装此文件

2.2 卸载

使用Control Panel上的[Add/Remove Programs]选项来卸载此程序。

第3章 启动和终止

本章将介绍关于ID78K0-QB启动和终止的几项内容：

- 启动前注意事项（当连接MINICUBE时）
- 启动选择和参数规范
- 启动
- 终止

3.1 启动前注意事项（当连接 MINICUBE 时）

当连接MINICUBE和MINICUBE2时，在启动ID78K0-QB之前要启动下面这个检查工具来检查仿真器和目标系统是否被正常调试。

.OCD Checker

注意：对于仿真器和目标系统之间的连接和启动应用程序顺序的相关信息，请参见INICUBE和MINICUBE2用户手册。不正确连接有可能损坏仿真器和目标系统。

3.2 启动选择和参数规范说明

下面将介绍指定ID78K0-QB中的启动选择和参数过程。

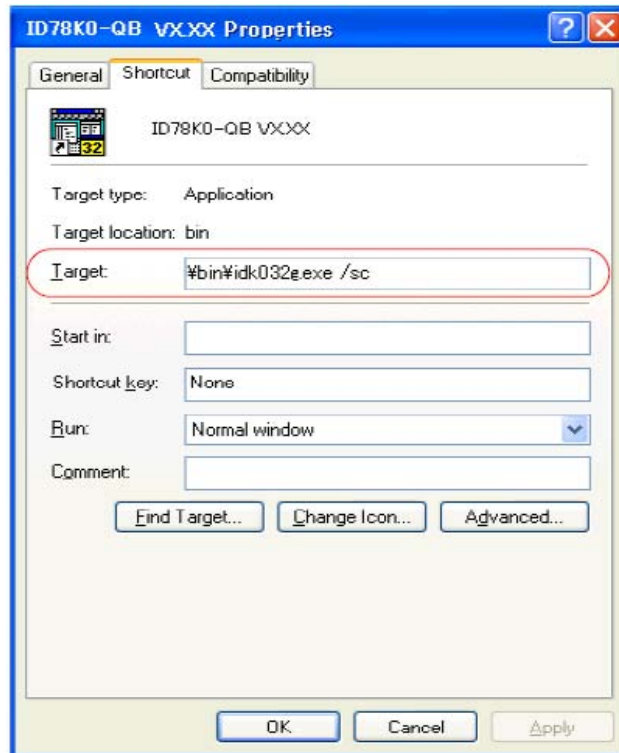
指定启动选择和参数时需要指定启动和工程文件中的脚本文件。

备注：当从PM+中启动ID78K0-QB时，就需要在PM+【工具】菜单中的【调试器设置】来对启动选择和参数进行设置（参见“第4章 和PM+关联”）。在选项栏中设置调试器启动选项。

3.2.1 规范说明方法

- 1) 在桌面上创建ID78K0-QB快捷方式。
将ID78K0-QB可执行文件放置在二进制文件夹中。并且在此路径中进行安装。
- 2) 打开创建的快捷属性。在[Target:]中显示可执行文件名之后，再指定此选项和参数（参见“3.2.2 规范说明格式和选项”）。

图 3-1 启动选项（例子）



3.2.2 规范说明格式和选项

(1) 规范说明格式

```
idk032g.exe ?options?  
idk032g.exe ?options? project
```

每个选项和参数都独立分开。在字符串中这种情况尤其明显。“？”之间的封装参数被忽略。
当指定一个工程文件时，那么此工程文件将在启动的时候就被读取。
但是在PM+启动期间，将忽略工程文件的规范说明。
当文件名和路径之间存有空间时，要使用双引号（“”）来指定封装的工程文件名和脚本文件名。（参见“例3）当路径之间存有空间时的规范说明”）。

(2) 规范说明选项

下面是指定的选项。

表3-1启动选项

选项	含义
/SC	改变窗口背景底色为系统颜色
/脚本：脚本文件名	在启动时指定可执行的脚本文件

(3) 规范说明实例

例1）仅脚本文件的规范说明

```
idk032g.exe /script:c:\work\script.tcl
```

例2）脚本文件和工程文件的规范说明

```
idk032g.exe /script:c:\work\script.tcl c:\work\project.prj
```

例3）当路径中存有空间时的规范说明

```
idk032g.exe /script:"c:\work folder\script.tcl" "c:\work folder\project.prj"
```

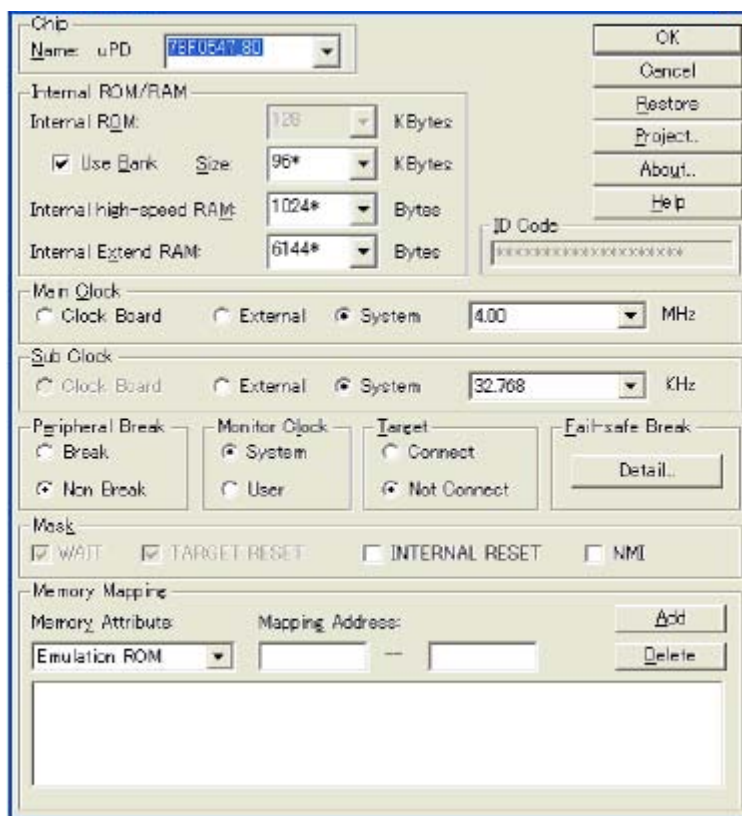
3.3 启动

1) 从PM+[Start]菜单中或点击桌面上创建的快捷方式来启动ID78K0-QB。

当从PM+启动ID78K0-QB时，请参见“4.3 从PM+中启动ID78K0-QB”中的内容。

启动ID78K0-QB时将会打开配置对话框。

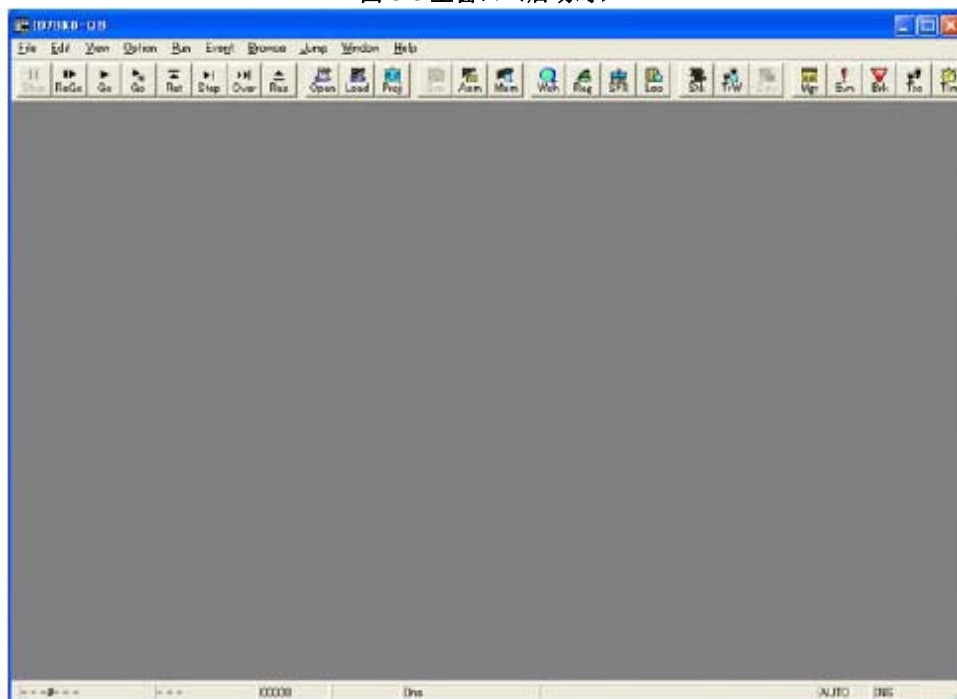
注意：在这种情况下不显示配置对话框，但是要显示错误消息，请参见“3.5启动时产生的错误消息”部分中相关的处理方式。



2) 在配置对话框中设置与ID78K0-QB操控环境相关的条目。设置完每个条目以后，再点击对话框中<OK>按钮。

3) 此时将打开主窗口并对ID78K0-QB进行操作。注意是使用此窗口进行调试。

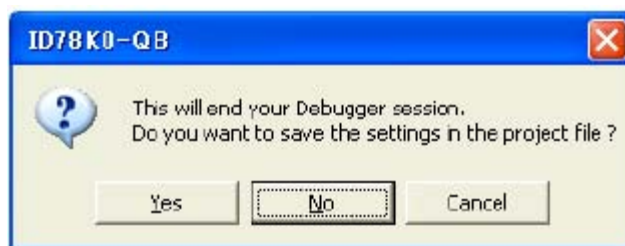
图 3-3 主窗口（启动时）



3.4 终止

1) 在主窗口中选择[File]菜单→[Exit]。将打开下面这个退出调试器对话框：（程序执行期间当进行终止操作步骤时将显示执行终止确认信息。）

图 3-4 退出调试器对话框



2) 点击按钮<Yes>将当前调试环境保存在工程文件中。
如果点击<N>按钮，所有的窗口将关闭，ID78K0-QB立即终止。

3.5 在启动时产生的错误消息

下面列出的是启动ID78K0-QB时产生的错误信息（按照发生顺序列出）。参见“附录 D 消息”中关于这些消息的描述。

3.5.1 当连接上 IECUBE 时：

根据与目标的连接状态及配置对话框中的设置，输出错误消息的模式有以下不同：

表3-2错误消息输出模式

错误消息	[Target]配置对话框中的区域		目标		交换适配器		目标电源	
	连接	未连接	连接	未连接	使用	未使用	开	关
Ff606: 请检查目标板的连接并启动它。	选择							选择
Wf607: 请检查交换适配器的连接		选择		选择		选择		选择
Ff608: 请断开和目标板连接。		选择	选择					选选
Ff609: 请关闭目标板电源并将其断开。		选择					选择	

3.5.2 当连接 MINICUBE 时

F0100: 不能和ICE进行通信。请确认PC界面设备驱动程序的安装。
F03a0: 目标没有打开
A0105: 读取设备文件时产生故障(d1xxx.78k)
F0ca2: 设备文件不包括片上调试信息。
F0ca3: 在设备文件的片上调试信息中包括不支持的信息。
A01a0: 没有得到仿真器CPU的回应。请确认CLOCK 或RESET信号等。

第4章 和PM+关联

ID78K0-QB可以自动地执行开发过程中的一系列操作，比如创建源文件→编译→调试→改正源文件，这都和PM+关联。本章介绍的是与PM+关联的内容。

对于PM+功能具体方面的细节信息，参见PM+用户手册。

- 设置Build模式
- 将调试器注册到PM+工程中
- 从PM+中启动ID78K0-QB
- 自动加载

注意：如果使用Windows命令提示来创建加载模块文件，那么与PM+关联的ID78K0-QB就不能使用此功能。

4.1 设置 Build 模式

为了在源代码级调试由PM+（在ID78K0-QB上）创建的加载模块文件，必须执行输出调试符号信息的结构，以此来创建一个加载模块。通过选择PM+上的[Debug Build]来进行此设置。

4.2 将调试器注册到 PM+工程中

为PM+中每个工程指定使用的调试器或要下载的加载模块文件。

4.2.1 选择调试器

选择调试器过程如下：

ID78K0-QB是注册作为活动工程的调试器。ID78K0-QB图标将显示在PM+工具条上。

(1) 创建新的工作站

- 1) 在PM+中选择[File]菜单→[New Workspace...]
→使用向导格式打开对话框来创建新工作区。
- 2) 使用向导工具为工作区进行必要设置，打开[Select Debugger]对话框。指定此对话框中的ID78K0-QB。
对于设置方面的细节，参见用户手册。

(2) 使用现有的工作区

- 1) 在PM+中选择[Tool]菜单→[Debugger Settings...]
→打开[Debugger Settings...]对话框。
- 2) 指定ID78K0-QB并在此对话框中点击<OK>按钮。对于设置方面的细节，参见用户手册。

注意：如果选择[Debugger Settings]对话框中的[Execute symbol reset after download]（此产品要有内部闪存），那么在下载之前要清除内部闪存中的内容（当连接MINICUBE时）。

4.3 启动 PM+ 中的 ID78K0-QB

从 PM+ 中启动 ID78K0-QB 的过程如下：

- 在 PM+ 工具栏中点击启动 ID78K0-QB 按钮。
- 在 PM+ 中选择 **【Build】** 菜单 → **【Debug】**。
- 在 PM+ 中选择 **【Build】** 菜单 → **【Debug and Debug】**。
- 在 PM+ 中选择 **【Build】** 菜单 → **【Debug and Debug】**。

如果将 ID78K0-QB 调试环境保存在工程文件中（当前使用 PM+），那么要从保存在工程文件中的调试环境中启动。

如果 ID78K0-QB 调试环境没有被保存在工程文件中（使用 PM+），打开 [配置对话框](#)。此时，设备型号（芯片名）不会改变。

注意：对于 PM+，如果在一个工程中注册过多的源文件，文件数量超过源路径长度的上限（要能被注册到 ID78K0-QB 中），那么源文件就不能自动显示。

关于源路径长度方面的细节信息，参见 [调试器选项对话框](#) 中的“(1) 源路径”。

4.3.1 恢复调试环境

当从 PM+ 中启动 ID78K0-QB 时，可以通过下面这个过程来恢复以前的调试环境：

- 1) 在 PM+ ^{Note} 中创建新的工作站（工程文件：例如，sample.prj）。
- 2) 从 PM+ 中启动 ID78K0-QB。在 [配置对话框](#) 中设置各个条目而不是设备型号（芯片名）。只有当 ID78K0-QB 启动时设置方式才相同。
- 3) 下载加载模块文件，在 ID78K0-QB [下载对话框](#) 中进行调试。
- 4) 在 ID78K0-QB 中调试加载模块文件。
- 5) 当 ID78K0-QB 终止时，在 [退出调试器对话框](#) 中点击 <Yes> 按钮。

→ 当 ID78K0-QB 终止时，将调试环境保存在 PM+ 工程文件（sample.prj）中（通过时常重写工程文件而不是完成 ID78K0-QB 调试，也可以将 sample.prj 调试环境保存在文件中。）。

- 6) sample.prj 文件被 PM+ 读取之后，当下次启动 ID78K0-QB 时，此点的调试环境（当工程文件已经被保存时）将自动恢复。

注意：在 ID78K0-QB 和 PM+ 中，环境信息将保存在工程文件中，以备参考。工程文件（要能被 ID78K0-QB 使用）的扩展名是“.prj”。对于被工程文件保存或恢复的信息，参见每种产品的“用户手册”。

4.4 自动加载

当使用ID78K0-QB对加载文件进行调试时，如果发现bug，可以使用下面这个过程来纠正源文件。编译和重新下载文件可以自动执行。（参见“4.4.1通过纠正源代码来实现自动加载”）

通过编译和连接PM+（激活的ID78K0-QB）中的文件，将加载文件下载到ID78K0-QB中。（参见“4.4.2通过启动调试器来实现自动加载”）

注意：如果选择的是PM+标准编辑器就不能实现此处理方式。

4.4.1 通过纠正源代码来实现自动加载

纠正源代码来实现自动加载的过程如下：

- 1) 载源窗口中打开要纠正的源文件。选择[File]菜单→[Open]，并指定ID78K0-QB中要纠正的文件（如果文件已经在源窗口中打开，此窗口要在最前端显示）。
→指定的文件将在源窗口中打开。
- 2) 在ID78K0-QB中选择[Edit]菜单→[Edit Source]。
→将打开编辑器并读取指定的文件。
- 3) 纠正编辑器中的源文件。
- 4) 终止编辑器。

注意：当加载文件不是自动下载时，就不能对CPU进行重新设置。当调用编辑器时，调试窗口将被打开，并且每个事件设置将被恢复。如果纠正源文件时造成以前使用的行或符号被删除，将会产生如下情况：

- 变量显示时是暗淡色。
- 时间状态的事件标志是以黄色显示。
- 软件断点也许被删除。

- 5) 在PM+中选择【Build】菜单→【Build and Debug】或【Build】菜单→【Rebuild and Debug】。

4.4.2 通过启动调试器来实现自动加载

如果在PM+中执行下面操作（ID78K0-QB已经启动），加载模块将被自动地下载到中。

-在PM+中选择【Build】菜单→【Build and Debug】。

-在PM+中选择【Build】菜单→【Rebuild and Debug】。

备注：要指出从[Debugger Settings...]（在PM+[Tool]菜单中）下载之后是否使用CPU重新设置（CPU重新设置是在默认情况下进行设置的）。

第5章 调试功能

本章介绍的是ID78K0-QB调试功能。

表5-1 调试功能列表（调试操作流）

条目	章节
设置调试环境	5.1设置调试环境
下载加载模块	5.2下载功能，上传功能
显示源文件和分离结果	5.3源代码显示，反汇编显示功能
设置断点	5.4中断功能
执行用户程序	5.5程序执行功能
检查变量值	5.6查看功能
检查和编辑存储器内容	5.7存储器操控功能
检查和改变寄存器变量	5.8寄存器操控功能
检查执行时间	5.9计算器功能（当连接IECUBE时）
检查跟踪数据	5.10跟踪功能（当连接IECUBE时）
检查代码覆盖测试结果	5.11覆盖测试功能（当连接IECUBE时）
管理事件	5.12事件
截图功能	5.13截图功能（当连接IECUBE时）
存根功能	5.14存根功能（当连接IECUBE时）
RAM取样	5.15RRM功能
DMM功能	5.16DMM功能
保存调试环境和窗口状态	5.17加载/保存功能
跳转功能，链接窗口和注意事项	5.18每个窗口共同的功能

5.1 设置调试环境

本部分将介绍关于设置调试环境的几项说明：

- 设置操作环境
- 设置选项
- 设置映射

5.1.1 设置操作环境

仿真器操作环境设置在[配置对话框](#)中实现，ID78K0-QB启动时配置对话框将自动显示。

如果工程文件已经存在，点击<Project...>按钮就可以恢复调试环境。（参见“[5.17.1 调试环境\(工程文件\)](#)”）。

5.1.2 设置选项

在下面的设置对话框中进行与调试器或仿真器相关的设置。

- 配置对话框
- 扩展的选择对话框
- 失效保护断点对话框
- RRM对话框
- 记号选择对话框
- 调试器对话框
- 伪仿真对话框

5.1.3 设置映射

在[配置对话框](#)中进行映射设置。
将会用到下面几种映射属性类型：

表5-2 映射属性

属性	含义
内部ROM	内部RAM（当连接IECUBE时） 被指定为内部ROM的存储区域相当于目标设备的内部ROM（核心）。如果目标设备试图在此存储区中写入信息，将会产生书写保护中断。
内部高速的RAM	被指定为内部高速RAM的存储区域相当于目标设备的内部高速RAM（核心）。 实际存储配置要依赖于目标系统
内部扩展的RAM	被指定为内部扩展RAM的存储区域相当于目标设备的内部扩展RAM（核心）。 实际存储配置要依赖于目标系统
目标	用户区映射 被指定为用户区的存储区域将是变成这样一个区域：访问目标系统的存储器或并入到CPU中的存储器。
I/O保护	I/O保护区 I/O保护区可以在指定的“目标”区域中设置。 I/O保护区将显示在 存储窗口 中，其显示方式和没有映射的区域相同（显示符号：??）。通过对具有此属性的区域进行映射，在 存储窗口 中就不可能从（或向）此区域读取（或写入）数据。这样此区域中的数据就可以避免非法访问。 为了从具有此属性的区域读取或写入值，需要将这个值注册到 SFR窗口 或 查看窗口 中。
堆栈	只有内部高速的RAM区域才能被设置为堆栈区（内部扩展RAM区域不能被设置为堆栈区）（当连接IECUBE时）。 注意：如果堆栈区没有被指定，那么在默认情况下，整个内部高速RAM区是被作为堆栈区。

5.2 下载功能，上传功能

ID78K0-QB允许以下面列表中的格式来下载和上传目标文件：表5-3，表5-4。

此部分将介绍：

-下载

-上传

备注：当连接MINICUBE时，就可以在内部闪存中写入信息，并可以下载加载模块（参见“5.7.4 闪存写入功能（当连接MINICUBE 时）”）。

5.2.1 下载

目标文件下载在下载对话框中。

通过下载具有调试信息加载模块显示相应的源文本文件（源窗口）。

(1) 能被下载的文件格式

能被下载的文件格式如下：

表5-3 能被下载的文件类型

格式	扩展
加载模块（XCOFF（*lnk:*lmf））	加载模块（*lnk:*lmf）
Intel Hex格式（标准和扩展）	Hex 格式(*.hex;*.hxb;*.hxf) ^{备注1} Hex 格式 [Bank](*.hex;*.hxb;*.hxf) ^{备注1}
Motorola Hex格式S类型（S0, S2, S3, S5, S7, S8记录）	Hex 格式 [64KB] (*.hex;*.hxb;*.hxf) ^{备注1}
扩展的Tektronix Hex格式	
二进制数据	二进制数据(*.bin) ^{备注2} 二进制数据[Bank](*.bin) ^{备注2} 二进制数据[64KB](*.bin) ^{备注2}
覆盖结果（当连接IECUBE时）	覆盖(*.cvb)

备注1：包括与闪存自编程模式相兼容的Hex格式。

Hex文件格式可以自动终止。

Hex格式(*.hex;*.hxb;*.hxf)：当不使用存储bank时。

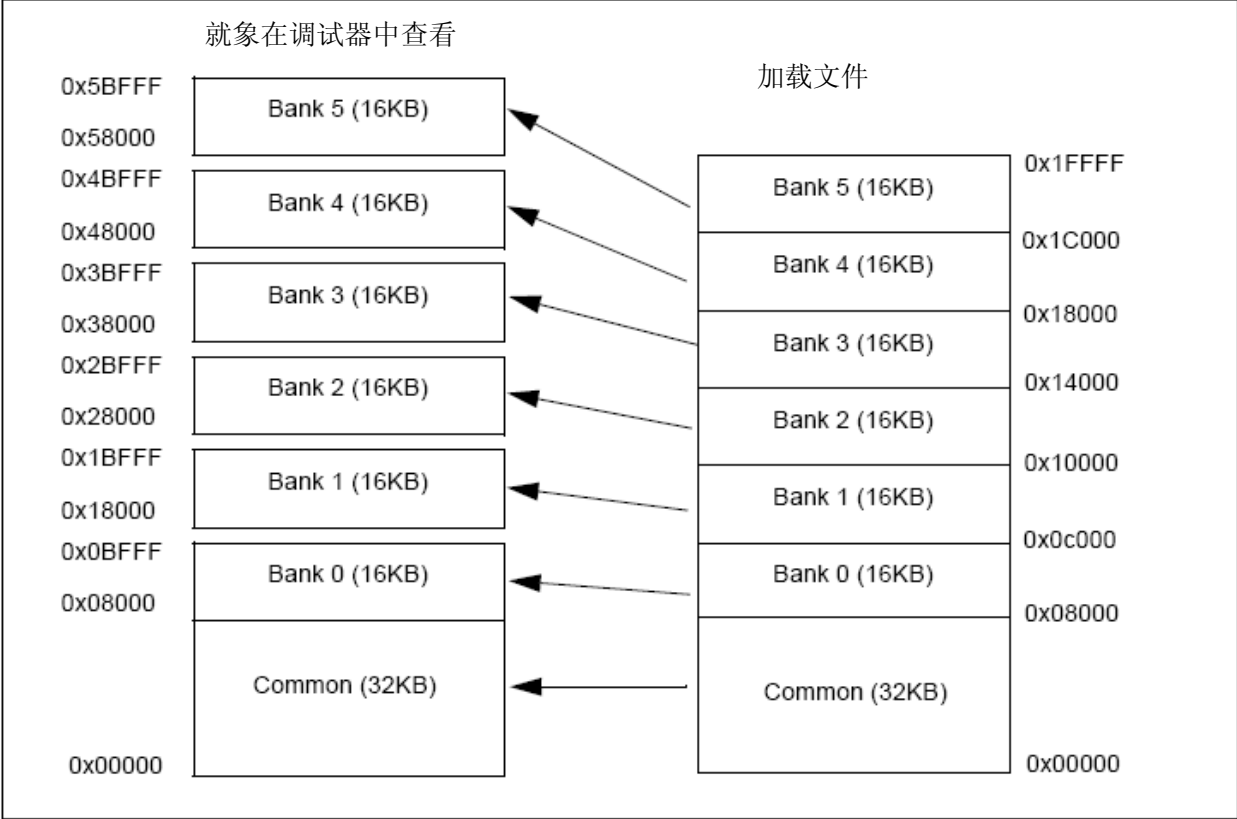
Hex格式 [Bank] (*.hex;*.hxb;*.hxf)：当使用存储bank时（为存储bank）。

Hex格式[64KB](*.hex;*.hxb;*.hxf)：当使用存储bank时（当存储容量是低于64kB时）。

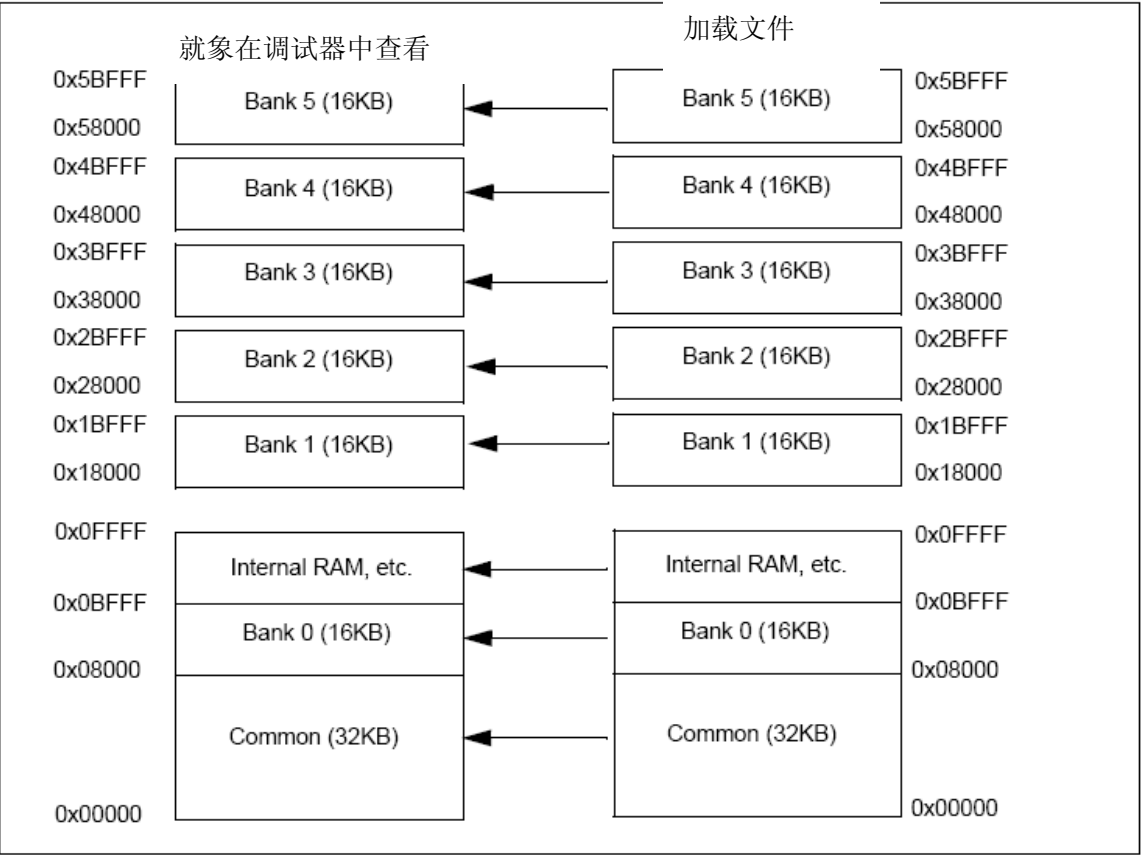
备注2: 二进制数据(*.bin): 当不使用存储bank时。
二进制数据[Bank](*.bin): 当使用存储bank时（为存储bank）。
二进制数据[64KB](*.bin): 当使用存储bank时（当存储容量是低于64kB时）。

(2) 当使用存储bank时的分配图
下面这个图显示的是当下载Hex格式或二进制格式文件的分配图（当使用存储bank时）。

【当指定“Hex格式[64KB] (*.hex)”或 “二进制数据(*.bin)”时】



【当指定“Hex格式 [Bank] (*.hex)”或 “二进制数据 [Bank] (*.bin)”时】



5.2.2 上传

在上传对话框中上传存储器中的内容。设置保存范围。

(1) 能被上传的文件格式

能被上传的文件格式如下：

表5-4 能被上传的文件类型

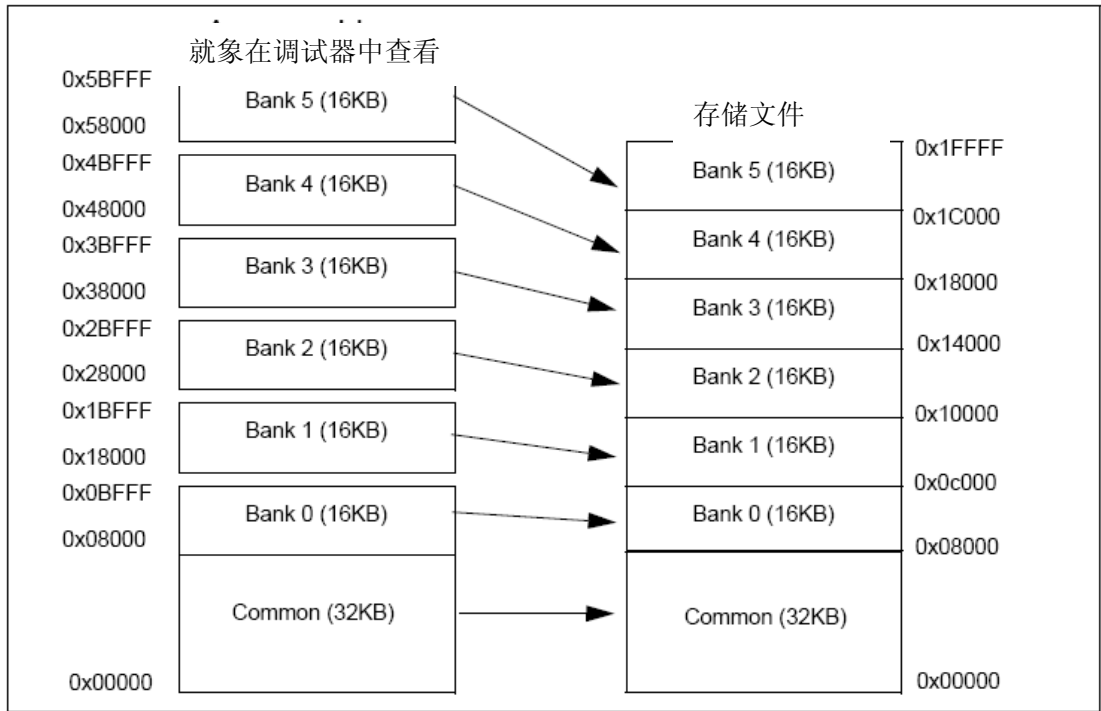
扩展	格式
Intel Hex (*.hex)	Intel Hex格式扩展（20位地址）
Motorola Hex (*.hex)	Motorola Hex格式S类型（S0,S2,S8 -24位地址）
Tektro Hex (*.hex)	扩展的Tektro Hex格式
二进制数据(*.bin)	二进制数据
覆盖(*.cvb)	覆盖结果（当连接IECUBE时）

备注：当使用存储bank时，如果上传覆盖结果，所有的存储区域（下图中阴暗的部分）都将被保存（当内部bank ROM大小是48KB时）（当连接IECUBE时）。

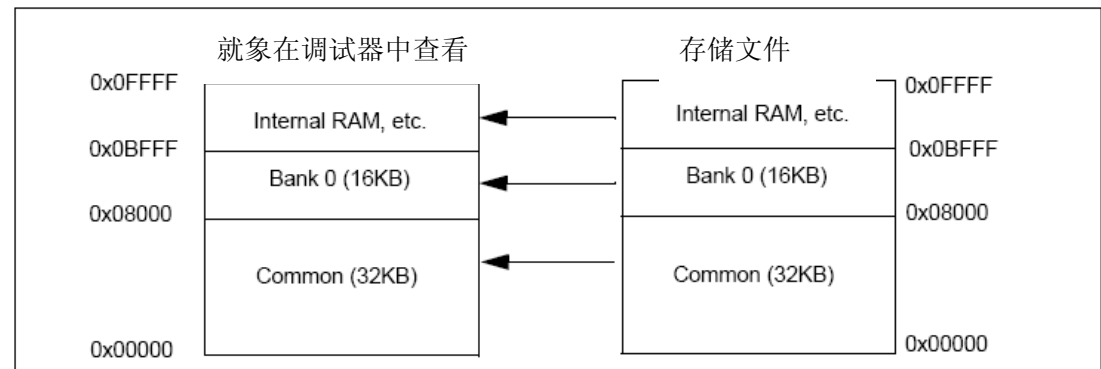
0x2FFFF	
0x2BFFF	
0x28000	Bank 2 (16KB)
0x1BFFF	
0x18000	Bank 1 (16KB)
0x0FFFF	
0x0BFFF	
0x08000	Bank 0 (16KB)
0x00000	Common (32KB)

(2) 当使用存储bank时的存储图

下面这个图显示的是当上传Hex格式或二进制格式文件的存储图（当使用存储bank时）。存储结果是有区别的，这依赖于高于0x10000的地址或低于0x10000的地址是否是指定为存储地址范围的末地址，如下图。
【当高于0x10000的地址被指定为末地址时】



【当低于0x10000的地址被指定为末地址时】



5.3 源代码显示，反汇编显示功能

在源窗口中执行源文件显示。在汇编窗口中执行反汇编显示和行汇编。

本部分将介绍：

- 源代码显示
- 反汇编显示
- 混合显示模式（源窗口）
- 转换符号（将符号转地址）

注释：在用户程序中执行的覆盖测试位置将显示在源窗口和汇编窗口中（参见“5.11.3 执行的覆盖测试位置显示”）。

5.3.1 源代码显示

通过下载具有调试信息的加载模块文件将相应的文本文件显示在源窗口中。在源文本移动对话框中的显示开始位置可能会改变（通过选择[View]菜单→[Move...]来显示）。

与标签大小、显示字体等规范说明和源路径规范说明可以在调试器选项对话框中进行设置。在源搜索对话框（点击<Search...>按钮打开）中指定搜索方法。搜索结果将在源窗口中突出显示。

表5-5 显示的文件类型

文件类型（扩展名）	含义
源代码(*.c, *.s, *.asm)	源文件 （扩展名可以在调试器选项对话框中进行改变。）
文本(*.txt)	文本文件
所有(*.*)	所有文件

5.3.2 反汇编显示

在汇编窗口中进行反汇编显示。

在地址移动对话框（通过选择[View]菜单→[Move...]打开）中可以改变显示的开始位置。

在调试器选项对话框中指定偏移量显示和寄存器名显示。

在汇编搜索对话框（点击<Search...>按钮打开）中指定搜索方法。搜索结果将在汇编窗口中突出显示。

5.3.3 混合显示模式（源窗口）

在源窗口中选择[View]菜单→[Mix]，可以将程序分开并和源文件一起显示。显示在混合显示模式中的内容可以保存在一个查看文件中。

正常显示模式

		58	/* Timer Set */
*		59	TUM1 = 0x200;
*		60	CE1 = 1;
*		61	time_over = 0;

在正常显示模式下，可以显示总的文本文件和源文件。

混合显示模式

		58	/* Timer Set */
*		59	TUM1 = 0x200;
*	00000394	20660002	movea 0x200, r0, r12
*	00000398	606740F2	st.h r12, TUM1
*	60	CE1 = 1;	
*	0000039C	c03f42F2	set1 0x7, TMC1
*	61	time_over = 0;	
*	000003A0	440e0000	movhi 0x0, gp, r1
*	000003A4	61071184	st.w r0, -0x7bf0[r1]

如果程序代码与显示的源文件各行对应，则反汇编行将紧接着源代码行显示。显示地址、代码数据和反汇编助记符标签（助记符显示的开始位置可以通过设置标签大小值来进行调节）。

注意：只有当下载加载模块、符号信息被读取时混合显示模式才是有效的。并且显示相应的源文件。

备注：当在混合显示模式中使用光标键进行回滚时，可能造成回滚过多的情况。使用光标键不可能回滚到最后一行。

5.3.4 转换符号（将符号转地址）

在符号转换到地址对话框中，可以显示指定变量或函数的地址，或指定符号的值。
通过在源窗口中或汇编窗口中选择要转换的字符串来执行符号转换。然后选择上下文菜单→[Symbol...]。
下面表格中列出规范说明的符号。

表5-6指定符号

变换目标	规范说明方法
变量	var file#var (指定具有文件名的静态变量) func#var (指定具有函数名的静态变量) file#func#var (指定具有文件名和函数名的静态变量)
函数	func file#func (指定具有文件名的静态函数)
标签	label file#label (指定具有文件名的本地标签)
EQU符号	equsym file#equsym (指定具有文件名的本地EQU符号)
位符号	bitsym file#bitsym (指定具有文件名的本地位符号)
源文件行数	file#no prog\$file#no
I/O端口名	portname
SFR名	sfrname
寄存器名	regname
PSW标记名	pswname

备注：分隔符“#”

“#”是作为文件名、变量、函数名和行号的分隔符。如果指定的符号在某一范围内没有发现，将会搜索所有符号（静态变量，静态函数，本地标签）。

5.4 中断功能

中断功能是用来停止用户程序的执行（通过CPU和跟踪器操作）。

本部分将介绍：

- 中断类型
- 设置断点为变量
- 硬件中断和软件中断
- 失效保护断点

5.4.1 中断类型

ID78K0-QB有以下几个中断功能。

表5-7中断类型

条目	内容
硬件中断 ^{备注1} （事件探测中断）	功能：在设置中断事件状态的探测中终止用户程序的执行。 →参见“5.4.2断点设置”
软件中断 ^{备注1}	功能：取代指定的地址软件中断指令，终止执行的用户程序（参见“5.4.4硬件中断和软件中断”）。 →参见“5.4.2断点设置”
【Come Here】中断 ^{备注2}	功能：在设置中断事件状态的探测中通过选择【Run】菜单→【Come Here】来终止用户程序的执行（在源窗口和汇编窗口中）。
安装时单步执行时的中断	功能：在每个命令（[Step In], [Next Over], [Return Out], [Slowmotion]）终止的状态下终止执行，
强迫中断	功能：通过选择【Run】菜单→【Stop】或来STOP按钮来强制终止执行。 它对所有执行命令都有效。
失效保护断点（当连接IECUBE时）	功能：当用户程序执行与存储器或寄存器相关的非法操作时强迫终止执行（参见“5.4.5 失效保护断点”）。 →参见“失效保护断点对话框”
超时中断（当连接IECUBE时）	功能：当测试事件超过指定的事件时终止用户程序的执行（参见“定时器对话框”）。

备注1：此中断对[Go]、[Go & GO]、[Come Here]和[Restart]是有效的。

备注2：终止用户程序执行之后，此功能的断点将被删除。在执行用户程序期间，在光标位置之前设置的中断事件将不会发生。

5.4.2 断点设置

通过点击源窗口或汇编窗口可以将断点设置到需要的位置中。

因为断点是被设置作为中断事件状态并使用事件功能进行管理，所以限制条件也可以应用到被设置的断点个数（参见“5.12.4 每个事件状态中有效事件的个数”）。

注意：当MINICUBE连接时，在执行用户程序时不可以设置或删除断点。

(1) 断点设置方法

通过点击显示“*”的行来执行断点（存在程序代码的行）。

在默认设置情况下，可以设置软件断点（B），但是如果在上下文菜单中选择【断点】，就要设置硬件断点（B或B）。

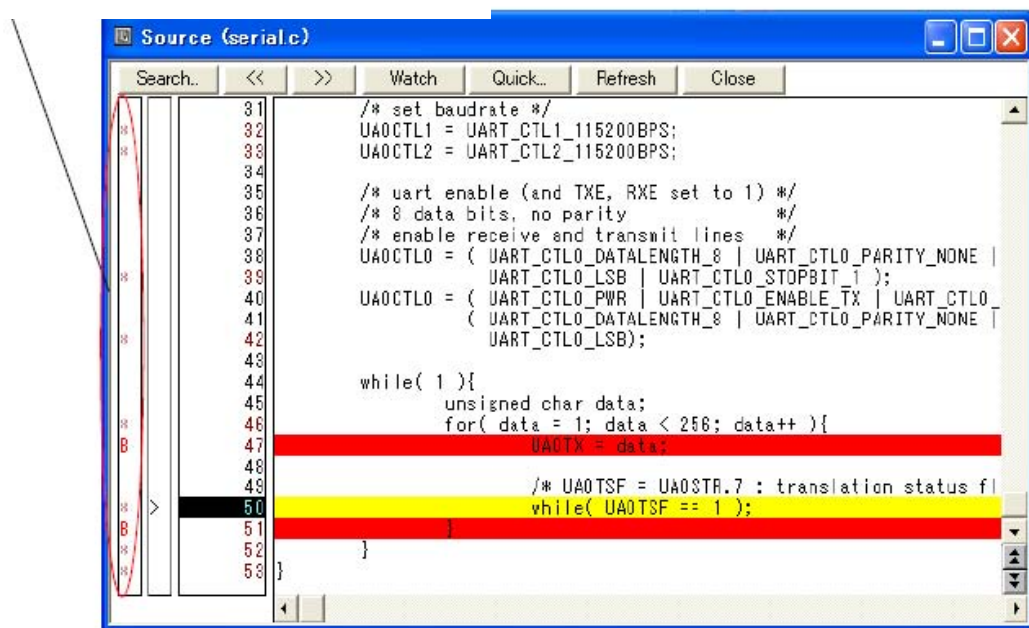
如果在已经设置事件断点的行中再设置断点，将会标注“A”，表示多个事件被设置（参见“标记 6-7 事件设置状态（事件标记）”）。

注意：软件断点不能在外部映射的ROM区域中设置或删除。

备注：默认情况下中断设置可以在扩展的选项对话框中进行改变。

图 5-1 断点设置

在此区域中点击星号（*；程序代码）



(2) 删除断点方法

在被设置断点的位置上点击，此断点将要被删除。

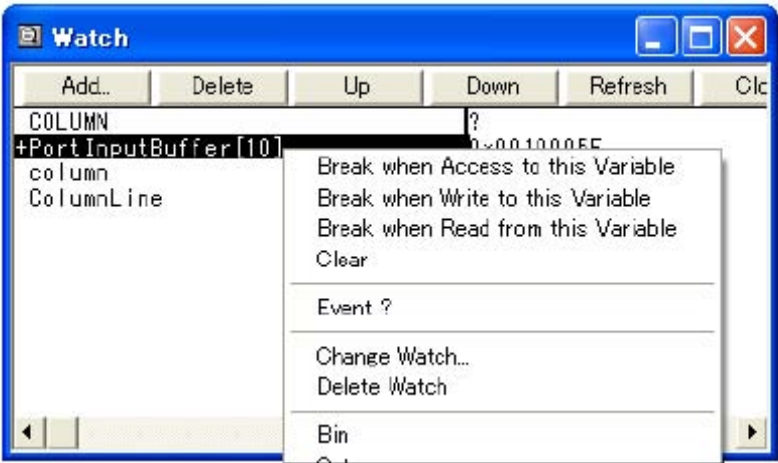
在执行设置的同时，在默认情况下，删除软件断点（B），但是如果在上下文菜单中选择[Breakpoint]，就要删除硬件断点（B或B）。

删除之后，如果另一个事件依然存在，将显示此事件的标记符号。

5.4.3 将断点设置为变量

在源窗口或查看窗口中利用上下文菜单可以很容易将访问断点设置成变量。

图 5-2 将断点设置为变量



5.4.4 硬件中断和软件中断

(1) 硬件中断

硬件中断是指对于每种事件状态下使用硬件资源来设置的中断。
因此，在ID78K0-QB, 中，他们可以使用“5.12事件功能”作为中断事件状态进行管理。
有效断点的数量是根据设备的多少而变化的（参见“5.12.4每个事件状态下有效事件的数量”）。

(2) 软件中断

软件中断：通过将指定地址的指令重新写入到软件中断指令中而设置的中断。对于外部ROM设置，不能指定可变的访问时间，来实现终止操作。
注意：如果为了重新执行开始地址中的代码而使软件中断设置为执行开始地址，那么下面被设置为地址的这些事件就不会产生（当连接IECUBE时）。

- 1) 开始部分跟踪
- 2) 开始部分测试
- 3) 跟踪延时触发
- 4) 执行之后的事件
- 5) 访问事件

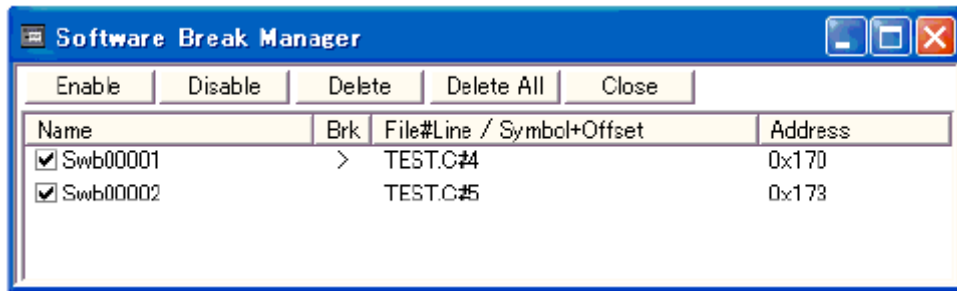
为了重新执行开始地址中代码，那么就需要在执行之前使用事件中断。

表5-8有效软件中断的数量

连接的IE	有效数
IECUBE	2000
MINICUBE	

通过[软件中断管理器](#)来管理软件软件中断。

图5-3软件中断的管理



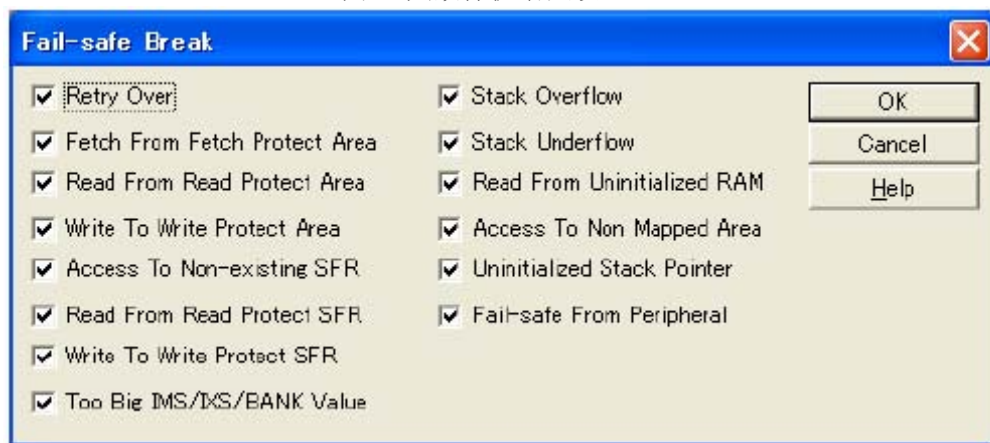
5.4.5 失效保护断点

失效保护断点功能：当用户程序执行与存储器或寄存器相关的非法操作时强制终止执行。

失效保护断点设置是在[失效保护断点对话框](#)中完成的。

在这里也可以选择检查对话框进行个人设置。

图5-4失效保护断点设置



5.5 程序执行功能

程序执行功能：通过CPU和跟踪器的操作来开始/终止用户程序的执行。

在执行用户程序时，如果没有产生设置中断或强制中断程序计算器是（PC）一直在增加的。（参见“5.4中断功能”）

备注：当执行用户程序时，可以对跟踪事件状态和定时器事件状态进行设置。（参见“跟踪对话框”，“定时器对话框”）

以下提供的是ID78K0-QB执行功能的类型。他们是通过使用工具条上的执行按钮或从【Run】菜单进行操作。

图 5-5 执行按钮



图 5-6 【Run】菜单

File	Edit	View	Option	Run	Event	Browse	Jump	Simulator	Window	Help
					Restart	F4				
					Stop	F2				
					Go	F5				
					Ignore break points and Go	Ctrl+F5				
					Return Out	F7				
					Step In	F8				
					Next Over	F10				
					Start From Here	Shift+F6				
					Come Here	F6				
					Go & Go					
					Slowmotion					

表5-9执行类型

条目	内容
[Restart] 	重新设置CPU，从地址RESET中来执行用户程序。此操作和“在执行用户程序和执行[Go]之前来执行[CPU Reset]”相同。
[Go] 	从当前PC寄存器指定的地址开始来执行用户程序。直到建立中断状态（参见“5.4.1 中断类型”）才终止执行。
[Ignore break points and Go] 	从当前PC寄存器指定的地址开始来执行用户程序。用户程序会一直执行，而不考虑设置的断点。
[Return Out] 	用户程序一直执行，直到执行返回到C语言中的调用函数才终止。
[Step In] 	在源代码模式，源文件的单步执行是从当前PC寄存器值开始的并且每个窗口的内容都会被更新。在指令模式，单条指令的执行是从当前PC寄存器值开始的并且每个窗口的内容都会被更新。
[Next Over] 	CALL / CALLT / CALLF 指令 执行下一步，假设由CALL / CALLT / CALLF指令调用的函数或子程序是单步执行的（直到嵌套级和CALL / CALLT / CALLF执行时相同才终止单步执行。）。 非CALL / CALLT / CALLF 指令 处理方式同[Step In]。
[Start From Here]	从命令是从指定的地址开始来执行用户程序。当满足一个设置中断事件状态时，用户程序将立即终止执行。
[Come Here]	用户程序是从当前PC寄存器指定的地址处开始执行的，终止地址是从源窗口或汇编窗口的line/address显示区域中选择的地址，然后就会产生一个中断。在用户程序执行期间，不能对当前中断事件进行设置。
[Go & Go]	用户程序是从当前PC寄存器指定的地址开始执行的，如果满足一个设置中断事件状态时，用户程序将立即终止执行。每个窗口中的内容都会被更新，用户程序会从程序终止处重新开始执行。此操作会反复执行，直到用户执行[Stop]操作才终止。
[Slowmotion] 	每行的单步执行是从当前PC寄存器中指定的地址开始执行的（在源代码模式）。在指令模式下，每条指令的单步执行都会被执行到。每当进行单步执行时，每个窗口中的内容都会被更新。
[CPU Reset] 	重新设置CPU。
[Stop] 	强制终止执行的程序。

5.6 查看功能

本部分将介绍与查看功能有关的内容：

- 显示和改变数据值
- 显示和改变本地可变值
- 注册和删除查看数据
- 改变查看数据
- 临时显示和改变数据值
- 回调查看功能
- 堆栈跟踪显示功能

5.6.1 显示和改变数据值

在查看窗口中显示和改变数据值。通过注册查看数据来检查改变的数据值。
在调试器选项对话框中指定显示格式。

图5-7查看窗口

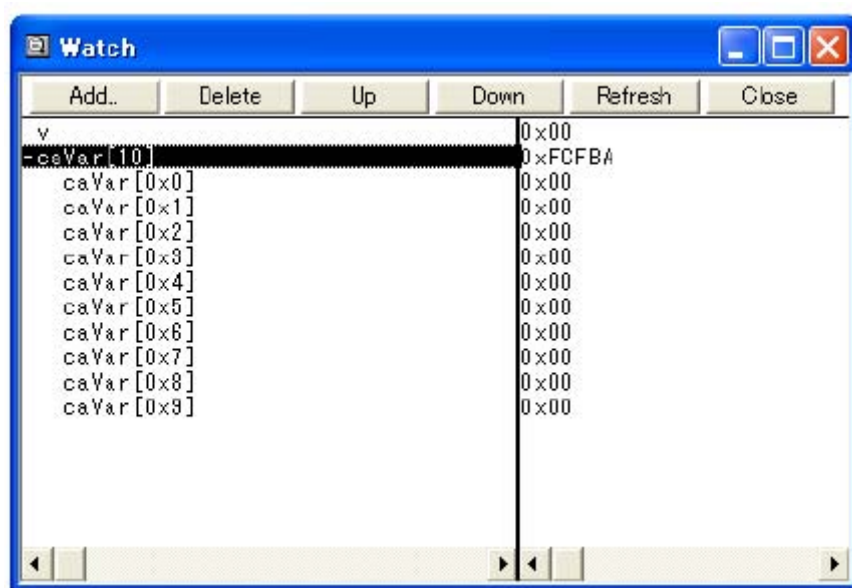
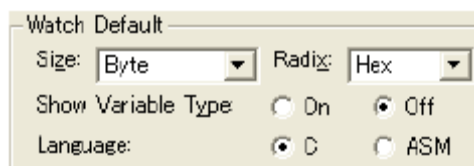


图 5-8 显示格式的规格说明（调试器选项对话框）

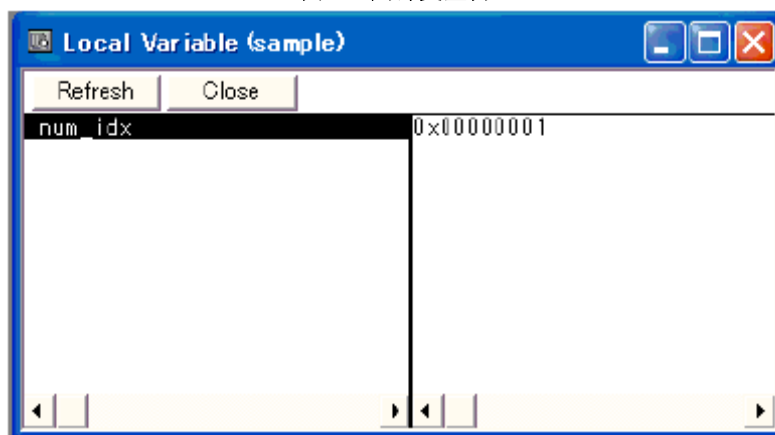


5.6.2 显示和改变本地可变量

在[局部变量窗口](#)中显示和改变局部变量。

在此功能中，局部变量可以自动地显示在这个窗口中。（不能增加/删除变量。）

图 5-9 局部变量窗口



5.6.3 注册和删除查看数据

从[源窗口](#)或[汇编窗口](#)中可以将数据注册到[查看窗口](#)中。操作很简单：在相应的窗口中选择变量或符号名，然后点击<Watch>按钮即可。

也可以使用下面这个方法注册。

- 直接在查看窗口中拖放选择的变量或符号名。（参见“[5.18.4 拖放功能](#)”）。

- 在[快速查看窗口](#)或[增加窗口对话框](#)中点击<Add>按钮。

为了删除查看数据，您需要点击变量名或符号名（使用Shift键或Ctrl键也可以进行多个选择），然后点击<Delete>按钮。但是不能删除可以延伸层的行（比如，数组元素，结构体和共用体成员）。

5.6.4 改变查看数据

在[改变查看对话框](#)中改变查看数据。

注意：即使具有数据的符号名在使用过程中已经产生了复本，也可以对其进行改变。

图5-10改变查看对话框



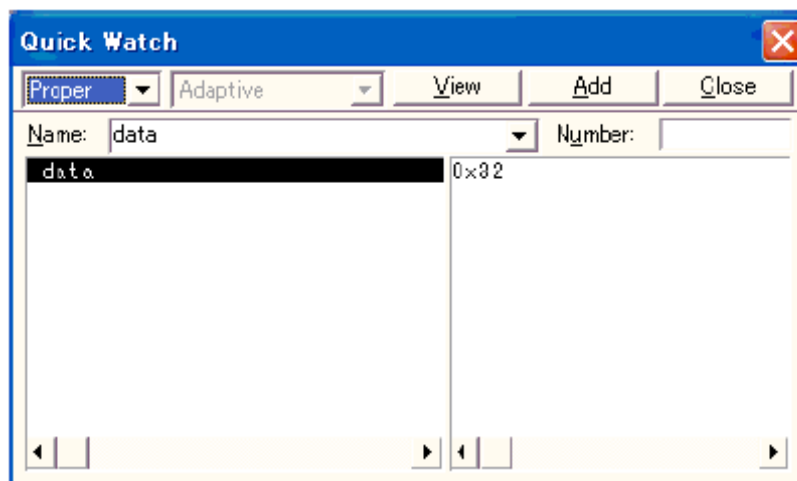
5.6.5 临时显示和改变数据值

在[快速查看对话框](#)中临时的显示和改变数据值。

在源窗口或汇编窗口中选择需要的变量或符号名，然后点击<Quick...>按钮来执行查看注册的数据。

在此窗口中可以改变显示的基数、大小和数量。

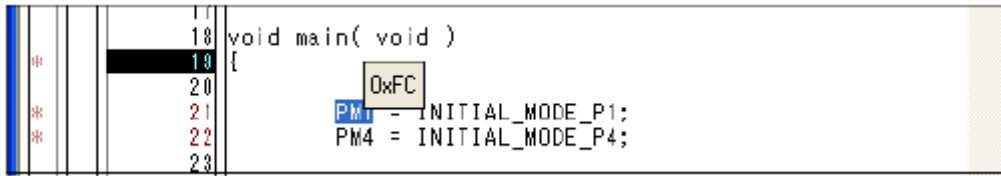
图5-11快速查看对话框



5.6.6 回调查看功能

在源窗口或汇编窗口中将鼠标放置选择的变量时，就会弹出变量相应的值。

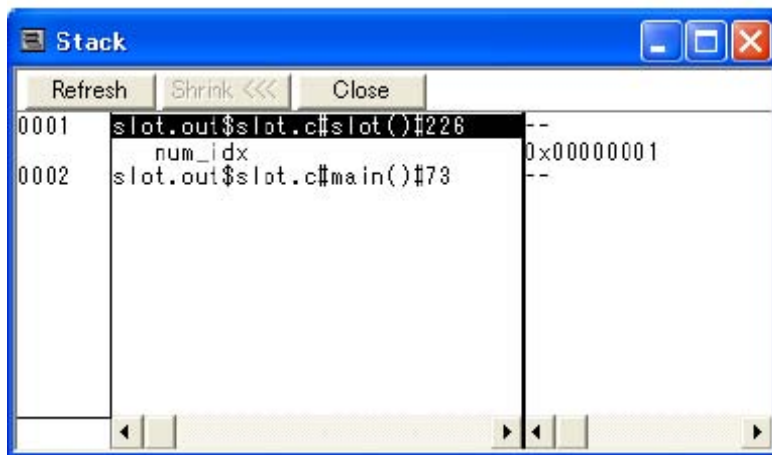
图5-12回调查看功能



5.6.7 堆栈跟踪显示功能

在堆栈窗口中显示当前用户程序堆栈中的内容。

图5-13堆栈窗口



5.7 存储器操控功能

本部分将介绍与存储器操控相关的内容：

在[扩展选项对话框](#)中指定检验等操作。

-显示和改变存储器内容

-填充、复制和比较存储器内容

-访问监控器功能（当连接IECUBE时）

-闪存写入功能（当连接MINICUBE时）

5.7.1 显示和改变存储器内容

在[存储窗口](#)中，使用记忆码、十六进制码和ASCII码来显示或改变存储器内容。搜索操作是在[存储器搜索窗口](#)中完成的。通过点击<Search...>按钮来显示[存储器搜索窗口](#)。搜索的结果将在[存储窗口](#)中突出显示。

可以在[地址移动对话框](#)中改变显示开始位置。此对话框可以通过选择[View] 菜单→[Move...]来显示。

分配给采样范围内的变量和数据可以实时的显示，即使是在程序执行期间（参见“[5.15 RRM功能](#)”）。

备注：当使用存储bank时改变地址宽度。

5.7.2 填充、复制和比较存储器内容

在[存储器填充对话框](#)、[存储器复制对话框](#)和[存储器比较对话框](#)中对存储内容进行填充、复制和比较。这些对话框通过选择[Edit]菜单→[Memory] → [Fill.../Copy.../Compare...]来显示。

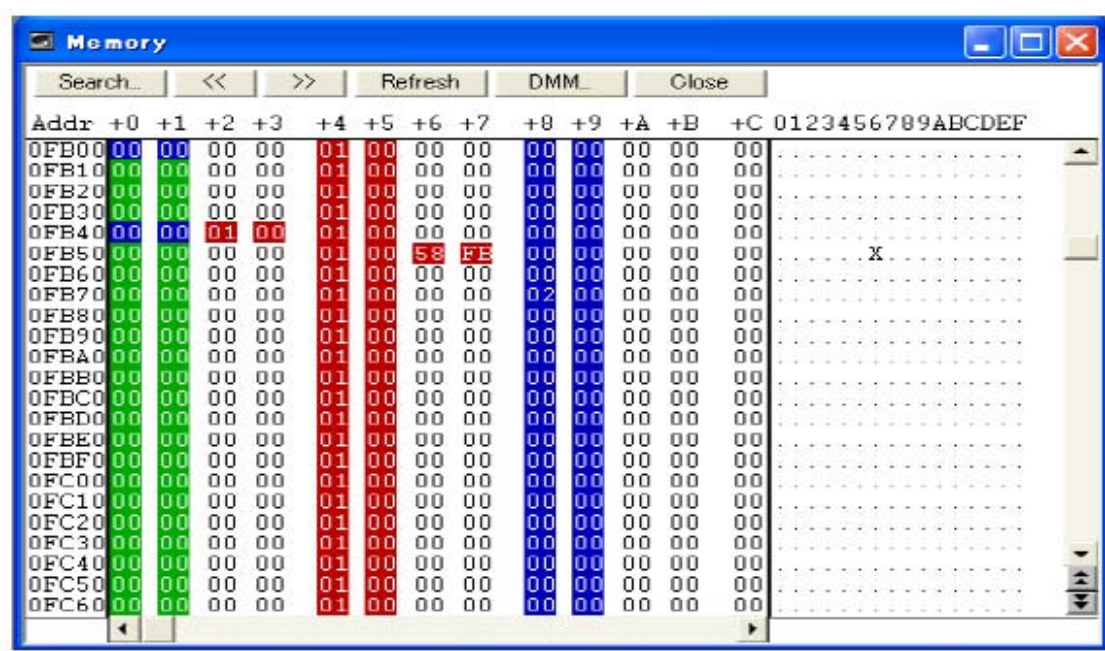
比较的结果将显示在[存储器比较结果对话框](#)中。

备注：当使用存储bank时改变地址宽度。

5.7.3 访问监控器功能（当连接 IECUBE 时）

在存储窗口中使用不同色彩，访问监控器功能可以显示RRM功能采样范围的访问状态（读，些，读和写）。只有在显示单位中选择“Byte”，访问监控查看功能才可用。
色彩不会显示在ASCII显示区域中。通过选择[View]菜单→[Access Monitoring]来清除累积显示设置和访问状态。
注意：在程序执行期间通过DMA重新写入到存储器中的值和从ID78K0-QB重新写入到存储器中的值不能在访问监控器中显示。
备注：从内部ROM区域中获取的数据可以象读取信息一样显示。

图5-14访问监控器功能（存储窗口）



5.7.4 闪存写入功能（当连接 MINICUBE 时）

在ID78K0-QB中，可以象访问普通存储器操作的方法向内部闪存写入数据、下面装置模块。
可以在存储窗口、汇编窗口、查看窗口、存储器填充对话框和存储器复制对话框中改变内部闪存中的数据，而且不需要知道是否是内部闪存中的数据。通过使用闪存自我编程功能也可以将加载模块下载到内部闪存当中（参见“闪存选项对话框”）。
注意：在执行用户程序期间不能将数据写入到内部闪存当中。

5.8 寄存器操控功能

本部分将介绍与寄存器操控功能相关的内容：

- 显示和改变寄存器内容
- 显示和改变SFR内容
- 显示和改变I/O端口内容

5.8.1 显示和改变寄存器内容

可以在寄存器窗口中显示和改变寄存器内容。
在调试器选项对话框中来转换寄存器名的显示（全名/函数名）。

图5-15全名/函数名转换

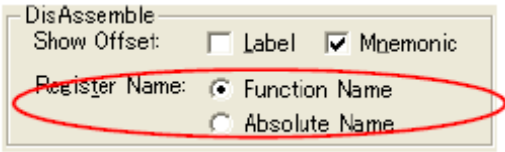


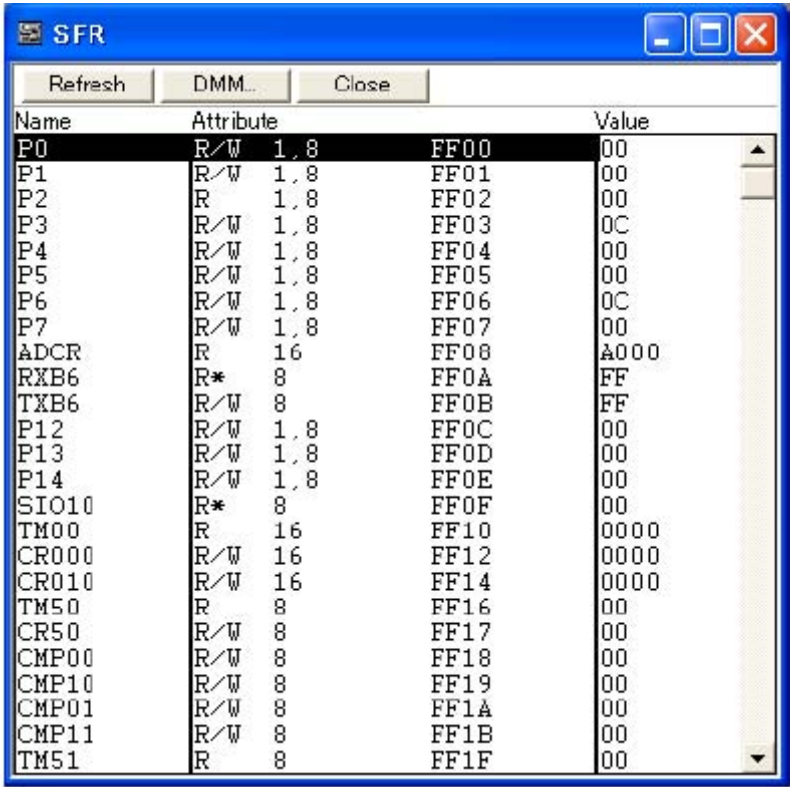
表5-10全名和函数名对应比较

函数名		全名	
双寄存器	寄存器	双寄存器	寄存器
ax	x	rp0	r0
	a		r1
bc	c	rp1	r2
	b		r3
de	e	rp2	r4
	d		r5
hl	l	rp3	r6
	h		r7

5.8.2 显示和改变 SFR 内容

可以在SFR窗口中显示和改SFR内容。
在地址移动对话框中来改变显示的开始位置，此对话框通过选择[View]菜单→[Move...]来显示。
在SFR选择对话框中选择显示的寄存器。

图5-16显示SFR内容



The SFR window displays a table of registers. The table has three columns: Name, Attribute, and Value. The registers listed are P0 through TM51. The values are shown in hexadecimal.

Name	Attribute	Value
P0	R/W 1, 8	FF00
P1	R/W 1, 8	FF01
P2	R 1, 8	FF02
P3	R/W 1, 8	FF03
P4	R/W 1, 8	FF04
P5	R/W 1, 8	FF05
P6	R/W 1, 8	FF06
P7	R/W 1, 8	FF07
ADCR	R 16	FF08
RXB6	R* 8	FF0A
TXB6	R/W 8	FF0B
P12	R/W 1, 8	FF0C
P13	R/W 1, 8	FF0D
P14	R/W 1, 8	FF0E
SIO10	R* 8	FF0F
TM00	R 16	FF10
CR000	R/W 16	FF12
CR010	R/W 16	FF14
TM50	R 8	FF16
CR50	R/W 8	FF17
CMP00	R/W 8	FF18
CMP10	R/W 8	FF19
CMP01	R/W 8	FF1A
CMP11	R/W 8	FF1B
TM51	R 8	FF1F

5.8.3 显示和改变 I/O 端口内容

一旦用户定义的I/O端口被注册到增加I/O端口对话框中，那么就可以在SFR窗口中对其进行显示和改变。
如果产品支持可编程I/O寄存器，那么就可以设置配置对话框中的可编程I/O区域来显示和改变可编程I/O寄存器中内容。

图5-17注册I/O端口



The Add I/O Port dialog box contains the following fields and controls:

- I/O Port List:** A list box for registered ports.
- Name:** A text input field.
- Address:** A text input field.
- Access:** Radio buttons for ☒ Byte and ☐ Word.
- Read / Write:** Checkboxes for ☐ Read Only, ☐ Write Only, and ☐ Read Protect.
- Buttons:** OK, Cancel, Restore, Help, Add, Change, and Delete.

5.9 计时器功能（当连接 IECUBE 时）

计时器功能可以测量从开始执行用户程序到中断时所用的时间，或使用计时器事件测量在特定用户程序间隔内的执行时间。Run-Break时间也将显示在主窗口的状态条中。

本部分将介绍：

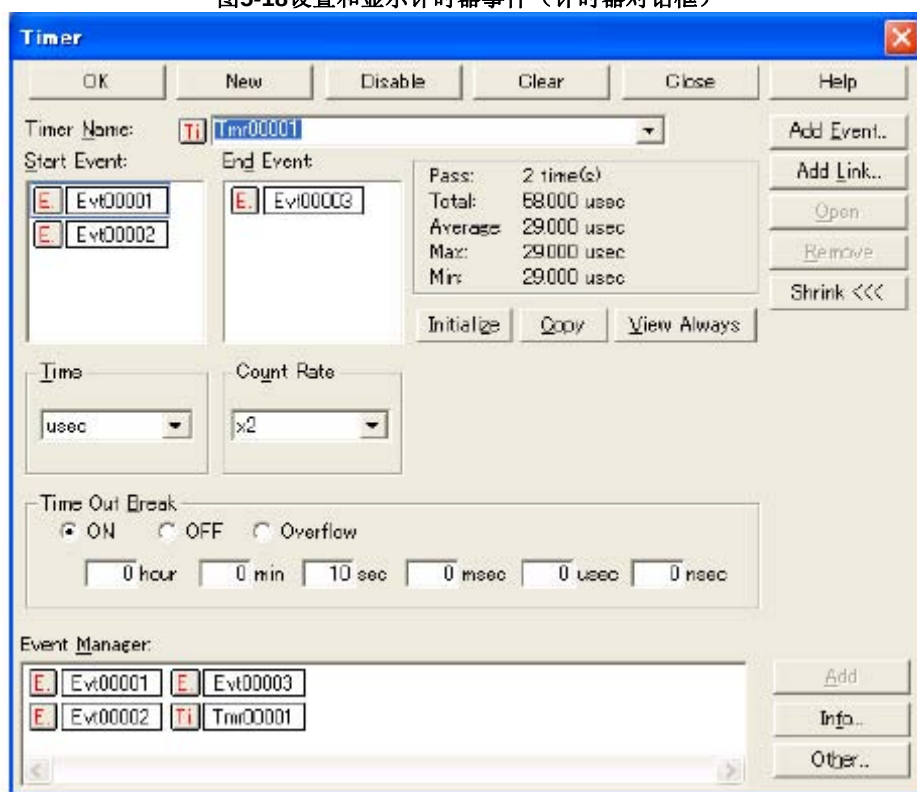
-计时器时间状态

-Run-Break事件

5.9.1 计时器事件状态

计时器事件状态指定触发器，来触发时间测量的开始或终止。计时器事件状态在计时器对话框中设置（参见“5.12事件功能”）。在ID78K0-QB中，在Time Out Break区域进行超时中断设置。

图5-18设置和显示计时器事件（计时器对话框）



通过点击<View Always>按钮可以在计时器结果对话框中连续显示。

选择[Run]→[Timer Start/Timer Stop]可以在程序执行期间操控计时器。

5.9.2 Run-Break 事件

Run-Break事件是计时器事件名（具有一个计时器事件状态），测量的是从开始执行到中断的时间。Run-Break可以预先注册，并且Run-Break时间可以通过规格说明显示在[计时器对话框](#)中。

Run-Break时间也可以显示在主窗口的状态条中。

因为Run-Break事件包括在能够同时有效的计时器事件数当中（参见“[5.12.4 每个事件状态中有效事件的数量](#)”），所以它们可以增加到有效计时器事件状态中进行使用。

5.10 跟踪功能（当使用 IECUBE 时）

跟踪功能可以用来将数据的历史记录（指出用户程序的执行过程）保存到跟踪存储器中。本部分将介绍：

- 跟踪存储器
- 检查跟踪数据
- 混合显示模式（跟踪查看窗口）
- 跟踪器操作
- 设置状态跟踪

5.10.1 跟踪存储器

ID78K0-QB具有环结构的跟踪存储器。最大跟踪存储器容量如下：

表5-11 跟踪存储器大小

连接的IE	最大值
IECUBE	128KB

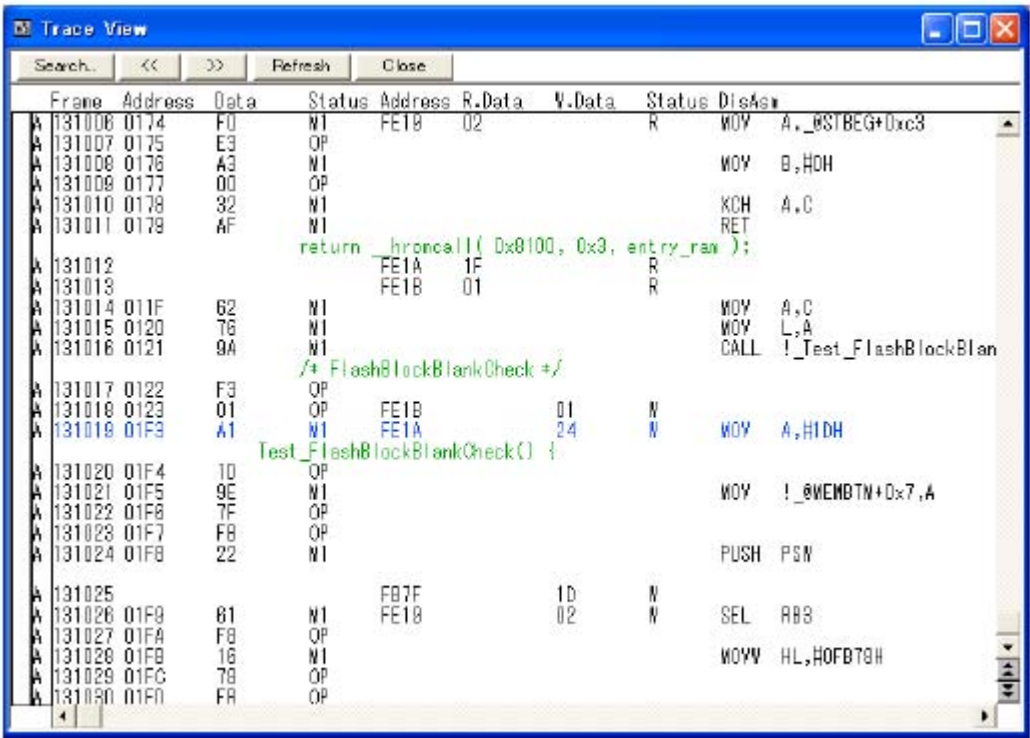
5.10.2 检查跟踪数据

可以在跟踪查看窗口中对保存在跟踪存储器中的跟踪数据进行检查。跟踪数据是在跟踪搜索对话框中进行搜索的。此对话框通过点击< Search...>按钮来显示。

可以在跟踪移动对话框中改变显示的开始位置。此对话框通过点击[View] -> [Move]按钮来显示。

在跟踪查看窗口中显示的条目可以在跟踪数据选择对话框中进行选择。

图5-19检查跟踪数据



5.10.3 混合显示模式（跟踪查看窗口）

在跟踪查看窗口中选择[View]→[Mix]可以显示源文件和跟踪结果（混合显示模式）。
如果代码和要显示的程序获取地址相符合，那么指示跟踪程序获取地址结果的行前显示源文件行。

Frame	Time	Address	Data	Status	Address	Data	Status	DisAsm
32757	3	000005A2	85058505	BRM1				br _main+0x21a
			while(1);					
32758	3	000005A2	85058505	BRM1				br _main+0x21a
			while(1);					
32759	3	000005A2	85058505	BRM1				br _main+0x21a
			while(1);					
32760	3	000005A2	85058505	BRM1				br _main+0x21a
			while(1);					

显示的源文件行以绿色着重。
注意：只有当下载加载模块、阅读符号信息并且获取地址、获取数据、获取状态或反汇编结果被显示的时候混合显示模式才有效。

5.10.4 跟踪器操作

跟踪器操作根据用户程序执行格式和跟踪器控制模式而各有不同。
在程序执行期间，通过选择[Run] → [Tracer Start/Tracer Stop]来实现跟踪器操控。

(1) 执行期间的操作

跟踪器操作是不同。可以通过选择[Run] -> [Cond. Trace ON/Cond. Trace ON]来进行不同的操作。

表5-12跟踪模式类型

条目	内容
无条件跟踪	当用户程序执行时跟踪开始，当产生中断时结束。
有条件跟踪	根据跟踪对话框中的条件设置来启动或终止跟踪。 （参见“5.10.5 设置条件跟踪”） 当正在跟踪时产生中断，跟踪也将立即终止。

(2) 在Step In执行期间的操作

跟踪器对每一步的执行都进行跟踪，然后将跟踪数据添加到跟踪存储器中。

(3) 在Next Over期间的操作

跟踪器的操作是不同的，这要依赖于要执行的Next Over的指令。

(a) CALL/CALLT/CALLF指令

跟踪CALL/CALLT/CALLF指令和被调用的子程序

(b) 其他指令

与Step In期间执行的操作相同。

(4) 跟踪器控制模式

有以下几种跟踪控制模式。这些跟踪模式设置可以通过[Run]菜单来实现。

表5-13跟踪器控制模式的类型

模式	内容
不停止	浏览跟踪存储器并从旧帧中重叠写入数据。
全部停止	浏览跟踪存储器，然后停止跟踪器。
全部中断	浏览跟踪存储器，然后停止跟踪器和执行的程序。
延迟触发停止	当产生延迟触发事件时，跟踪数据（延迟帧的数量）并停止跟踪器。
延时触发中断	当产生延迟触发事件时，跟踪数据（延迟帧的数量）并停止跟踪器和执行的程序。

5.10.5 设置条件跟踪

当设置有条件跟踪时，跟踪事件条件就会触发启动、停止跟踪执行。
通过在跟踪对话框中设置跟踪事件状态，就可以设置状态跟踪（参见“5.12 事件功能”）。
选择[Run]→菜单[Cond. Trace ON]，使用状态跟踪。
下面是几种条件跟踪类型。
在ID78K0-QB中，下面的条件跟踪都被设置为单个跟踪事件条件。

表5-14条件跟踪类型

条目	内容， 设置方法
部分跟踪	在两种特殊条件下执行跟踪（在特殊区域）。 通过在跟踪对话框中设置跟踪开始事件和跟踪结束事件并选择[Run]→[Cond. Trace ON]来执行部分跟踪。 选择在ID78K0-QB中，可以同时指定4个范围。
限制性跟踪	只有当满足条件时才执行跟踪。 如果将两个或更多的事件设置为限制性跟踪，就要根据执行条件跟踪来执行限制性跟踪。 通过在跟踪对话框中设置限制性跟踪事件并选择[Run]→[Cond. Trace ON]来执行限制性跟踪。
延迟触发跟踪	如果满足条件，就可以根据延迟数目来执行跟踪。 通过在跟踪对话框中设置延迟触发事件、在延迟计数对话框中设置延迟计数事件并选择[Run]→[Cond. Trace ON]来执行延迟触发跟踪。在此情况下，选择[Delay Trigger Stop]，[Delay Trigger Break]。

5.11 覆盖测试功能（当连接 IECUBE 时）

虽然有几种覆盖测试类型，但是ID78K0-QB只执行C0覆盖测试。

C0覆盖（指令覆盖）： 代码中所有被执行语句的百分比至少是一次。

个别是通过**下载对话框**或**上传对话框**来上传或下载覆盖测试结果（覆盖数据）。

备注：当启动调试器时覆盖数据被清除。

本部分将介绍：

覆盖测试结果显示

一覆盖测试范围

-被执行的覆盖测试位置的显示

5.11.1 覆盖测试结果显示

可以在代码覆盖窗口中检查覆盖测试结果。

在**代码覆盖窗口**中，测试结果是**为功能、部分和中断处理（向量）进行单个地显示**。当产生中断时覆盖测试结果就会被更新（在用户程序执行期间不会自动更新）。

从[Option]菜单中选择[Coverage Clear]来清除覆盖数据。

覆盖数据可以以CSV格式进行保存（参见“5.17.2 窗口显示信息(查看文件)”）。

图5-20覆盖测试结果显示

[illegible]

5.11.2 覆盖测试范围

覆盖测试范围如下。

表5-15代码覆盖测试范围

连接的IE	代码覆盖测试范围
IECUBE	内部ROM空间+外部存储器空间

5.11.3 被执行的覆盖测试位置的显示

在用户程序中被执行的覆盖测试位置显示在源窗口和汇编窗口中（基于覆盖测试信息）。
显示的结果可以作为视图文件保存在源窗口和汇编窗口中（参见“5.17.2窗口显示信息(查看文件)”）。
被执行覆盖测试测试行或地址的数量在下面这个表格中突出显示。在此查看文件中，下面这个表中记号被添加到相关的行数或地址上，取代了以前的背景颜色。

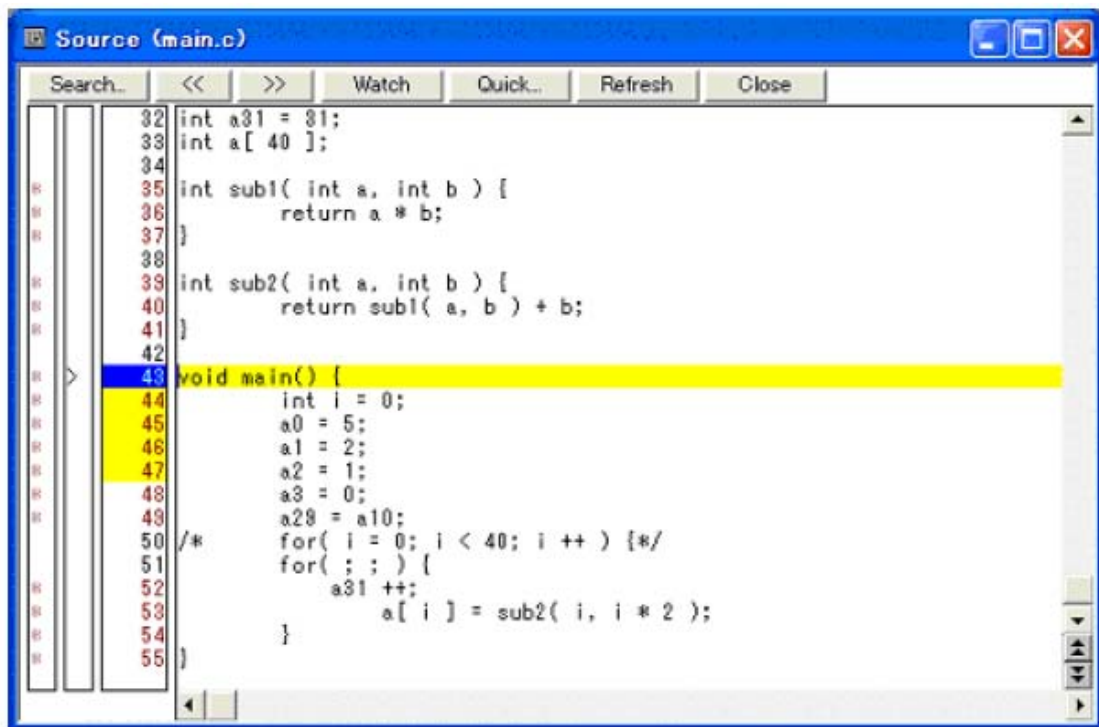
表5-16被执行的覆盖测试位置的查看格式

覆盖	源窗口		汇编窗口	
	背景颜色	记号	背景颜色	记号
此行代码：100%执行。	黄色	'@'	黄色	'@'
此行代码：1%到99%执行。	桔黄色	'+'		
此行代码：0%执行（即还没被执行）。	未加亮颜色	无	未加亮颜色	无

注意：根据PC指向的指令（未被执行的指令）认为是要被部分执行的并显示覆盖测试信息（在源窗口和汇编窗口中使用背景颜色进行加亮，或在代码覆盖窗口中使覆盖变成非0。）。

备注：在ID78K0-QB中，可以在源窗口或汇编窗口中使用上下文菜单清除覆盖数据并显示被执行的覆盖测试位置的累积查看。

图5-21被执行的覆盖测试位置的查看



5.12 事件功能

在调试期间事件可以指定目标系统的特殊状态，比如“获取的地址0x1000”或“将数据写入到0x2000”。在ID78K0-QB中，这些事件是作为每个调试功能的活动触发器，比如中断和跟踪。

本部分将介绍：

- 使用事件功能
- 创建事件
- 设置事件状态
- 每个事件状态中有效事件的数量

5.12.1 使用事件功能

事件（事件状态和事件rlink状态）是由下面这个表列出的事件状态组成（通过分配各种调试功能）。根据调试目的来利用事件状态。

表5-17各种事件状态

事件状态	记号	内容
		→ 设置对话框
中断事件	B	在此状态下，用户程序的执行或跟踪器的操作被停止。（参见“5.4 中断功能”） → 中断对话框
跟踪事件	T	在此状态下，用户程序的执行过程被保存在跟踪寄存器中。（参见“5.10 跟踪功能（当连接IECUBE时）”） → 中断对话框
计时器事件	Ti	在此状态下，指定时间测试开始和结束。（参见“5.9 计时器功能（当连接IECUBE时）”） → 计时器对话框
存根事件	Y	在此状态下，指定插入程序的时机。（参见“5.1 存根功能（当连接IECUBE时）”） → 存根对话框
截图事件	S	在此状态下，指定执行截图的时机。（参见“5.13 截图功能（当连接IECUBE时）”） → 截图对话框
事件DMM	M	在此状态下，指定执行DMM的时机。（参见“5.16 DMM功能”） → 事件DMM对话框

5.12.2 创建事件

事件可以作为各种事件状态（上述所描述的）的行为触发器（通过注册事件状态和事件链接状态），单个地命名事件状态。

(1) 创建和注册事件

在事件对话框中创建事件状态。

在此对话框中设置地址条件、状态条件和数据条件。指定他们的结合体作为单个事件状态，并对此事件状态命名和注册。简单方法存在于使用事件状态（通过在源窗口和汇编窗口中设置断点产生）。（参见“5.4.2 断点设置”）。

(2) 创建和注册事件链接

事件链接状态是单个事件状态，对事件状态提供有序限制，当执行用户程序时根据特殊的顺序产生。

为了创建事件链接状态，您需要使用事件链接对话框。

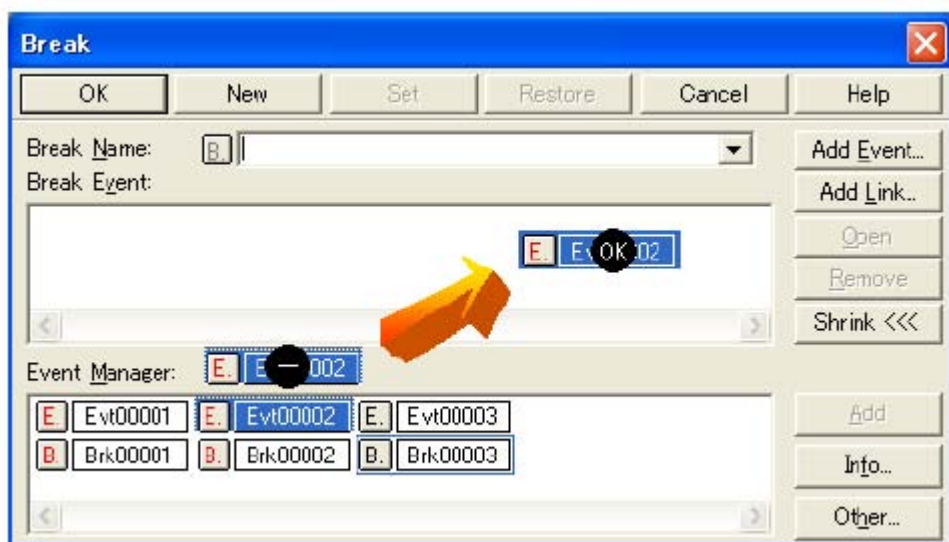
5.12.3 设置事件状态

各种事件状态参见表5-17，他们是在相应的对话框中进行创建。

(1) 设置各种事件状态

通过选择需要的事件状态的事件图标来设置各种事件状态，这些事件状态显示在各自设置对话框的事件管理器区域（或事件管理器）中，然后将此图标拖放到要设置的状态区域中。

图5-22设置各种事件状态



当事件被拖放到可设置状态区域中时，鼠标形状变成“OK”。

在创建事件状态时，事件图标变成红色，通过点击设置对话框中的<Set>按钮或<OK>按钮就可以使设置有效。设置完事件以后，即可以根据各种事件状态来进行调试。

(2) 设置使用选择模式（检查内容之后设置）

将焦点放置在状态区域中，然后单击<Add Event...>按钮或<Add Link...>按钮，就可以在选择的模式中打开**事件对话框**或**事件链接对话框**。当选择对话框中设置的状态时，会显示相应状态细节方面的信息，检查内容之后即可以对状态进行设置。

(3) 复制和移动事件图标

在事件状态设置区域，使用下面的方法通过拖放操作即可以对事件状态进行复制和移动。

-如果只使用鼠标来拖放事件状态，它表示移动事件状态。

-如果按住Ctrl键来拖放事件状态，它表示复制事件状态。

(4) 在事件管理器区域的操控

将焦点放置在状态区域中，选择事件图标，然后单击<Add>按钮，就可以设置事件状态。

事件设置内容显示

选择一个事件，然后单击<Open>按钮或双击此事件。那么将打开与选择的事件对应的设置对话框，并显示事件的设置内容。

删除

选择一个事件，然后单击<Remove / Delete>按钮或按住Delete键即可将此事件删除。

改变显示模式和排序

通过单击<Info...>按钮，可以选择事件管理器区域中的显示模式和排序材料。

非显示区域

通过单击<Shrink<<< >按钮可以隐藏一个区域。

5.12.4 每个事件状态的有效事件数

可以将256个状态注册为事件状态或各种事件状态。

一个事件状态或链接事件状态可以设置为多个事件类型，比如中断，跟踪和定时。但是，被同时设置的事件状态数量（有效的）会有如下限制。

因此，如果超过有效数量或使用的事件状态超过可以同时使用的最大数量，就需要立即取消各种设置的事件状态，然后再对他们重新注册。（参见“5.12.5管理事件”）

注意：除了事件前执行，已经设置地址范围或数据范围的事件状态在内部都使用2个事件状态，因此同时使用的事件状态数将会减少（根据那些事件状态数）。

但是对于执行前的事件，只能设置一个事件状态，即使设置了地址范围也是如此。

表5-18每个事件状态的有效事件数

连接的IE	事件		事件链接	中断	跟踪	截图	存根	计时器	事件DMM
	执行	访问							
IECUBE	8	10 ^a	2	34 ^b	1 ^c	3	3	2	4
MINI CUBE	1	1	–	2 ^d	–	–	–	–	–

a. 8个专门字节访问事件和2个专门字访问事件。

b. 包括执行前16个中断。

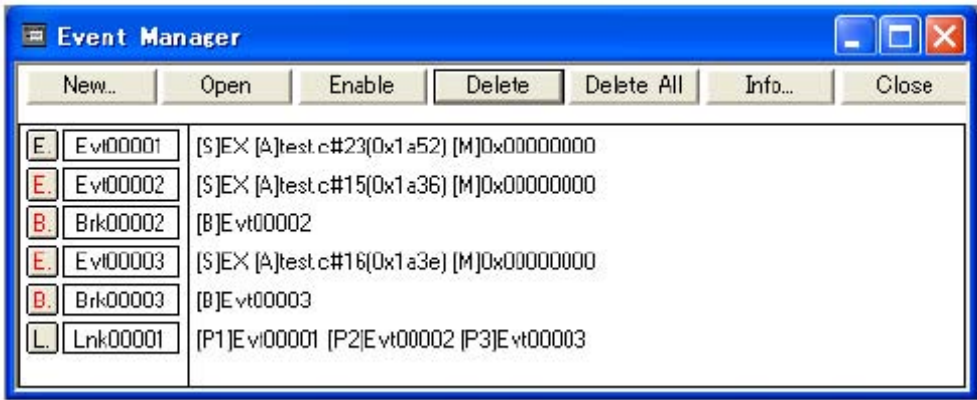
c. 可以同时设置4个部分跟踪事件，1个限制性跟踪和1个延迟触发跟踪。

d. 执行前一个中断（不包括软件中断）。

5.12.5 管理事件

在事件管理器实现事件管理。
事件管理器允许对各种事件状态进行显示、启动/取消和删除。

图5-23管理事件（事件管理器）



(1) 事件图标

事件图标由标记和事件名组成，表示事件的类型。每个事件图标的颜色表示此事件的设置状态。
通过点击记号部分来转换启动/取消。

表5-19事件图标

字体颜色	记号	含义
红色	E.L.	表示事件状态或事件链接状态(用于各种事件状态)有效。
	B.T.Ti.U.S.M.	表示各种事件状态是有效的。 当满足条件时就会产生各种事件。
黑色	E.L.	表示事件状态或事件链接状态(用于各种事件状态)是无效的。
	B.T.Ti.U.S.M.	表示各种事件状态是无效的。 当满足条件时不会产生各种事件。
黄色	E. L.	表示每个事件指定的符号不确定，因为它不能被当前加载的程序识别。
	B. T. Ti. U. S. M.	表示各种事件状态不确定。 当满足条件时不会产生各种事件。

5.13 截图功能（当链接 IECUBE 时）

截图功能是用在用户程序执行过程中寄存器、存储器和SFR中的内容作为捕获的数据保存到跟踪存储器中。

本部分将介绍：

-截图事件状态

-捕获数据

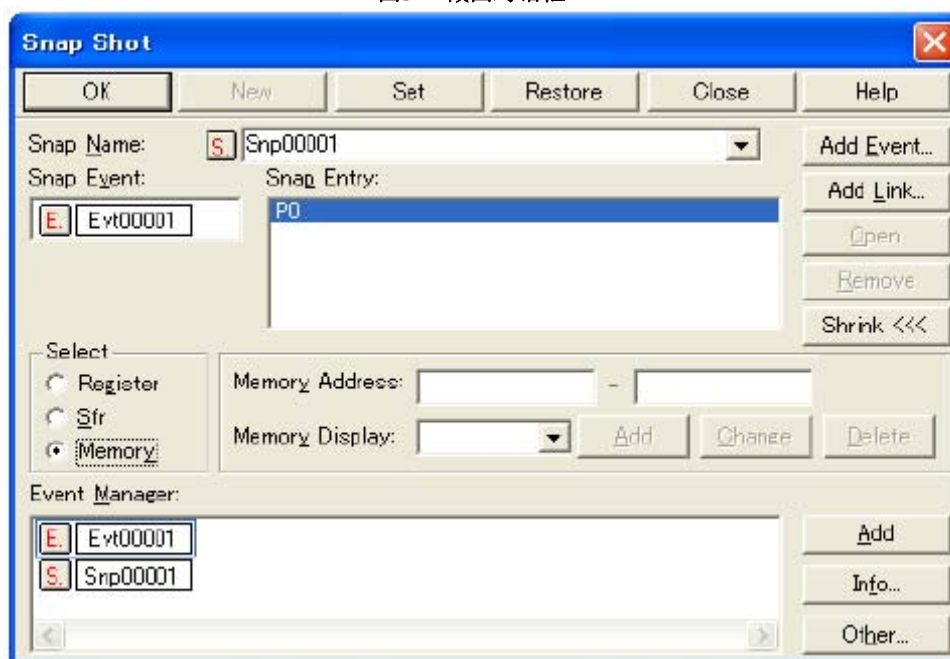
5.13.1 截图事件状态

截图事件状态是指定触发将要被执行的截图。

在[截图对话框](#)中设置截图事件状态。（参见“5.12 事件功能”）。

当执行截图时，执行的用户程序将立即中断。

图5-14截图对话框



5.13.2 捕获数据

一个捕获事件状态可以指定包含16个捕获数据的集合体。

下面这些数据类型可以作为捕获数据进行集合。

-注册值

-SFR值

-存储器内容

5.14 存根功能（当连接 IECUBE 时）

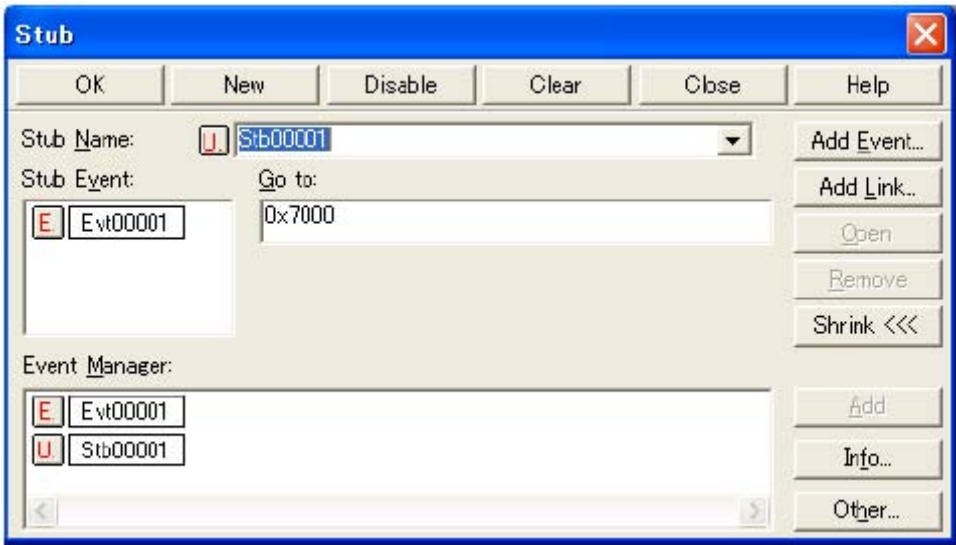
存根功能：当产生事件时，提前执行下载到或写入到空闲存储区域中的用户程序（子程序）。
注意：将RETB指令增加到子程序（当产生存根事件时执行）的末端，否则，会产生功能失常。
本部分将介绍：

- 设置存根事件状态
- 存根流功能

5.14.1 设置存根事件状态

存根事件状态指定通过触发器来执行存根功能。在存根对话框中设置存根事件状态。（参见“5.12 事件功能”）。
当执行存根事件时，正在执行的用户程序将立即中断。

图5-25设置存根事件状态



当产生存根事件时，执行的功能开始地址如下。

表5-20将要被执行的功能开始地址（存根功能）

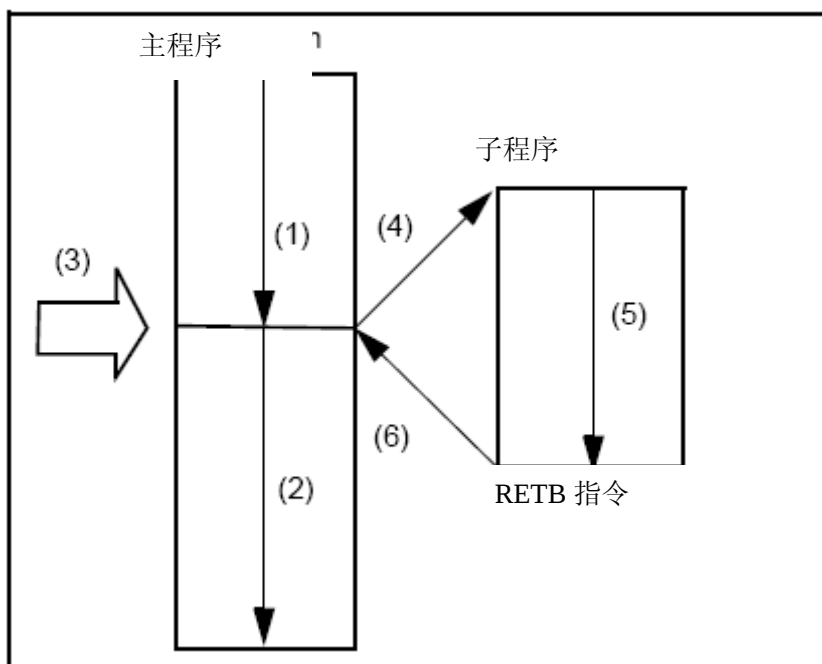
设置范围
0<=功能开始地址<= 0xFFFF\

备注：即使使用存储bank也不能改变指定的地址范围。如果地址设置在0x8000和0xBFFF之间，那么当建立存根事件时，要跳转到对应地址中执行（当前bank）。

5.14.2 存根流功能

存根流功能如下图。

图5-26存根流功能



(1) 如果不使用存根功能

执行(1)和(2)。

(2) 如果使用存根功能

- 1) 执行(1)。
- 2) 在(3)中删除存根事件，并产生中断。
- 3) 内电路仿真器在当前PC中形成一个调用堆栈（在(4)中），然后将PC值重新写入到子程序的入口地址中。
- 4) 执行(5)。
- 5) 通过在子程序末端的RETB指令转回到主程序中执行。

5.15 RRM 功能

本部分将介绍与RRM功能相关的内容：
-实时RAM监控功能
-伪实时RAM监控功能（当读出时中断）

5.15.1 实时 RAM 监控功能

表5-21显示的是使用实时RAM监控功能下载的数据范围。
分配到此区域中的变量和数据可以实时的显示在查看窗口和存储器窗口中。
在扩展选项对话框中可以指定采样时间间隔。

表5-21实时RAM监控功能采样范围

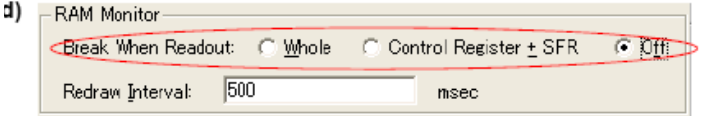
连接的IE	采样范围
IECUBE	内部ROM/内部RAM区域，通用目的寄存器
MINICUBE	无

5.15.2 伪实时 RAM 监控功能（当读出时中断）

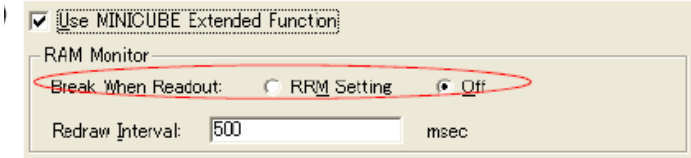
此功能是通过软件仿真器来执行伪实时RAM监控。当利用此功能读取数据时，在执行用户程序期间立即产生一个中断（读操作）。
分配到此区域中的变量和数据可以几乎实时的显示在查看窗口和存储器窗口中^{注意}。
在扩展选项对话框中指定打开/关闭伪实时RAM监控功能和采样范围。
注意：在此情况下，目标存储区域以“**”. 显示。
备注：只有当在扩展选项对话框中选择(1) 使用MINICUBE扩展功能此功能才有效（当连接MINICUBE时）。

图5-27伪实时RAM监控功能规范说明

（当连接 IECUBE 时）



（当连接 MINICUBE 时）



5.16 DMM 功能

DMM（动态存储修改）功能：用户程序执行期间实时地重新书写存储器（RAM）的内容。

在**存储器窗口**、**寄存器窗口**或**SFR窗口**中点击按钮来打开**DMM对话框**。指定DMM目标地址和数据。

备注1：可以通过检查**DMM对话框**中的“Break When Write”来选择伪DMM（当连接IECUBE时）。利用伪DMM可以在用户程序执行期间，当进行写入操作的时候立刻产生中断。

如果指定一个不允许写入的区域（比如寄存器和SFR），那么就需要固定伪DMM的使用。

备注2：当连接MINICUBE时，执行伪DMM。只有当在**扩展选项对话框**中选择**(1) 使用MINICUBE扩展功能伪DMM功能**才有效。利用伪DMM，在执行用户程序期间立即产生一个中断（写操作）。

图5-28DMM对话框



本部分将介绍：

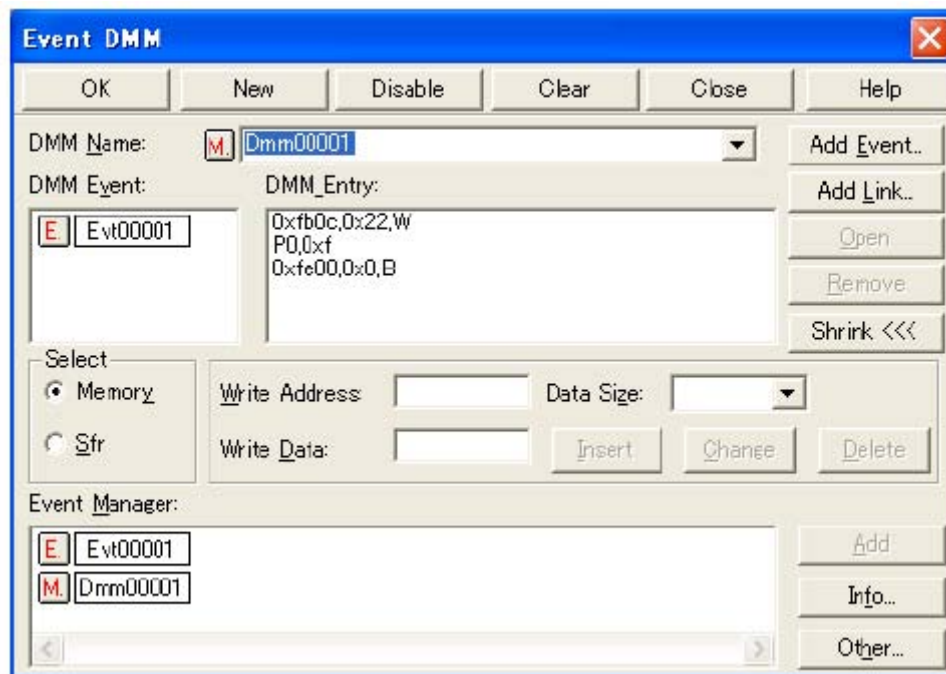
-事件DMM状态（当连接IECUBE时）

5.16.1 事件 DMM 状态（当连接 IECUBE 时）

事件DMM功能：当在用户程序执行期间建立特殊事件时，将任何数据写入到指定的地址当中（存储器或SFR）。在执行用户程序期间，当进行写入操作的时候立即产生一个中断。

在事件DMM对话框中设置事件DMM状态（参见“5.12事件功能”）。

图5-29 事件DMM对话框



5.17 加载/功能保存

ID78K0-QB可以将以下信息类型以文件形式进行保存和加载。这样就可以恢复各种类型的信息。

- 调试环境（工程文件）
- 窗口现象信息（查看文件）
- 窗口设置信息（设置文件）

备注：通过选择[Window]菜单→[Static]可以保持简单窗口的状态。（参见“5.18.1 激活状态和静止状态”）。

5.17.1 调试环境（工程文件）

工程文件(*.prj)是记录调试环境的文件。

当调试环境在特殊点实时保存时，创建工程文件，在以后工过程中也可以通过加载此文件来恢复调试环境。

工程文件分别加载和保存在[工程文件保存对话框](#)和[工程文件加载对话框](#)中。为了在启动时加载工程文件，您需要在[配置对话框](#)中点击<Project...>按钮。

下面内容保存在工程文件中：

表5-22保存到工程文件中的内容

窗口名称	保存的内容
配置对话框	所有条目（目标设备，时钟设置，pin掩码设置，映射信息）
主窗口	显示位置，工具条/状态条/按钮显示信息，执行代码信息
下载对话框	下载的文件信息
扩展选项对话框 调试器选项对话框 失效保护断点对话框 伪仿真对话框 记号选项对话框 RRM对话框	设置信息
汇编窗口 存储器窗口	显示窗口信息，显示开始地址
源窗口 堆栈窗口 SFR窗口 本地可变窗口 跟踪查看窗口 代码覆盖窗口 事件管理器 控制台窗口 扩展窗口 寄存器窗口	显示窗口信息

窗口名称	保存的内容
事件对话框	显示窗口信息，事件信息
事件链接对话框	显示窗口信息，链接事件信息
中断对话框	显示窗口信息，中断事件信息
跟踪对话框	显示窗口信息，跟踪事件信息
计时器对话框	显示窗口信息，计时器事件信息
截图对话框	显示窗口信息，截图事件信息
存根对话框	信息窗口信息，存根事件信息
事件DMM对话框	显示窗口信息，事件DMM信息
查看对话框	显示窗口信息 ^{注意} ，查看注册信息
增加I/O端口对话框	增加I/O端口信息
DMM对话框	DMM信息
延迟计数对话框	延迟计数值
软件中断管理器	显示窗口信息，软件中断信息

注意：不保存结构体指针、数组指针等成员的显示状态，也不保存显示独立构件的基数。

5.17.2 窗口显示信息（查看文件）

查看文件是记录窗口显示信息的文件。

可以为每个窗口加载和保存查看文件。

当加载一个查看文件时，将显示一个参考窗口（在静止状态下的源窗口），并显示保存期间所显示的信息。

查看文件分别加载和保存在视图文件保存对话框和查看文件加载对话框中。

表5-23查看文件的类型

文件类型	当前窗口名，文件名
源文本(*.svw)	源窗口 ^{备注1}
汇编(*.dis)	汇编窗口 ^{备注1}
存储器(*.mem)	存储器窗口
查看(*.wch)	查看窗口
寄存器(*.rgw)	寄存器窗口
SFR (*.sfr)	SFR窗口
局部变量(*.loc)	局部变量窗口
堆栈跟踪(*.stk)	堆栈窗口
跟踪(*.tvw)*	跟踪查看窗口
代码覆盖(*.csv)	代码覆盖窗口（数据为选择的标签单独保存）
控制台(*.log)	控制台窗口
所有(*.*)	所有文件
源(*.c, *.s, *.asm)	源文件 ^{注意2}
文本(*.txt)	文本文件

注意1：当链接IECUBE时，用于指出代码覆盖测试结果（包括执行的/还未执行的）的标记被添加到显示文件的内容当中（参见“Table 5-16被执行的覆盖测试位置的查看格式”）。

备注2：在扩展选项对话框中可以改变源文件的扩展名。

5.17.3 窗口设置信息（设置文件）

设置文件是记录窗口设置信息的文件（查看数据设置，SFR设置和事件设置）
可以为每个窗口加载和保存设置文件。
当加载一个设置文件时，将显示目标窗口并恢复以前被保存的设置信息。
设置文件分别加载和保存在[环境设置文件对话框](#)和[环境设置文件保存对话框](#)中。

表5-24设置文件类型

文件类型	当前窗口名
查看(*.wch)	查看窗口 ^{注意}
SFR (*.sfr)	SFR窗口 ^{注意}
事件(*.evn)	事件管理器

注意：不能加载可变值。

5.18 每个窗口共同的功能

这些窗口具有以下共同的功能：

- 活动状态和静止状态
- 跳转函数
- 链接窗口跟踪结果（当连接IECUBE时）
- 拖放功能
- 注意事项

5.18.1 活动状态和静止状态

下面每个窗口都有两个状态：[活动状态](#)和[静止状态](#)。

- 源窗口（显示源文件，读取符号信息）
- 汇编窗口
- 寄存器窗口

在活动状态只能打开一个窗口。但是在静止状态下能打开两个或更多的窗口，所以窗口的当前状态能够短暂保持。通过[Window]菜单选择此状态。

(1) 活动状态

在活动状态下，自动更新窗口的显示位置和内容，这和当前PC有关。

此窗口也是[跳转函数](#)的跳转目的地。如果此窗口和[跟踪查看窗口](#)链接，那么显示在活动窗口中的内容将被更新，这和[跟踪查看窗口](#)有关。

在活动状态下只能打开一个窗口。

(2) 静止状态

在静止状态下，窗口显示的位置不会根据当前PC值而移动，但是显示的内容会被更新。

静态窗口不能作为[跳转函数](#)的跳转目的地。此外，此窗口也不和[跟踪查看窗口](#)链接。

如果打开一个活动窗口，那么下一个窗口是以静止状态打开。

可以同时打开两个或更多静止窗口。

5.18.2 跳转函数

函数作用：从光标放置的行或地址（跳转指针）上跳转到下面列出的任何一个窗口中。跳转的窗口显示在跳转指针上。

- 源窗口
- 汇编窗口
- 存储器窗口

您可以在上面这些窗口之间跳转，或从[跟踪查看窗口](#)、[堆栈窗口](#)、[事件管理器](#)和[寄存器窗口](#)跳转到上面这些窗口中。

(1) 跳转方法

跳转方法如下：

- 1) 将光标移动到要被作为跳转指针的行或地址上，当然是在能够实现跳转的窗口中进行（在事件管理器中选择一个事件图标）。
- 2) 从菜单[Jump]选择要执行的跳转。

注意：如果光标所在行不存在程序代码，那么是将此行的上面或下面具有程序代码那一行的首地址作为跳转指针。

(2) 跳转源地址的细节信息。

跳转源地址的细节信息如下：

表5-25跳转源地址的细节信息。

目标窗口	跳转指针细节信息	
从 寄存器窗口	选择的记录	
从 存储器窗口	光标位置的地址	
从 事件窗口	如果的事件图标是事件状态，那么地址状态就作为跳转指针。	
	如果地址状态以点设置	跳转到指定的地址
	如果地址状态以范围设置	跳转到较低的位置（如果指定了记号，是指记号前面的点地址）
	如果地址状态以位设置	跳转到位位置的地址

目标窗口	跳转指针细节信息	
从堆栈窗口	在光标位置上的函数（指示栈的帧数）是被作为跳转指针。	
	利用当前函数	
	如果跳转目的地是源窗口	跳转到当前PC行
	除了上述这个窗口	跳转到当前PC地址
	利用除了当前函数的其他函数	
	如果跳转目的地是源窗口	跳转到调用嵌套功能的行
	除了上述这个窗口	跳转到与调用嵌套功能的指令毗邻的地址
从跟踪查看窗口	跳转到存储器窗口	
	如果光标位置是在访问地址、访问数据或访问状态上	访问地址
	除了上述情况	获取地址
跳转到源窗口或汇编窗口	获取地址	

5.18.3 链接窗口跟踪结果（当连接 IECUBE 时）

通过将跟踪查看窗口和每个窗口进行链接（源窗口，汇编窗口或存储器窗口），在链接的窗口上显示相应的部分信息，使用跟踪查看窗口中光标位置的地址作为指针。

如果将光标移动到跟踪查看窗口中，那么会在链接的窗口上突出显示相应的部分信息或由光标位置指出。

(1) 链接方法

链接方法如下：

- 1) 将跟踪查看窗口设置为当前窗口。
- 2) 利用[View] 菜单→[Window Synchronize] 来选择要链接的窗口。
- 3) 将光标移动到在跟踪查看窗口跟踪结果显示区域中将要链接的行上。
- 4) 使用3) 中选择的行地址作为指针，显示相应的部分信息将会突出显示（或由光标位置指出）在2) 中所选窗口的显示区域中。

备注：如果链接的是存储器窗口，那么链接源地址会根据跟踪结果显示区域中光标位置而变化。

访问地址，访问数据，访问状态→访问地址，其他→获取地址

当链接的是源窗口或汇编窗口，那么获取的地址总是作为指针。

5.18.4 拖放功能

使用下面的方法将选择的突出行数、地址和文本拖放到另一窗口中。

- 1) 拖拽所选的行数、地址或文本
→ 鼠标形状变成一个箭头“-”。
 - 2) 将选择的对象放置在可以被放置的窗口或区域中。
→ 当鼠标被放在选择对象被放置的窗口或区域中时，鼠标形状会由一个箭头“-”变成“OK”。
- 在地址行数拖放的窗口中，可以直接在放置的地址或是从放置的行数而获取的地址上进行操作。例如，通过拖放功能可以很简便地将变量注册到查看窗口中（在源窗口中的这些变量）。

(1) 拖放细节信息

将地址行数放置在相应的位置之后，接下来的操作也是不同，这要取决于地址行数被放置的窗口或区域。

表5-26拖放功能的细节信息（行/地址）

放置的窗口/区域	放置后的操作
在各种事件设置对话框中的事件管理器或事件管理器区域	通过将放置的行数或地址作为地址状态，自动地创建执行事件状态。事件状态名可以自动创建，比如Evt00001，Evt00002等。不指定路径计数。 可以为紧密的符号设置地址状态，格式是：符号名+偏移量值
在各种事件设置对话框中的状态设置区域（不是地址和数据设置区域）	通过将放置的行数或地址作为地址状态，自动地创建执行事件状态。自动创建的事件状态设置在每个设置区域（行数或地址拖放的区域）中。 事件状态名可以自动创建，比如Evt00001，Evt00002等。不指定路径计数。 可以为紧密的符号设置地址状态，格式是：符号名+偏移量值
在各种事件设置对话框中的状态设置区域（地址和数据设置区域）	拖放的行数或地址文本设置在行数或地址被放置的区域中。 可以为紧密的符号设置地址状态，格式是：符号名+偏移量值

表5-27拖放功能的细节信息（字符串）

放置的窗口/区域	放置后的操作	
在各种事件设置对话框中的 事件管理器 或事件管理器区域	如果拖放的文本作为符号转地址值，那么就会在访问状态（所有访问状态）或Execute状态中自动创建事件状态（将转换的地址值作为地址状态）。 事件状态名可以自动创建，比如Evt00001，Evt00002等。不指定数据状态和路径计数。 通过拖放的文本来设置地址状态。 要创建的事件状态和符号之间的关系如下：	
	符号	状态
	变量	访问（R/W）
	函数	执行
	数据部分中的符号	访问（R/W）
	代码部分中的符号	执行
	其他	访问（R/W）
在各种事件设置对话框中的状态设置区域（不是地址和数据设置区域）	如果拖放的文本作为符号转地址值，那么就会在访问状态（所有访问状态）或Execute状态中自动创建事件状态（将转换的地址值作为地址状态）。 在每个状态设置区域（行数或地址是被拖放的区域）中自动设置创建的事件状态。 事件状态名可以自动创建，比如Evt00001，Evt00002等。不指定数据状态和路径计数。 通过拖放的文本来设置地址状态。 要创建的事件状态和符号之间的关系如下：	
	符号	状态
	变量	访问（R/W）
	函数	执行
	数据部分中的符号	访问（R/W）
	代码部分中的符号	执行
	其他	访问（R/W）
在各种事件设置对话框中的状态设置区域（地址和数据设置区域）	在此区域中设置拖放的文本	
查看窗口	如果拖放的文本是以符号形式来识别，那么显示的是符号的内容。	

备注：各种事件设置对话框如下。

- 事件对话框
- 事件链接对话框
- 中断对话框
- 跟踪对话框
- 计时器对话框
- 截图对话框
- 存根对话框
- 事件DMM对话框

5.18.5 注意事项

- (1) 能够在每个窗口区域中1行显示的字符串数是319。
- (2) 如果显示的宽度过窄，那么在显示时候就会变得很拥挤。在这种情况下，要增加窗口的宽度。
- (3) 当窗口处于激活状态时，如果改变光标位置，在窗口中使用<Refresh>按钮也许不能成功重新刷新屏幕。点击<Refresh>按钮执行重新刷新操作。
- (4) 利用F1键可以打开帮助。那么显示的帮助内容是与鼠标放置的窗口对应。因为F1键不能放置在跟踪查看窗口中（在此窗口中不显示跟踪结果）（比如启动后），因此即使按下F1键也不能打开帮助。在此情况下，在[Help]菜单中选择[Current Window Help]来打开帮助。
- (5) 在Go & Go执行期间不要从[Run]菜单中选择[Slowmotion]。在Go & Go执行期间[Run]菜单中[Slowmotion]选项通常都是暗淡色，但是有时也会用到此项。因此如果选择[Slowmotion]，那么即使选择了[Run]菜单中[Stop]选项（或点击STOP按钮）也不能终止此程序。
- (6) 如果由于某种原因在处理事件图标时应用程序发生了转换，那么再对此图标进行拖放。
使用ESC键退出拖放操作，然后重新尝试。
- (7) 如果头文件（在这里定义函数）被包含再用户程序中，那么定义的函数要受到以下限制。
（此项来源于编译器中的注意事项(CC78K0)）
 - 不能对在头文件中定义的函数设置断点。
 - 在头文件中定义的函数不能通过选择([Run]菜单→[Step In]来实现逐步执行。
 - 头文件中定义的函数不会在代码覆盖窗口函数列表中。

第6章 窗口参考

本章详细解释ID78K0-QB的窗口和对话框的功能。

— [窗口列表](#)

— [窗口解释](#)

6.1 窗口列表

以下为ID78K0-QB窗口列表。

表6-1窗口列表

窗口名称	内容
主窗口	在ID78K0-QB启动时本窗口最先显示。该窗口控制用户程序的执行，可以从该窗口打开各个窗口。
配置对话框	显示和设置ID78K0-QB的操作环境。
扩展选项对话框	显示和设置ID78K0-QB的扩展选项。
失效保护中断对话框	设置失效保护中断。
RRM对话框	设置RRM采样率。（MINICUBE）
屏蔽选项对话框	设置屏蔽选项和管脚模式。（IECUBE）
闪存选项对话框	设置闪存自编程模拟。（IECUBE）
调试器选项对话框	显示和设置其他选项。
伪仿真对话框	生成伪仿真。（IECUBE）
项目文件保存对话框	将当前的调试环境保存到项目文件。
项目文件加载对话框	加载调试环境。
下载对话框	加载一个目标文件和二进制文件。
上传对话框	将存储器内容保存到文件。
源窗口	显示源文件和文本文件。
源搜索对话框	在源窗口中搜索。
源文本移动对话框	指定在源窗口中显示的文件以及文件开始显示的位置。
汇编窗口	反汇编程序并执行行汇编。
汇编搜索对话框	在汇编窗口中搜索。
地址移动对话框	指定显示存储器窗口或汇编窗口内容的开始地址。
符号转地址对话框	显示指定的变量或函数的地址或显示指定的符号的值。
监视窗口	显示和修改指定的监视数据。
快速监视对话框	显示临时指定的监视数据。
添加监视对话框	记录在监视窗口中显示的监视数据。
修改监视对话框	修改在监视窗口中显示的监视数据。
局部变量对话框	显示和修改当前函数中的局部变量。
堆栈窗口	显示当前的堆栈内容。
存储器窗口	显示存储器的内容。
存储器搜索对话框	在存储器窗口中搜索。

窗口名称	内容
存储器填充对话框	用指定的数据填充存储器内容。
存储器复制对话框	复制存储器。
存储器比较对话框	比较存储器。
存储器比较结果对话框	显示存储器的比较结果。
DDM对话框	将地址和数据主体设置到DDM。
寄存器窗口	显示寄存器的内容。
SFR窗口	显示SFR的内容。
SFR选择对话框	选择在SFR窗口显示的SFR和I/O端口。
添加I/O端口对话框	记录在SFR窗口显示的I/O端口。
计时器窗口	记录和设置计时器事件条件，显示执行时间测量结果。（IECUBE）
计时器结果对话框	显示执行时间测量结果。（IECUBE）
跟踪查看窗口对话框	显示跟踪数据。（IECUBE）
跟踪搜索对话框	搜索跟踪数据（IECUBE）
跟踪数据选择对话框	选择在跟踪查看窗口中显示的项目。（IECUBE）
跟踪移动对话框	指定在跟踪查看窗口中显示的内容的起始地址。（IECUBE）
跟踪对话框	记录和设置跟踪事件触发条件。（IECUBE）
延迟计数对话框	设置一个延迟触发跟踪事件的延迟计数。（IECUBE）
代码覆盖窗口	显示代码覆盖的结果。（IECUBE）
软件中断管理器	显示、启用或禁用和删除软件中断。
事件管理器	显示、启用或禁用和删除软件事件条件。
事件对话框	记录事件条件。
事件链接对话框	记录事件链接条件。（IECUBE）
中断对话框	记录和设置中断事件条件。
截图对话框	记录和设置截图事件条件。（IECUBE）
存根对话框	记录和设置存根事件条件。（IECUBE）
事件DDM对话框	记录和设置事件DDM条件。（IECUBE）
视图文件保存对话框	将当前窗口的显示信息保存到一个视图文件
视图文件加载对话框	加载每个窗口的视图文件。
环境设置文件保存对话框	将当前窗口的设置信息保存到一个设置文件
环境设置文件加载对话框	加载每个窗口的设置文件。
复位调试器对话框	初始化ID78K0-QB、CPU和符号信息。
退出调试器对话框	终止ID78K0-QB。
关于对话框	显示ID78K0-QB的版本。

窗口名称	内容
控制台窗口	输入命令。
浏览对话框	选择要设置的文件。

6.2 窗口的解释

本节解释每个窗口或对话框。

窗口名称 / 对话框名称

本节简要地解释窗口或对话框以及要注意的地方。

如果由于连接的IE造成无效的窗口或对话框存在，则无效连接的IE的名称显示在窗口或对话框名称的右下方。

此外，还展示了窗口或对话框的图象，并解释了相关的操作。

打开

解释如何打开窗口或对话框，

每个区域的解释

解释在窗口或对话框的每个区域中设置或显示的项目。

右键菜单

解释点击鼠标右键时出现的右键菜单。通常可以从右键菜单中通过一个动作选择此窗口中常用的功能（仅限窗口）。

功能按钮

解释窗口或对话框中每个按钮的操作。

相关操作

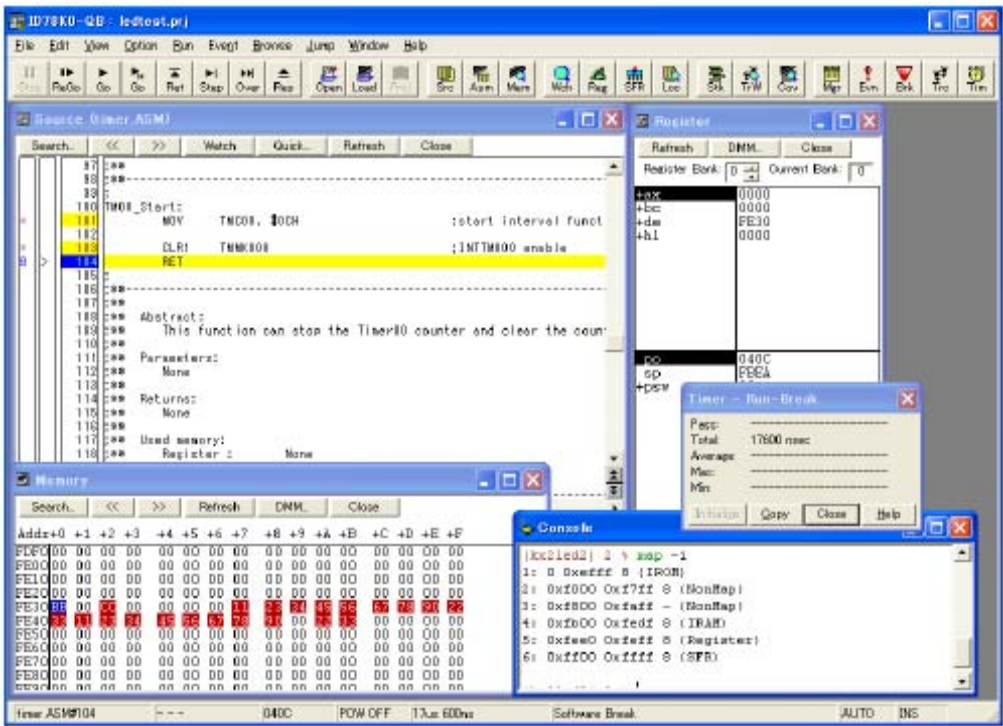
解释与此窗口或对话框相关的一个窗口或对话框的操作。

主窗口

本窗口在ID78K0-QB启动或初始化时自动打开。
在ID78K0-QB中，其他窗口是通过本窗口操作的。（参见“表6-1：窗口列表”。）
在本窗口中控制用户程序的执行。
用户程序的执行通过以下三种模式控制：

- 源模式（在源层次调试用户程序）
- 指令模式（在指令层次调试用户程序）
- 自动模式（自动选择源模式或指令模式）（默认）

图6-1：主窗口



- 菜单条
- 工具条
- 窗口显示区域
- 状态条

菜单条

- (1) [菜单(File)]菜单
- (2) [编辑(Edit)]菜单
- (3) [查看(View)]菜单
- (4) [选项(Option)]菜单
- (5) [运行(Run)]菜单
- (6) [事件(Event)]菜单
- (7) [浏览(Browse)]菜单
- (8) [跳跃(Jump)]菜单
- (9) [窗口(Window)]菜单
- (10) [帮助(Help)]菜单

(1) [菜单(File)]菜单

打开...	加载一个视图文件、源文件或文本文件。打开 视图文件加载对话框 。此操作根据对话框中选择的文件的扩展名而不同。
另存为...	将当前窗口中显示的内容保存到指定名称的文件。 打开 视图文件保存对话框 。
关闭	关闭当前窗口。
下载	下载一个文件。 打开 下载对话框 。
上传	上传一个文件。 打开 上传对话框 。
项目	操作一个项目文件。
打开...	下载一个项目文件。 打开 项目文件加载对话框 。
保存	覆盖当前加载到ID78K0-QB的项目文件的状态。
另存为...	保存当前状态到一个指定的项目文件。 打开 项目文件保存对话框 。
环境	操作一个设置文件。
打开...	下载一个设置文件。 打开 设置文件加载对话框 。
另存为...	保存当前状态到一个指定的设置文件。 打开 设置文件保存对话框 。
调试器...	初始化CPU、符号和ID78K0-QB。 打开 复位调试器对话框 。
退出	终止ID78K0-QB。 打开 退出调试器对话框 。
(打开文件)	列出打开的文件的名称。

(2) [编辑(Edit)]菜单

剪切	剪切选中的字符串并将其保存到剪贴板缓存。
复制	复制选中的字符串并将其保存到剪贴板缓存。
粘贴	将剪贴板缓存中的内容保存到光标位置。
写入	将修改的内容写入到目标。
恢复	撤销修改。
存储器	操作存储器的内容。
填充...	用指定的代码填充存储器的内容。 打开 存储器填充对话框 。
复制...	复制存储器。 打开 存储器复制对话框 。
比较...	比较存储器。 打开 存储器比较对话框 。
DMM...	在用户程序执行的过程中实时重写存储器内容。打开 DMM对话框 。（当MINICUBE连接且使用(1) MINICUBE扩展功能 清空时不能选择。）
编辑源	在PM+运行时用PM+指定的编辑器在活动的 源窗口 中显示的源文件。

(3) [查看(View)]菜单

[查看(View)]菜单包括共同部分以及根据活动窗口加上的专门部分。关于专门部分的详细信息，参见每个窗口的描述。

(a) 共同项目

搜索...	进行搜索。 打开当前窗口对应的搜索对话框。 与<搜索>按钮操作相同。
移动...	移动显示位置。 打开当前窗口对应的规定对话框。
快速监视...	临时显示指定数据的内容。 打开 快速监视对话框 。
添加监视...	将指定数据记录到监视窗口。 打开 添加监视对话框 。
查看监视...	将选择的数据添加到监视窗口。 如果数据是符号，根据 调试器选项对话框 的设置添加。
修改监视...	修改监视窗口中选择行中的数据。 打开修改监视对话框。 此菜单仅在 监视窗口 中选中一个变量时有效。
删除监视...	从 监视窗口 中删除选择的监视点。 此菜单仅在 监视窗口 中选中一个变量时有效。
符号...	显示指定变量或函数的地址或指定符号的值。 打开 符号转地址对话框 。

(4) [选项(Option)]菜单

工具条	选择工具条是否显示（默认为显示）。
状态条	选择状态条是否显示（默认为显示）。
按钮	选择每个窗口上的按钮是否显示（默认为显示）。
源模式	在源层次（在线单元中）选择分步执行。
指令模式	在指令层次（在指令单元中）选择分步执行。
自动模式	自动选择在源层次或指令层次（默认）选择分步执行。 如果源窗口处于活动状态，分步执行在源层次进行；如果汇编窗口处于活动状态，分步执行在指令层次进行；如果没有窗口处于活动状态，分步执行在源层次进行。
配置...	设置环境。 打开配置对话框。
屏蔽选项...	进行屏蔽选项的设置和管脚模式设置（当IECUBE连接时）。 打开屏蔽选项对话框。
扩展选项...	设置扩展选项。 打开扩展选项对话框。
RRM设置...	设置RRM功能的采样范围。 打开RRM设置对话框（当MINICUBE连接时）。
闪存选项...	此对话框用于进行闪存自编程仿真设置。（当IECUBE连接时） 打开闪存选项对话框。
调试器选项...	设置ID78K0-QB的选项。 打开调试器选项对话框。
添加I/O端口...	添加用户定义的I/O端口。 打开添加I/O端口对话框。
跟踪清空	清空跟踪数据。 此项目在跟踪视图窗口处于活动情况下显示（当IECUBE连接时）。
覆盖清空	清空覆盖测量结果（当IECUBE连接时）。

(5) [运行(Run)]菜单

重新开始	复位CPU并执行程序。  与此按钮操作相同。
停止	强制停止程序运行。  与此按钮操作相同。
运行	执行来自当前程序指针的程序。  与此按钮操作相同。
忽略断点并运行	忽略设置的断点并执行程序。  与此按钮操作相同。

返回退出	执行用户程序直到执行返回。  与此按钮操作相同。 注：本命令用于用C语言描述的函数。
单步执行 (Step In)	一步一步执行程序中的指令（分步执行）。 如果调用了函数或子程序，其指令被分步执行。  与此按钮操作相同。
下一步执行 (Next Over)	一步一步执行程序中的指令（下一步执行）。 如果调用了函数或子程序，其指令不被分步执行。  与此按钮操作相同。
从这里开始	执行从 源窗口 或 汇编窗口 中光标位置起的程序。
到这里	执行从当前程序指针到 源窗口 或 汇编窗口 中光标位置的程序。
继续	继续执行程序。 如果由于中断条件满足而出现中断，则更新窗口并再次执行程序。  每次出现中断时与此按钮操作相同。
慢动作	继续单步执行。 每次进行了单步执行后，更新窗口并再次进行单步执行。  每次出现中断时与此按钮操作相同。
CPU复位	复位CPU。  与此按钮操作相同。
改变程序指针	将 源窗口 或 汇编窗口 中光标位置的地址设置到程序指针。
断点	在 源窗口 或 汇编窗口 中的光标位置设置或删除一个断点。
软件断点	在 源窗口 或 汇编窗口 中的光标位置设置或删除一个软件断点。
删除所有的断点	删除所有设置的断点（包括软件断点）。
无条件跟踪开启	验证无条件跟踪，这样在程序执行过程中跟踪总是被执行（默认）。（当连接IECUBE时）。 此时设置的跟踪事件条件被忽略。
有条件跟踪开启	在程序执行过程中根据跟踪事件条件验证有条件跟踪。（当连接IECUBE时）。
跟踪器控制模式	设置跟踪控制模式。（当连接IECUBE时）。

不停止	遍历跟踪存储器并从最旧的帧开始覆盖数据（默认）。
完全停止	遍历跟踪存储器然后停止跟踪器。
完全中断	遍历跟踪存储器然后停止跟踪器和程序执行。
延迟触发器停止	按延迟计数帧的数量跟踪数据，在延迟触发器事件发生时停止跟踪器。
延迟触发器中断	按延迟计数帧的数量跟踪数据，在延迟触发器事件发生时停止跟踪器和程序执行。
计时器开始/停止	在计时器停止时开启计时器测量，或在计时器在运行时将其停止。（当连接IECUBE时）。 如果程序未在执行中并且计时器事件未被使用，此项目无效。在程序执行开始后计时器测量立即开始。
跟踪器开始/停止	在跟踪器停止时开启跟踪器，或在跟踪器在运行时将其停止。（当连接IECUBE时）。 如果程序未在执行中，此项目无效。 在程序执行开始后跟踪器立即执行。
伪仿真...	打开 伪仿真对话框 （当连接IECUBE时）。

(6) [事件(Event)]菜单

事件管理器	管理各种事件条件。打开 事件管理器 。  与此按钮操作相同。
软件中断管理器	管理软件中断事件条件。 打开 软件中断管理器 。
事件...	记录一个事件条件。打开 事件对话框 。  与此按钮操作相同。
事件链接...	记录一个事件链接条件。 打开 事件链接对话框 。
中断...	记录并设置一个中断事件条件。打开 中断对话框 。  与此按钮操作相同。
跟踪...	记录并设置一个跟踪事件条件（当连接IECUBE时）。 打开 跟踪对话框 。  与此按钮操作相同。
截图...	记录并设置一个截图事件条件（当连接IECUBE时）。 打开 截图对话框 。
存根（stub）...	记录并设置一个存根事件条件（当连接IECUBE时）。 打开 存根对话框 。
计时器...	记录并设置一个计时器事件条件（当连接IECUBE时）。 打开计时器对话框。  与此按钮操作相同。
事件DMM...	记录并设置一个事件DMM条件（当连接IECUBE时）。 打开 事件DMM对话框 。

延迟计数...	设置延迟计数（当连接IECUBE时）。 打开 延迟计数对话框 。
---------	---

(7) [浏览(Browse)]菜单

源文本	显示一个源文本。打开 源窗口 。 如果此窗口已经打开处于活动状态，窗口在静止状态下打开。  与此按钮操作相同。
汇编	显示反汇编的结果。打开 汇编窗口 。 如果此窗口已经打开处于活动状态，窗口在静止状态下打开。  与此按钮操作相同。
存储器	显示存储器中的内容。打开 存储器窗口 。 如果此窗口已经打开处于活动状态，窗口在静止状态下打开。  与此按钮操作相同。
监视	显示监视中的内容。打开 监视窗口 。  与此按钮操作相同。
寄存器	显示寄存器中的内容。打开 寄存器窗口 。  与此按钮操作相同。
SFR	显示SFR容。打开 SFR窗口 。  与此按钮操作相同。
局部变量	显示局部变量。打开 局部变量窗口 。  与此按钮操作相同。
堆栈跟踪	显示堆栈跟踪的结果。打开 堆栈窗口 。  与此按钮操作相同。
跟踪	显示跟踪结果（当连接IECUBE时）。 打开 跟踪视图窗口 。  与此按钮操作相同。
代码覆盖	显示代码覆盖测量的结果（当连接IECUBE时）。 打开 代码覆盖窗口 。  与此按钮操作相同。
控制台	打开 控制台窗口 。
其他	显示其他窗口。（参见“ A.2扩展窗口的样本列表 ”） 显示用户定义的窗口列表。

(8) [跳跃(Jump)]菜单

源文本	显示对应的源文本和源行，用当前窗口中选择的数据值作为跳跃目标地址。 如果跳跃目标地址中没有行信息，您无法跳跃。 打开 源窗口 。 如果活动的 源窗口 被打开，则该窗口会显示在最前面（这样就可以对窗口进行操作了）。
汇编	反汇编和显示来自当前窗口中选择的数据值指定的跳跃目标地址的结果。 打开 汇编窗口 。 如果活动的 汇编窗口 被打开，则该窗口会显示在最前面（这样就可以对窗口进行操作了）。
存储器	显示来自当前窗口中选择的数据值指定的跳跃目标地址的存储器内容。 打开 存储器窗口 。 如果活动的 存储器窗口 被打开，则该窗口会显示在最前面（这样就可以对窗口进行操作了）。
覆盖	无法选择。

(9) [窗口(Window)]菜单

新窗口	打开一个显示与当前窗口相同内容的新窗口。此菜单仅在当前窗口是 源窗口 、 汇编窗口 或 存储器窗口 是有效。
层叠	在主窗口中层叠显示窗口。
平铺	在主窗口中平铺显示窗口。
安排图标	重新排列主窗口中的图标。
全部关闭	关闭除主窗口外的所有窗口。
刷新	用最新的数据更新窗口的内容。
活动	将窗口设为活动状态。
静态	将窗口设为静止状态。
（打开窗口）	列出打开的窗口。 在图的一侧有构造标记的窗口是当前窗口。通过选择窗口名称，选择的窗口作为当前窗口。

(10) [帮助(Help)]菜单

ID78K0-QB帮助	显示帮助。
命令参考	显示 命令参考 的帮助窗口。
主窗口	显示主窗口的帮助。
当前窗口	显示当前窗口的帮助。
关于...	显示ID78K0-QB的版本。 打开 关于对话框 。

工具条

(1) 每个按钮的含义

(2) 工具条的操作

(1) 每个按钮的含义

工具条中每个按钮的含义如下。当鼠标指针放在工具条的一个按钮上数秒后会出现工具提示。

 Stop	停止用户程序的执行。 与[运行 (Run)]菜单 → [停止]的功能相同。
 ReGo	复位CPU并执行用户程序。 与[运行 (Run)]菜单 → [重新开始]的功能相同。
 Go	不复位CPU即执行来自当前程序指针的用户程序。 与[运行 (Run)]菜单 → [运行]的功能相同。
 Go	忽略设置的断点并执行用户程序。 与[运行 (Run)]菜单 → [忽略断点并运行]的功能相同。
 Ret	执行用户程序直到执行返回。 与[运行 (Run)]菜单 → [返回退出]的功能相同。 注： 本命令用于用C语言描述的函数。
 Step	单步执行（一步一步执行程序中的指令）。 如果调用了函数或子程序，其指令被分步执行。 与[运行 (Run)]菜单 → [单步执行]的功能相同。
 Over	下一步执行（执行程序，假设一个函数/调用语句为一步）。 如果调用了函数或子程序，其指令被分步执行。 与[运行 (Run)]菜单 → [下一步执行 (Next Over)]的功能相同。
 Res	复位CPU。 与[运行 (Run)]菜单 → [CPU复位]的功能相同。
 Open	打开视图文件加载对话框。 与[文件 (File)]菜单 → [打开...]的功能相同。
 Load	打开下载对话框。 与[文件 (File)]菜单 → [下载...]的功能相同。
 Proj	打开项目文件下载对话框。 与[文件 (File)]菜单 → [项目...] → [打开...]的功能相同。
 Src	显示源文本。打开源窗口。 与[浏览 (Browse)]菜单 → [源文本]的功能相同。
 Asm	显示反汇编结果。打开汇编窗口。 与[浏览 (Browse)]菜单 → [汇编]的功能相同。
 Mem	显示存储器内容。打开存储器窗口。 与[浏览 (Browse)]菜单 → [存储器]的功能相同。
 Wch	显示监视内容。打开监视窗口。 与[浏览 (Browse)]菜单 → [监视]的功能相同。
 Reg	显示寄存器的内容。打开寄存器窗口。 与[浏览 (Browse)]菜单 → [寄存器]的功能相同。
 SFR	显示SFR的内容。打开SFR窗口。 与[浏览 (Browse)]菜单 → [SFR]的功能相同。

 Loc	显示局部变量内容。 打开 局部变量窗口 。 与[浏览 (Browse)]菜单 -> [局部变量]的功能相同。
 Stk	显示堆栈跟踪结果。 打开 堆栈窗口 。 与[浏览 (Browse)]菜单 -> [堆栈跟踪]的功能相同。
 TrW	显示跟踪结果 (当连接IECUBE时)。 打开 跟踪视图窗口 。 与[浏览 (Browse)]菜单 -> [跟踪]的功能相同。
 Cov	显示代码覆盖测量的结果 (当连接IECUBE时)。 打开代码覆盖窗口。 与[浏览 (Browse)]菜单 -> [代码覆盖]的功能相同。
 Mgr	打开 事件管理器 。 与[事件 (Event)]菜单 -> [事件管理器...]的功能相同。
 Evn	记录和设置事件。 打开 事件对话框 。 与[事件 (Event)]菜单 -> [事件...]的功能相同。
 Brk	记录和设置中断事件。 打开 中断事件对话框 。 与[事件 (Event)]菜单 -> [中断...]的功能相同。
 Trc	记录和设置一个跟踪事件 (当连接IECUBE时)。 打开 跟踪对话框 。 与[事件 (Event)]菜单 -> [跟踪...]的功能相同。
 Tim	记录和设置一个计时器事件 (当连接IECUBE时)。 打开 计时器对话框 。 与[事件 (Event)]菜单 -> [计时器...]的功能相同。

(2) 工具条的操作

不论工具条是否显示, 可通过选择[选项 (Option)]菜单 -> [工具条]指定。

此工具条以下面两种模式显示。可在[调试器选项对话框](#)中选择模式。

图6-2: 工具条 (图)



图6-3: 工具条 (图+文字)



窗口显示区域

此区域显示各个调试窗口。

可以改变显示窗口的大小或在此区域创建图标。

状态条

状态条显示ID78K0-QB和仿真器的状态。

在用户程序运行时，状态条以红色显示。

不论状态条是否显示，可通过选择[选项（Option）]菜单 -> [状态条]指定。

备注：如果屏幕分辨率较低（800-600等），可能无法在状态条上无法显示所有的状态。

图6-4：状态条



(1) 程序名	显示程序指针值指定的程序文件名。
源名称	显示程序指针值指定的源文件名。
行号	显示程序指针值指定的行编号。
(2) 函数名	显示程序指针值指定的函数名。
(3) 程序指针名	显示程序指针名称。
(4) CPU状态	参见“表6-2：CPU状态”。
(5) IE状态	参见“表6-3：IE状态”。 （如果有两个或多个状态，可用“ ”区分显示。）
(6) 中断原因	参见“表6-4：中断原因”。
(7) 步进模式	显示单步执行模式。 显示从[选项（Option）]菜单选择以下模式： SRC.....源模式 INST.....指令模式 AUTO.....自动模式
(8) 键输入模式	显示键输入模式。 INS.....插入模式 OVR.....覆盖模式 存储器窗口设置为覆盖模式。

表6-2：CPU状态

显示	含义
HALT	暂停模式
STOP	停止模式
WAIT	等待模式
RESET	复位模式
POW OFF	目标无电源。

表6-3: IE状态

显示	含义
RUN	用户程序正在执行（状态条的颜色变化）。
STEP	单步执行在进行中。
TRC	跟踪操作中。
TIM	计时器运行中。
显示	含义
BREAK	中断出现。
Time	显示从用户程序执行到出现中断的事件测量结果（运行一中断时间）。 ^注
TIMER OVERFLOW	测量结果溢出。

注：可测量的范围是20纳秒（最小，时钟脉冲不分）到48小时50分钟（最大，时钟脉冲除以2000）（当连接IECUBE时）。

可测量的范围是100微秒（最小）到119小时18分钟（最大）（当连接MINICUBE2时）。

表6-4: 中断原因

显示	含义
Manual Break	强制中断
Temporary Break	临时中断
Software Break	软件中断
Trace Full Break	跟踪满造成的中断
Trace Delay Break	跟踪延迟造成的中断
Non Map Break	访问了未映射的区域
Write Protect	尝试在写保护区域进行写入操作
Read Protect	尝试读取读保护的区域
SFR Illegal	非法访问了SFR
SFR Write Protect	尝试写入到禁止写入的SFR
SFR Read Protect	尝试读取禁止读取的SFR
Stack Overflow	堆栈溢出造成的中断
Stack Underflow	堆栈下溢出造成的中断
Uninitialize Stack Pointer	未进行堆栈指针初始化造成的中断
Uninitialize Memory Read	对未经初始化的存储器进行了读取
Timer Over Break	检测到执行超时
Fetch Break	获取了受保护的区域或禁止获取的区域

显示	含义
Unspecified Illegal	出现了与周边芯片函数相关的用户程序的非法操作。 参见关于周边仿真板的文档。
IMS (IXS) Illegal	非法写入IMS、IXS寄存器造成的中断
Retry Over	重试计数中断
Flash Illegal	闪存非法中断
Peripheral Break	来自周边设备的中断
显示	含义
Break Event: "事件名称"	由于显示事件名称的事件原因造成的停止
Event Break "事件名称"	由于显示事件名称的事件原因造成的停止
Before Execution "事件名称"	由于显示事件名称的事件原因造成的在执行前的中断

配置对话框

此对话框用于显示和设置ID78K0-QB的运行环境。（参见“5.1 设置调试器环境”。）

此对话框在ID78K0-QB启动后自动显示。

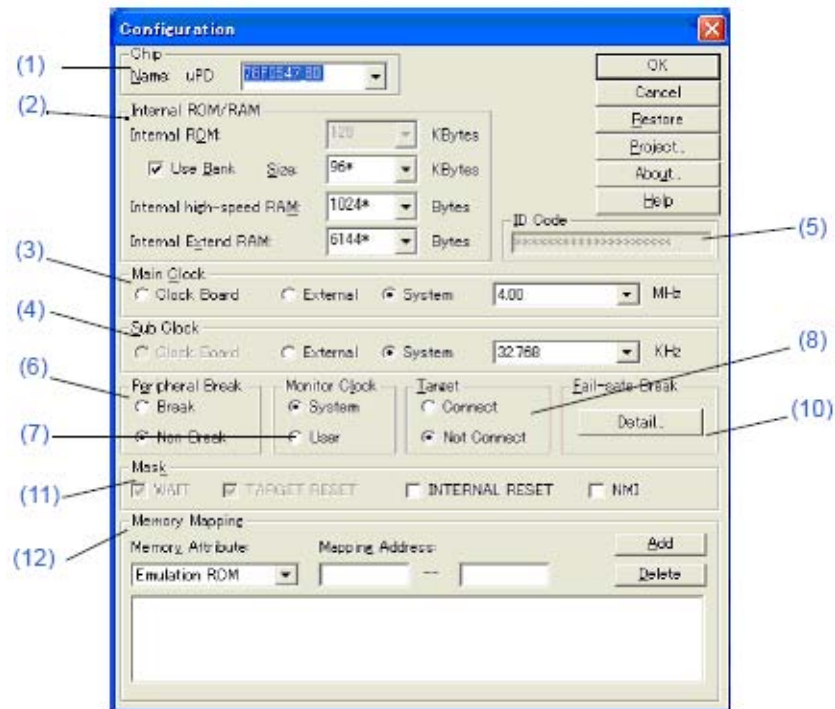
由于读取项目的结果都反映在此对话框中，因此不需要设置即可读取项目。（参见“5.17.1 调试器环境（项目文件）”。）

注意：只有在配置对话框在ID78K0-QB启动后自动显示时才可以设置为“芯片”和“目标”区域（如果在设置这些区域后将调试环境保存到一个项目文件，则配置对话框在ID78K0-QB启动后不再自动显示）。

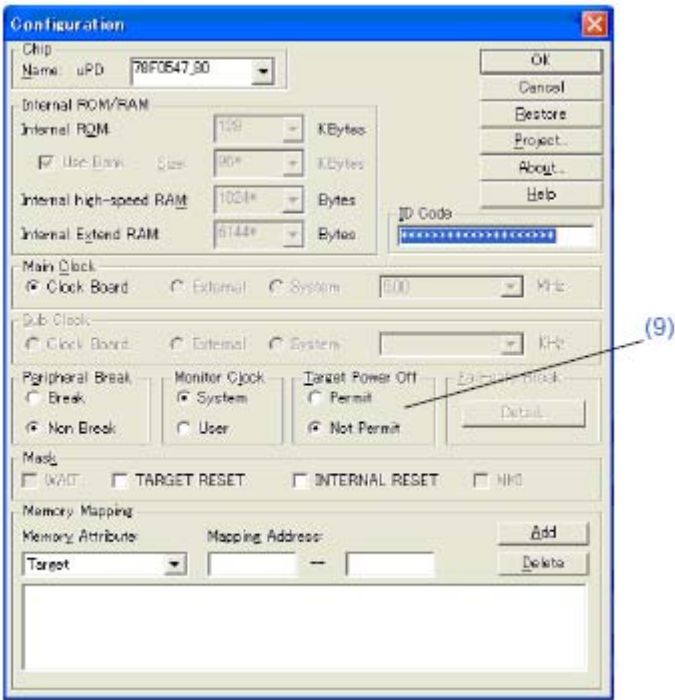
要修改这些区域的设置，打开项目文件所在的文件夹，然后删除扩展名为“.prj”的文件（.prj文件包含了调试器的配置）。

图6-5：配置对话框

（当连接IECUBE时：）



(当连接MINICUBE时：)



- 打开
- 对每个区域的解释
- 功能按钮

打开

(在ID78K0-QB启动后自动显示。)

选择[选项 (Option)]菜单 -> [配置...]

对每个区域的解释

(1) 芯片

此区域用于选择芯片名称。

芯片名称从下拉菜单中选择。

在下拉菜单中仅显示注册到设备文件安装程序的芯片名称。

只有在调试器启动后才能指定该区域。

注：在默认情况下显示的是前一次启动选择的芯片类型，如果该类型没有注册，则显示注册的第一个类型。

(2) 内部ROM/RAM

本区域用于显示CPU内部RAM和内部ROM的大小。（参见“表5-2：映射属性”。）

当连接MINICUBE时，此区域设置为设备文件值。

映射属性从下拉菜单中选择。

默认大小是通过在 (1) 芯片中选择从设备文件中获得的，并以带“*”的数值显示。

表6-5：内部ROM/RAM设置的范围和大小（当连接IECUBE时）

区域	含义	可设置的值
内部ROM ^{注1}	设置内部ROM大小	当存储器组没有使用时：0—60KB。 当存储器组使用时：52—256KB（灰色）（4KB为单位）。
内部存储器组ROM ^{注2}	设置内部存储器组ROM大小	当存储器组没有使用时：0（灰色）。 当存储器组使用时：20—224KB（灰色）（4KB为单位）。
内部高速RAM	设置内部高速RAM大小	64—1024字节（64字节为单位）。
内部扩展RAM	设置内部扩展RAM大小	0—14336字节（512字节为单位）。

注1：如果选择了“使用存储器组”，可修改内部存储器组ROM区域的大小。

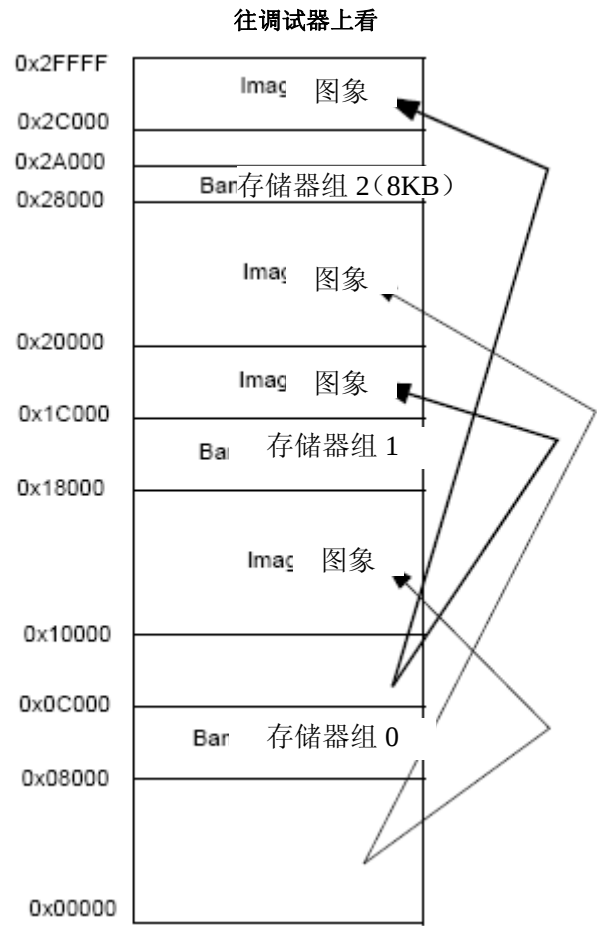
此时，“内部ROM”区域无法选择，内部存储器组ROM区域大小加上32KB（公共区域）变为灰色。

如果存储器组的信息包含在设备文件中，此项目默认为选择。如果没有存储器组信息，则无法选择该项目。

注2：如果内部存储器组ROM被使用，ID78K0-QB中的地址空间如图6-6所示。

因此，对0x10000或以上的地址空间无法指定地址范围（存储器组区域除外）。

图6-6：在内部ROM存储器组没有使用时的地址空间图（存储器组ROM大小为40KB）



(3) 主时钟

选择以下项目之一作为主时钟的源。

时钟板	安装在仿真器中的时钟。
外部	安装在目标系统（矩形波）上的时钟。
系统	在仿真器中生成的时钟。 在下拉列表中选择相关的时钟。 —1.00、2.00、3.00、3.57、4.00（默认）、4.19、4.91、5.00、6.00、8.00、8.38、10.00、12.00、16.00、20.00（当连接IECUBE时） —5.00（默认）（当连接MINICUBE时） —4.00（默认）、8.00、16.00（当连接MINICUBE2时）

当连接IECUBE时，此区域的默认状态和可用选项变化如下。变化取决于目标电源（TVDD）的探测结果（O：可用；X：在选择时会造成错误）。

	TVDD开启	TVDD关闭
时钟板	O	O
外部	O（默认）	X
系统	O	O（默认）

当连接MINICUBE时，如果仿真器的时钟插口安装有振荡器，此区域设置为“时钟板”；否则设置为“系统”。

（4）子时钟

此区域用于选择向CPU输入的子时钟源。

时钟板	安装在仿真器中的时钟。
外部	安装在目标系统（矩形波）上的时钟。
系统	在仿真器中生成的时钟。 在下拉列表中选择相关的时钟。 32.762（默认）、38.400

当连接IECUBE时，此区域的默认状态和可用选项变化如下。变化取决于是否在仿真器上安装了时钟（振荡器或者带有共振系统的主时钟板）以及目标电源（TVDD）的探测结果（O：可用；X：在选择时会造成错误；—：不可用）。

	TVDD开启		TVDD关闭	
	TVDD开启	TVDD关闭	TVDD开启	TVDD关闭
时钟板	O（设置）		—	—
外部	—		O（默认）	X
系统	—		O	O（默认）

当连接MINICUBE时，无法选择。

（5）ID代码

此区域用于在ID78K0-QB读取内部ROM存储器或内部闪存中的ID代码时输入要使用的ID代码（ID代码认证）（当连接MINICUBE时）。

对于没有ROM的产品或没有片上调试单元（OCD）的产品，不需要设置本区域。

输入一个20位（10字节）的十六进制数字作为ID代码（默认为所有数位均为F）。

ID代码被保存到寄存器。如果输入ID代码失败三次，ID78K0-QB被强制终止。

关于ID代码认证的详细信息，请参阅MINICUBE的用户手册。

(6) 周边中断

此区域用于指定在中断过程中是否停止仿真器的周边仿真功能。

中断	在中断过程中停止仿真器的周边仿真功能（当连接IECUBE时）。 在中断过程中，停止除了以下周边功能外的操作（当连接MINICUBE时）。 —8位计时器/事件计数器50。
非中断	在中断过程中，停止除了以下周边功能外的操作（默认）。 —监视计时器 —8位计时器H1 注： 只有在内部高速振荡器被选作运行时钟时，8位计时器H1才会停止运行；如果选择了内部高速振荡器以外的其他时钟，则8位计时器H1继续运行。

(7) 监视时钟

此区域用于指定在中断过程中是否将监视程序的运行时钟从子时钟切换到主时钟。

对于没有子时钟的产品，不需要设置本区域。

系统	切换运行时钟到主时钟，并执行监视程序（默认）。 注： 在ID78K0-QB中，时钟是通过操作PCC改动的，但不应在主时钟停止的时候进行。如果在中断过程中将运行时钟切换到主时钟，当执行返回到用户程序时，时钟被恢复到先前的设置。
用户	在执行监视程序的同时使用用户程序选择的时钟。

在中断过程中的运行时钟变化如下。变化取决于此区域中的设置（当连接MINICUBE时）。（用户：当出现中断时，监视程序运行并使用中断出现之前的时钟；主机：监视程序运行并使用MINICUBE提供的时钟）。

表6-6：在中断过程中的运行时钟（当连接MINICUBE时）

设备版本 ^{注1}	设置	当选择系统时		当选择用户时	
		闪存覆盖 ^{注2}	其他	闪存覆盖 ^{注2}	其他
2.00版或更新	周边	用户		用户	
	CPU	主机		主机	用户
迟于1.20版但早于2.00版	周边	主机		主机	用户
	CPU	主机		主机	用户
1.10版或更早	周边	主机		主机	
	CPU	主机		主机	

注1：“片上调试控制代码Vx.xx”中的Vx.xx可通过选择ID78K0-QB的[帮助（Help）菜单]的[关于...]查看，其显示设备的版本。

注2：设计闪存覆盖操作的处理，如下载数据并覆盖存储器。

（8）目标

此区域用于选择目标板是否连接到仿真器（当连接IECUBE时）。

备注：用于探测非法的供电状态。

默认为通过探测目标的电源（TVDD）确定。（参见“[表3-2：错误信息输出模式](#)”。）

Connect	连接
Not connect	不连接

（9）目标电源关闭

此区域用于选择是否电源关闭仿真功能是否有效（当连接MINICUBE时）。

在中断过程中不要关闭或开启电源。

在中断过程中目标系统的复位被屏蔽。

设置目标电源关闭	含义	功能
允许	启用电源关闭仿真功能	如果在用户程序运行的过程中开启目标系统的电源，用户程序在不产生中断的复位操作后立即重新执行。 如果在用户程序运行的过程中从目标系统输入复位，用户程序在不产生中断的复位操作后立即重新执行。
不允许	禁用电源关闭仿真功能	如果在用户程序运行的过程中开启目标系统的电源，会在复位操作后产生中断。 如果在用户程序运行的过程中从目标系统输入复位，会在复位操作后产生中断，然后调试器强制重新执行用户程序。

（10）失效保护中断

点击<详细信息...>按钮打开[失效保护对话框](#)，这样就可以单独设置失效保护功能（当连接IECUBE时）。

(11) 屏蔽

此区域用于屏蔽从目标发出的信号。

被屏蔽管脚的信号不输出到仿真器。

只有在目标系统的允许在调试阶段不稳定时屏蔽管脚。

注意1: 当连接MINICUBE时，只可以指定“TARGET RESET（目标复位）”和“INTERNAL RESET（内部复位）”。

注意2: 当连接IECUBE，在(8) 目标中指定“Not Connect（不连接）”时，“WAIT（等待）”和“INTERNAL RESET（内部复位）”被设置到选择的状态，不可改变屏蔽状态。

(12) 存储器映射

此区域用于通过指定存储器属性和地址设置映射。

(a) 存储器属性

可以选择以下映射属性。根据用途选择映射属性。（参见“5-2: 映射属性”。）

映射单位是1字节。

仿真ROM ^註	选择仿真器备用ROM（当连接IECUBE时）。 （最多可映射仿真ROM/目标的四个项目。）
目标 ^註	选择目标存储器。 （最多可映射仿真ROM/目标的四个项目。）
I/O保护 ^註	选择I/O输出保护区域。 （本区域只能在设为目标或外部SFR的区域中设置。）
堆栈	选择堆栈区域的存储器（当连接IECUBE时）。

注: 在使用没有外部空间的设备时无法选择此选项。

备注: 如果没有指定堆栈区域，默认指定整个内部高速RAM区域（除了寄存器区域）作为堆栈区域。

注意: 此区域作为I/O保护只有在记录到SFR窗口或监视窗口作为I/O端口后才可读。要读取此区域，强制在这些窗口读取。

(b) 映射地址

指定要映射的地址。

从键盘输入上部和下部地址。

(c) <添加>按钮、<删除>按钮

这些按钮用于设置和删除映射。

当点击<添加>按钮时，设置和显示指定为（a）存储器属性的映射。

功能按钮

（在ID78K0-QB启动后自动显示。）

确定	验证当前的环境。 设置环境并关闭此对话框。
取消	取消修改并关闭此对话框。
恢复	在此对话框打开前恢复先前的设置。
项目...	打开项目文件加载对话框。如果在打开或读取项目文件时出现错误，ID78K0-QB无法继续并将终止。
关于...	打开关于对话框。
帮助	显示此窗口的帮助窗口。

扩展选项对话框

此对话框用于显示和设置ID78K0-QB的扩展选项（参见“5.1 设置调试环境”）。

图6-7： 扩展选项对话框

（当连接IECUBE时）



（当连接MINICUBE时）



- 打开
- 对每个区域的解释
- 功能按钮

打开

选择[选项（Option）]菜单 -> [扩展选项...]

对每个区域的解释

(1) 使用MINICUBE的扩展功能

在使用MINICUBE的扩展功能时检查此项目（当连接MINICUBE时）。

如果选择此项目，RRM和DMM功能被开启（参见“5.12 伪实时RAM监视程序功能（读出时中断）”和“5.16 DMM功能”）。

如果选择此项目，(2) RAM监视程序被启用（默认选择此选项）。

(2) RAM监视程序

(a) 读出时中断

选择此项目，通过即时在用户程序执行中生成一个中断来指定RAM采样的目标范围。（参见“5.12 伪实时RAM监视程序功能（读出时中断）”）。

全部	整个存储器空间（当连接IECCUBE时） ^注 。 备注： 当打开大量窗口时，用户程序执行会被停止很长一段时间，因为存储器读取的范围会变大。
控制寄存器+SFR	控制寄存器和SFR区域（当连接IECCUBE时）。
RRM设置	在RRM对话框中的指定区域（当连接MINICUBE时）。
关闭	禁用伪实时RAM监视程序功能（读出时中断）。 备注： “表5-21实时RAM监视程序功能采样范围”中显示以外的存储器区域显示为“**”。

注：除“表5-21实时RAM监视程序功能采样范围”外。

(b) 刷新闻隔

指定RAM采样的采样时间（ms）。

可以指定采样时间为0到65500之间以100ms为单位的值。

如果指定为0，或者如果此区域为空，则数据不实时显示。

(3) 鼠标点击

本区域用于通过在源窗口或汇编窗口中点击鼠标按钮选择是否将软件断点或硬件断点设置为默认断点（如果在点标记区域设置了断点）。（参见“5.4.2 断点设置”。）

软件断点	设置一个软件断点（默认）。 断点的标记以蓝色显示。
硬件断点	设置一个硬件断点。 断点的标记以绿色显示（在执行前）。

(4) 在复位时清空寄存器

在CPU复位后选择此复选框将SP寄存器设置为0xFEE0（如果是MINICUDE的话为0xFED0）以及将所有的多用途寄存器（X、A、C、B、E、D、L、H）的存储器组设置为0x0。

在默认设置下，不清空寄存器。

(5) 在运行前清空跟踪存储器

选择此复选框在程序执行前清空跟踪存储器（当连接IECUBE时）。

(6) 中断声音

如果选择了此复选框，当出现中断时会发出哔哔声。

(7) 验证检查

此区域用于指定在数据已经写到存储器时是否进行验证检查。

验证检查在下载、存储器填充或存储器复制时进行。在[监视窗口](#)或[存储器窗口](#)中的变量或数据改变并写入存储器时也要进行验证检查。

注意：当连接MINICUBE时，不管本区域中的复选框是否选中，在写入到内部闪存（包括下载）的过程中不进行验证检查，但始终应进行闪存自写入的内部检查（不进行读取检查）。

功能按钮

确定	验证设置并关闭此对话框。
取消	取消修改并关闭此对话框。
恢复	在此对话框打开前恢复先前的设置。
帮助	显示此窗口的帮助窗口。

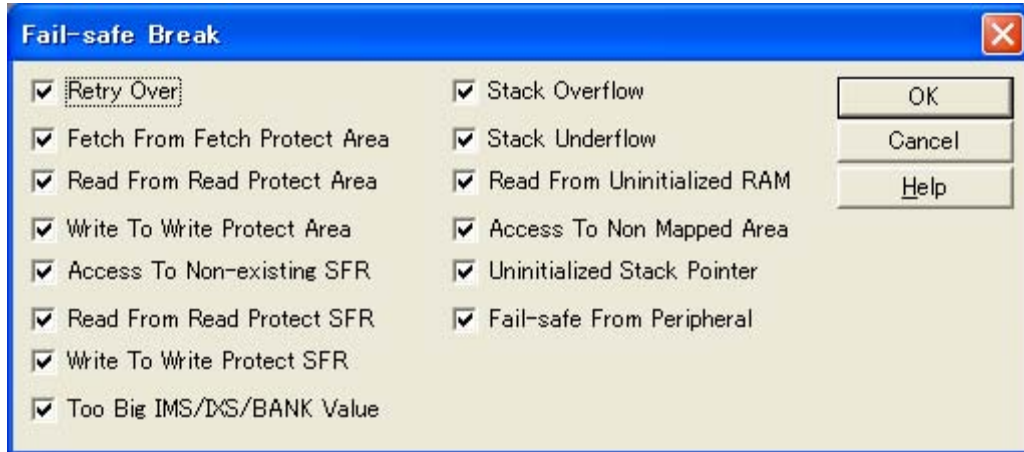
失效保护中断对话框

IECUBE

此对话框用于进行失效保护中断设置（参见“5.4.5 失效保护中断”）。

在读取一个项目文件时，读取的结果在此对话框中反映。

图6-8：失效保护中断对话框



—打开

—对每个区域的解释

—功能按钮

打开

点击配置对话框中的<详细信息...>按钮。

对每个区域的解释

(1) 失效保护中断设置区域

当选中一个复选框时，相应的失效保护中断功能被启用。在默认设置下，所有的复选框都被选中。

当连接MINICUBE时，只有“Uninitialized Stack Pointer（未初始化的堆栈指针）”始终启用。

Retry Over	超出了来自周边设备最大允许重试次数
Fetch From Fetch Protect Area	获取了禁止获取的区域
Read From Read Protect Area	读取了禁止读取的区域
Write To Write Protect Area	写入到禁止写入的区域
Access To None-existing SFR	访问了不存在的SFR
Read From Read Protect SFR	读取了禁止读取的SFR
Write To Write Protect SFR	写入到禁止写入的SFR
Too Big IMS/IXS/BANK Value	IMS（存储器大小切换寄存器）、IXS（内部扩展RAM大小切换寄存器）、存储器组设置错误
Stack Overflow	用户堆栈超过上限
Stack Underflow	用户堆栈未达到下限
Read From Uninitialized RAM	未进行RAM初始化
Access To Non Mapped Area	访问了未映射的其余
Uninitialized Stack Pointer	未进行堆栈指针初始化
Fail-safe From Peripheral	从周边设备失效保护。

功能按钮

确定	验证设置并关闭此对话框。
取消	关闭此对话框。
帮助	显示此窗口的在线帮助文件。

RRM 对话框

MINICUBE

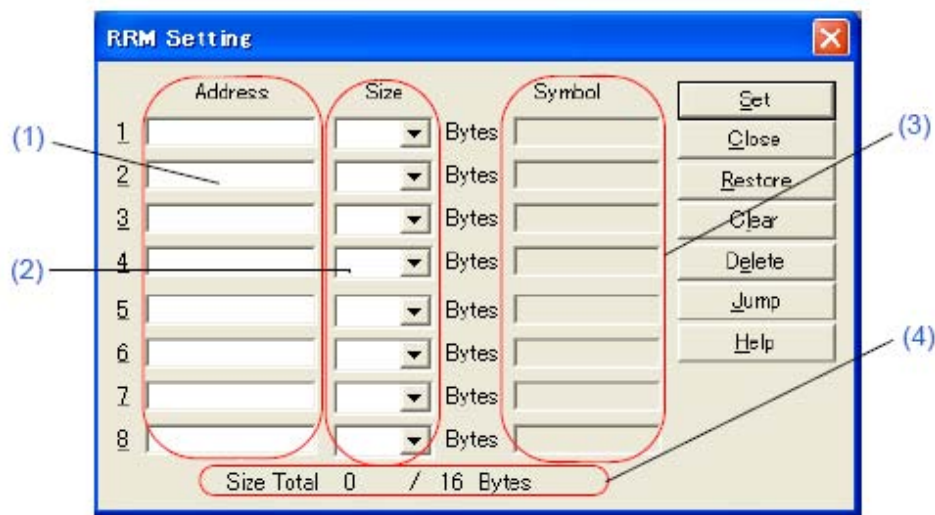
此对话框用于设定RRM功能的采样率。（参见“5.15 RRM功能”。）

最多可指定8个位置作为采样范围（单位为1字节）。

为8个位置指定的总大小不能超过16字节。

注意：本对话框只有在扩展对话框中的（1）使用MINICUBE扩展功能被选中时才能打开。

图6-9：RRM对话框



— 打开

— 对每个区域的解释

— 功能按钮

打开

本对话框打开时的设置根据打开方式的不同而变化。

(a) 当从RRM对话框进行设置时：

通过选择[选项（Option）]菜单 → [RRM设置...]打开本对话框。
在此情况下，（1）地址和（2）大小是手工输入的。

(b) 当从存储器窗口进行设置时：

通过打开存储器窗口，在窗口中选择一个地址，然后从右键菜单选择[RRM设置...]打开本对话框。

在此情况下，在（1）地址的一个空行中显示选择的地址，（2）大小中一个空行中显示“1”，（3）符号的一个空行中显示将地址转化为符号所获得的值。

备注: 如果地址域中为8个位置指定的总大小已经超过16字节, 或者已经对8个位置进行了设置, 则可以打开RRM设置对话框但无法设置数值。

(c) 当从[监视窗口](#)进行设置时:

通过打开[监视窗口](#), 在窗口中选择一个变量, 然后从右键菜单选择[RRM设置...]打开本对话框。

在此情况下, 在 (1) [地址](#) 的一个空行中显示将选择的变量转化为地址所获得的值, (2) [大小](#) 中一个空行中显示 “1”, (3) [符号](#) 的一个空行中显示将变量转化为符号所获得的值。

备注: 如果地址域中为8个位置指定的总大小已经超过16字节, 或者已经对8个位置进行了设置, 则可以打开RRM设置对话框但无法设置数值。

对每个区域的解释

(1) 地址

本区域用于指定RRM函数的采样起始地址。

输入数值的默认基数是十六进制。也可以用符号或表达式指定地址。(参见“[表5-6: 指定符号](#)”。)

在输入后, 点<确定>按钮启用设置。

(2) 大小

本区域用于指定从 (1) [地址](#) 开始的采样范围。

可以选择的值为1到16。

但为8个位置指定的总大小不能超过16字节。

(3) 符号

本区域用于显示 (1) [地址](#) 中指定的地址的符号。

指定的地址显示为一个符号或者一个符号+偏移量。

如果还没有设置地址, 则显示为空。

(4) 总大小

此区域显示的是 (2) [大小](#) 中指定的大小的总和。如果总和大小超过16字节, 则以红色显示。

功能按钮

设置	确定指定的采样范围。
关闭	关闭此对话框。
恢复	在此对话框打开前恢复先前的设置。
清除	清除当前的设置。
删除	删除设置中焦点所在的数字。
跳跃	打开 存储器窗口 并显示 (1) 地址 中有数字有焦点的地址。 跳跃针对活动状态的存储器窗口进行的。 如果要打开多个 存储器窗口 ，则必须在静止状态下设置。（参见“ 5-18: 活动状态和静止状态 ”。）
帮助	显示在线帮助文件。

屏蔽选项对话框

IECUBE

此对话框用于设置屏蔽选项和管脚模式（参见“5.1 设置调试环境”）。

如果是读取一个项目文件，则不必进行此项设置，读取文件的结果会反映在本对话框中（参见“5.17.1 调试环境（项目文件）”）。

备注：关于屏蔽选项和管脚模式的详细信息，请参见所使用的仿真器和仿真板的用户手册。

图6-10：屏蔽选项对话框



— 打开

— 对每个区域的解释

— 功能按钮

打开

通过选择[选项（Option）]菜单 -> [屏蔽选项...]

对每个区域的解释

（1）管脚组名称：

此区域用于选择要设置屏蔽选项和管脚模式的管脚组。

（2）选项名称：

此区域用于设置屏蔽选项和管脚模式。

设置项目根据“（1）管脚组名称：”中选择的管脚组的不同而变化。

显示为设置项目的第一个项目为当前设置。

功能按钮

确定	验证设置并关闭本对话框。
撤销	关闭此对话框。
设置	验证设置。可进行连续设置（对话框不关闭）。
恢复	在此对话框打开前恢复先前的设置。
帮助	显示在线帮助文件。

闪存选项对话框

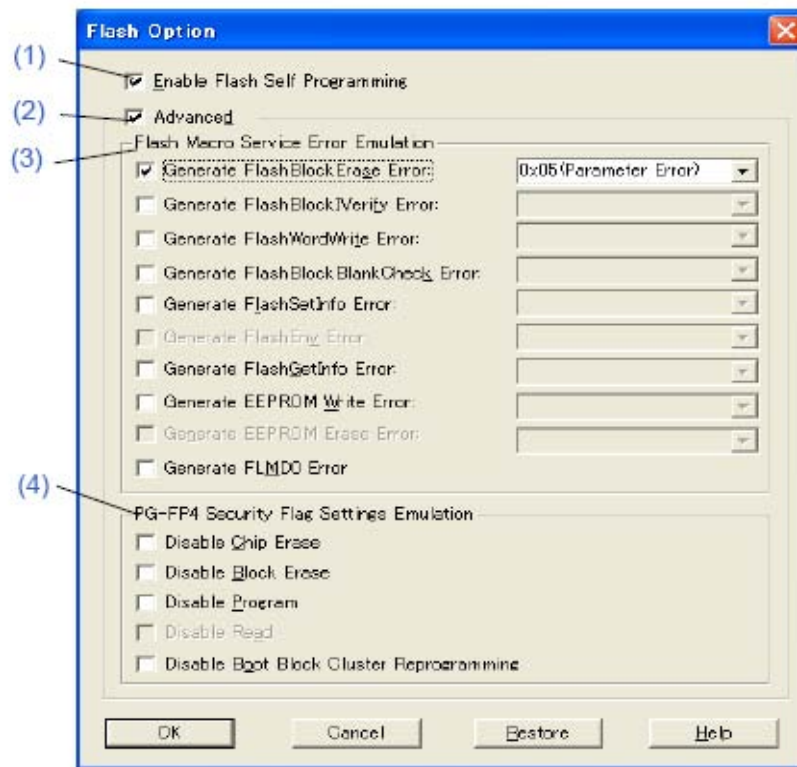
IECUBE

此对话框用于设置闪存自编程仿真。

此对话框可以在用户程序执行过程中打开。

注意： 在使用没有闪存的设备时不能打开此对话框。

图6-11：闪存选项对话框



— 打开

— 对每个区域的解释

— 功能按钮

打开

选择[选项 (Option)]菜单 -> [闪存选项...]

对每个区域的解释

(1) 启用闪存自编程:

如果选择了此项目，则启用闪存自仿真。

此项目会启用 (2) 高级的设置。在默认情况下不选择本项目。

(2) 高级

如果选择了此项目，则启用闪存自仿真的默认选项。

此项目会启用 (3) 闪存宏服务错误仿真和 (4) PG-FP4安全标记设置仿真的设置。在默认情况下不选择本项目。

(3) 闪存宏服务错误仿真

这些项目设置自由库功能的运行。通过构选项目，可以强制返回以下的错误值（在一般仿真情况下不返回）。

在相应的复选框被选中后，组合框会变为可用。在默认情况下不选择本项目。

可以在组合框中直接输入，但也可以选择以下的规定错误值。对每个存储器宏类型规定的错误值变化如下：

	CZ6系列 (Kx1+)	MF2系列 (Kx2)
产生FlashBlockErase错误	0x05 (参数错误) 0x1A (擦除错误)	0x05 (参数错误) 0x10 (保护错误) 0x1A (擦除错误) 0x1F (通过中断停止)
产生FlashBlockIVerify错误	0x05 (参数错误) 0x1B (内部检验错误)	0x05 (参数错误) 0x1B (内部检验错误) 0x1F (通过中断停止)
产生FlashWordWrite错误	0x05 (参数错误) 0x18 (FLMD错误) 0x1C (写错误)	0x05 (参数错误) 0x10 (保护错误) 0x1C (写错误) 0x1F (通过中断停止)
产生FlashBlockBlankCheck错误	0x05 (参数错误) 0x1B (空白错误)	0x05 (参数错误) 0x1B (空白检查错误) 0x1F (通过中断停止)
产生FlashSetInfo错误	0x05 (参数错误) 0x18 (FLMD错误) 0x1B (内部检验错误) 0x1C (写错误)	0x05 (参数错误) 0x10 (保护错误) 0x1A (擦除错误) 0x1B (内部检验错误) 0x1C (写错误) 0x1F (通过中断停止)
产生FlashEnd错误	0x05 (参数错误)	—
产生FlashGetInfo错误	0x05 (参数错误)	0x05 (参数错误) 0x20 (读错误)
产生EEPROM写错误	0x05 (参数错误) 0x18 (FLMD错误) 0x1C (写错误) 0x1D (MRG12错误) 0x1E (空白错误)	0x05 (参数错误) 0x10 (保护错误) 0x1C (写错误) 0x1D (MRG12错误) 0x1E (空白错误) 0x1F (通过中断停止)

	CZ6系列 (Kx1+)	MF2系列 (Kx2)
产生EEPROM擦除错误	0x05 (参数错误) 0x1A (擦除错误)	—
产生FLMD0错误	当目标连接时：不可用 (运行根据FLMD0管脚的状态而变化。)	
	当目标没有连接时： 如果选择，FLMD0在低层运行。 如果清除，FLMD0在高层运行	

(4) PG-FP4安全标记设置仿真

当使用闪存编程工具PG-FP4在闪存中设置安全标记时，对安全标记的初始值进行了仿真。

可设置的项目如下。所有的项目在默认情况下都是不选择的。

禁用芯片擦除	禁用/启用芯片擦除
禁用块擦除	禁用/启用块擦除
禁用编程	禁用/启用程序
禁用读取	禁用/启用读取 (不能选择)
禁用启动块簇重新编程 ^註	禁用/启用启动块簇重新编程

注：此项目仅对MF2固件可用。

功能按钮

确定	验证设置并关闭本对话框。
撤销	关闭此对话框。
恢复	在此对话框打开前恢复先前的设置。
帮助	显示在线帮助文件。

注意事项

(1) 在以下情况下的闪存自仿真。

—内部ROM大小被设为默认值。

变通办法：将默认大小设为内部ROM大小。

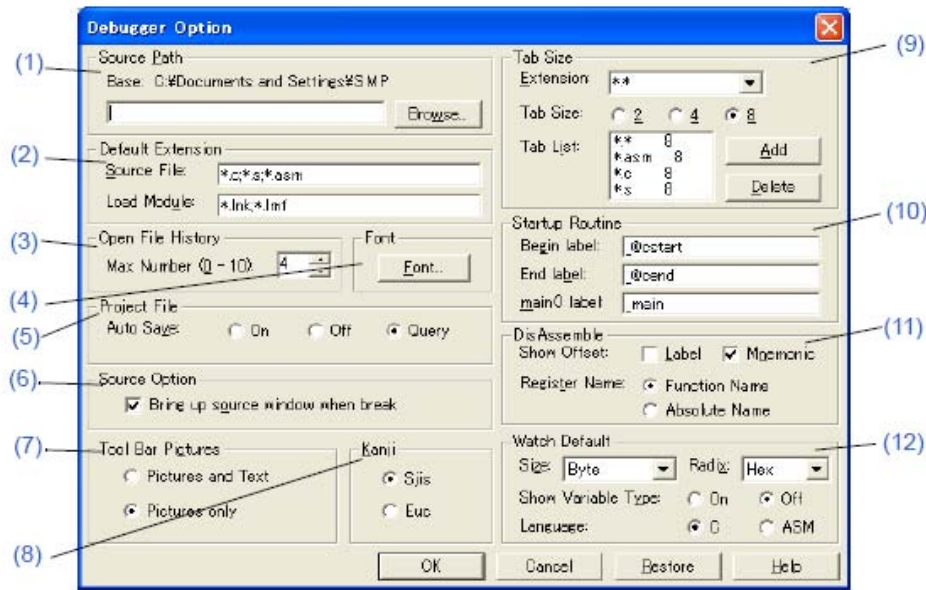
(2) 如果启用闪存自仿真，以下限制适用。

不能改变内部ROM的大小。

调试器选项对话框

此对话框用于显示和设置ID78K0-QB的各个选项。

图6-12： 调试器选项对话框



- 打开
- 对每个区域的解释
- 功能按钮

打开

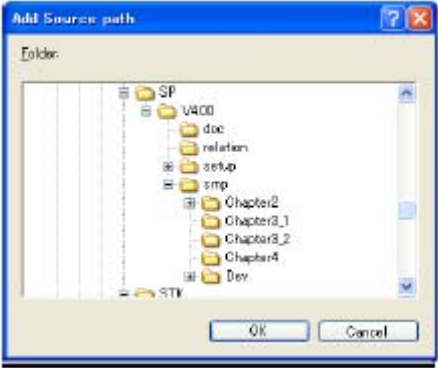
选择[选项（Option）]菜单 -> [调试器选项...]

对每个区域的解释

（1）源路径：

此其余用于指定要搜索源文件或文本文件的文件夹。

基础：	文件夹是显示的相对路径的基础。基础文件夹按以下顺序确定： 1) 项目文件加载到的文件夹 2) 负载模块或十六进制文件最近一次加载到的文件夹。 3) Windows的当前文件夹。
-----	---

文本框:	本区域用于指定要搜索的文件夹。 要指定一个文件夹，可以在文本框中直接输入文件夹或者点击<浏览...>按钮再选择文件夹。也可以指定相对路径。 点击<浏览...>按钮，打开[添加源路径]对话框。可用分号或冒号给路径分界。 图6-13: [添加源路径]对话框 
------	---

- 备注1:** 可以指定在源路径中包含“;”和/或“,”的目录，但不能指定不存在的目录。
- 备注2:** 在此对话框打开后，基础文件夹被立即选中和打开。如果选择的文件夹已经设为源路径，则不会添加源路径。
- 备注3:** 源路径的总长度可以包含4095个字符，包括扩展名前的点<.>。
- 如果用超过4096个字符指定源路径，则到4095个字符位置的有效路径作为源路径，而之后的字符被忽略。

(2) 默认扩展名

此区域用于指定默认扩展名。

用空格、冒号或逗号分隔扩展名。

源文件:	设置在选择[文件 (File)]菜单->[打开...]打开 浏览对话框 时显示的源文件的扩展名。 默认的扩展名是*.c、*.s、*.asm。
负载模块:	设置在打开 下载对话框 时显示的负载模块的扩展名。 默认的扩展名是*.lnk、*.imf。

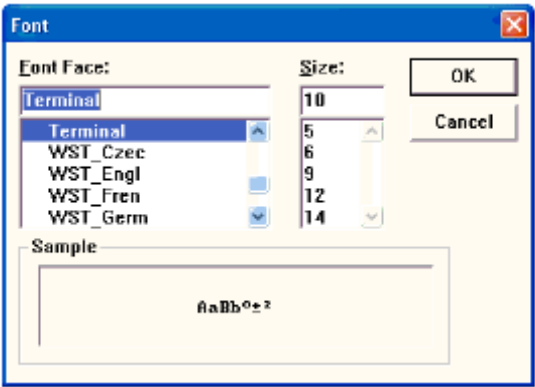
(3) 打开文件历史

此区域用于设定在[文件 (File)]菜单底部区域显示的历史文件的个数，默认值是4。如果设为0，则菜单中不显示历史文件。

(4) 字体

此区域用于指定源窗口、监视窗口、快速监视对话框、局部变量窗口、局部变量窗口和堆栈窗口中显示字体的颜色。点击<字体...>按钮打开[字体]对话框，可在对话框中设置显示的字体以及字体大小。

图6-14: [字体]对话框



(5) 项目文件

此区域用于设置项目文件的自动保存。（参见“5.17.1 调试环境（项目文件）”。）

自动保存:	设置在ID78K0-QB终止时项目文件是否自动保存。	
	开	自动保存项目文件。
	关	不自动保存项目文件。
	询问	在ID78K0-QB终止时显示退出调试器对话框。（默认）

(6) 源选项

此区域用于设置源窗口在中断时的运行。

通过选择此复选框，在中断时活动的源窗口会在前面显示。

如果没有活动的源窗口或者在负载模块文件中没有调试信息，活动的汇编窗口会在前面显示。

(7) 工具条图象

此区域用于设置工具条中显示的按钮。

图象和文字	显示带图象和文字的按钮。
仅图象	显示仅带图象的按钮。（默认）

(8) 汉字

无法在此区域中选择。

(9) Tab制表符大小

扩展名:	设置一个扩展名。用键盘输入一个扩展名或者从下拉菜单选择一个扩展名。
制表符大小:	选择制表符大小。选择一个制表符码显示多少个空格（2、4或8）。
制表符列表:	显示每个扩展名的制表符大小。
<添加>	要修改制表符大小的设置，选择“扩展名：”和“制表符大小：”，然后点击此按钮。
<删除>	要删除制表符大小的设置，选择“制表符列表：”，然后点击此按钮。

(10) 启动程序

此区域用于指定起始地址、结束地址以及显示移动程序文本区域的起始符号。

如果在下载对话框中下载了负载模块中的项目文件，即可打开源文件。（如果程序指针在此时定位在“起始标签：”和“结束标签：”之间的地址，则ID850QB显示从“main()标签：”开始的代码。）

起始标签:	指定起始地址的符号（默认：_@cstart）。
结束标签:	指定结束地址的符号（默认：_@cend）。
main()标签:	指定显示起始符号（默认：_main）。

注意1: 如果指定的符号不正确，直到程序指针到达相应源文件的地址范围后才能打开源文件。此外，无法通过单步执行跳过启动程序。

注意2: 一定要指定此区域。如果此区域空白，将无法关闭对话框。

(11) 反汇编

本区域用于设置反汇编显示。

显示偏移量:	指定是否在反汇编显示过程中显示偏移量（符号+偏移量）。 当显示偏移量时，只显示与数值匹配的符号（如果有的话）。如果没有发现匹配的符号，数值显示为不变的十六进制数字。	
	标签	指定是否在标签栏中显示偏移量。在默认情况下，偏移量是不显示的。
	助记	指定是否在助记栏中显示偏移量。在默认情况下，偏移量是显示的。

寄存器名称:	此区域用于选择在反汇编显示过程中以助记符显示寄存器名称的方法。	
	功能名称	显示寄存器名称为功能名称或别称（默认）。
	绝对名称	显示寄存器名称为绝对名称（默认）。

（12）监视默认值

此区域用于指定在[监视窗口](#)等中监视的符号。

大小:	如果在下拉菜单中指定了[自适应]，选择数据的默认显示大小。	
	字节（Byte）	以8比特为单位显示（默认）。
	词（Word）	以16比特为单位显示。
	双词（Double word）	以32比特为单位显示。
基数:	如果在下拉菜单中指定了[合适]，设置显示数据的默认基数。 在该表中选择的项目在监视数据的注脚中得到反映，如 监视窗口 中的数组变量（或标签）（默认为十六进制）。在此设置改变后，监视数据的注脚会被记录到 监视窗口 。	
	十六进制（Hex）	以十六进制数字显示。（默认）
	十进制（Dec）	以十进制数字显示。
	八进制（Oct）	以八进制数字显示。
	二进制（Bin）	以二进制数字显示。
	串（String）	以字符串显示。
显示变量类型:	选择显示/不显示指定的变量类型。	
	开	显示变量的类型。
	关	不显示变量的类型。（默认）
语言:	选择显示基数的语言。	
	C	显示像C语言一样的基数。（默认）
	ASM	显示像汇编语言一样的基数。

功能按钮

确定	验证设置并关闭本对话框。
撤销	撤销修改并关闭此对话框。
恢复	在此对话框打开前恢复先前的设置。
帮助	显示在线帮助文件。

伪仿真对话框

IECUBE

此对话框用于生成伪仿真。

由于生成的伪仿真的类型取决于设备文件，如果设备文件未包含相关的信息，则此对话框无法打开。

图6-15: 伪仿真对话框



- 打开
- 对每个区域的解释
- 功能按钮

打开

选择[运行 (Run)]菜单 -> [伪仿真...]

对每个区域的解释

(1) 命令:

可以通过点击带有要生成的伪仿真的名称的按钮生成目标伪仿真。按钮的数量和类型取决于设备文件。

不能点击没有伪仿真名称的按钮。

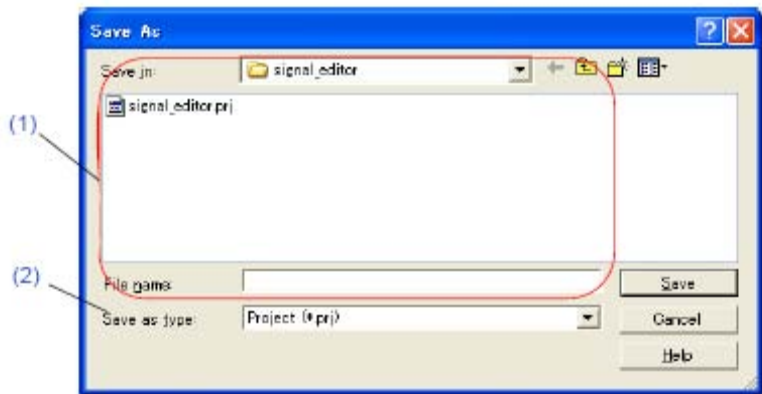
功能按钮

关闭	关闭此对话框。
帮助	显示在线帮助文件。

项目文件保存对话框

此对话框用于保存当前的调试环境到一个项目文件。（参见“5.17.1 调试器环境（项目文件）”。）在此对话框中可以将项目文件保存为新的文件或保存到已有的文件名下。

表6-16：项目文件保存对话框



- 打开
- 对每个区域的解释
- 功能按钮

打开

选择[文件（File）]菜单 -> [项目] -> [另存为...]。（如果要保存与先前加载或保存的项目文件相同名称的文件，则选择[文件（File）]菜单 -> [项目] -> [保存]。）

对每个区域的解释

（1）保存在（Save in）、文件名：

此区域用于指定文件名。可以直接输入文件名或者在上面区域的文件列表中选择文件名。指定的文件名最多可以包含257个字符，并可带有扩展名。

（2）文件类型：

此区域用于指定要保存的项目文件的扩展名（*.prj）。

如果忽略扩展名，会添加“*.prj”作为默认扩展名。

功能按钮

保存	保存当前的调试环境到一个选定的文件。保存后，此对话框关闭。
撤销	不保存文件而关闭此对话框。
帮助	显示在线帮助文件。

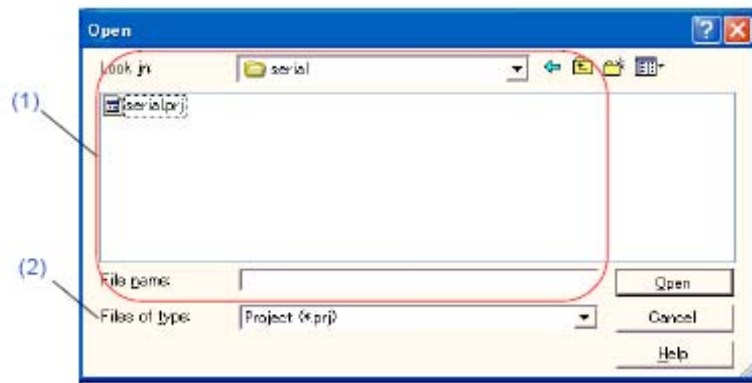
项目文件加载对话框

此对话框用于恢复保存到项目文件的调试环境。（参见“5.17.1 调试器环境（项目文件）”。）

如果在项目文件加载后有一个活动的源窗口，从窗口会显示在最上面。

注意：在ID78K0-QB启动后，如果在启动时加载了一个设置与目标设备设置不同的项目文件，则会使用在启动时指定的目标设备。

表6-17：项目文件加载对话框



— 打开

— 对每个区域的解释

— 功能按钮

打开



点击Proj按钮或选择[文件（File）]菜单 -> [项目] -> [打开...].

对每个区域的解释

（1）查看、文件名：

此区域用于指定要加载的文件名。可以直接输入文件名或者在上面区域的文件列表中选择文件名。指定的文件名最多可以包含257个字符，并可带有扩展名。

（2）文件类型：

此区域用于指定要保存的项目文件的扩展名（*.prj）。

功能按钮

打开	加载一个选定的文件。加载后，此对话框关闭。
撤销	不执行任何操作而关闭此对话框。
帮助	显示在线帮助文件。

下载对话框

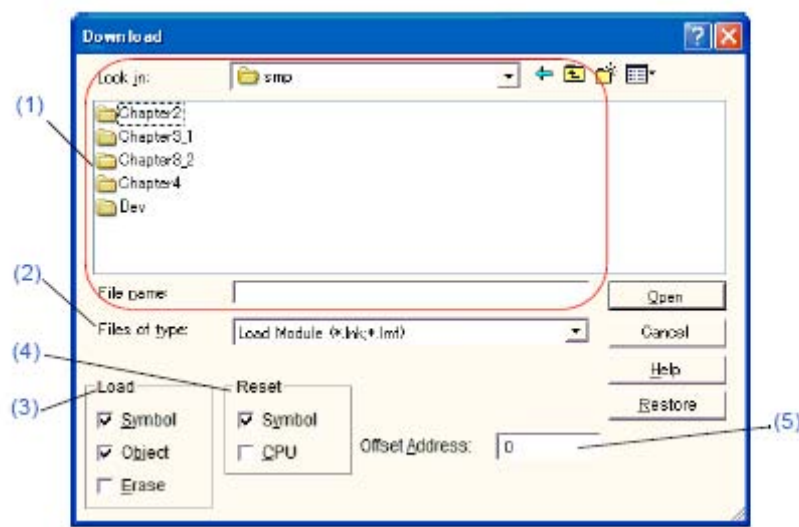
此对话框用于选择要下载文件的名称和格式，并将存储器内容下载到仿真器和目标系统。（参见“5.2 下载功能、上传功能”。）

如果已经下载了某个负载模块文件，则会搜索相应的源文件并自动打开源装口。

注意：如果加载一个负载模块文件，不能执行源的调试。

备注：关于在存储器组被使用时，下载十六进制格式或二进制数据格式文件后的文件分配图，请参见“（2）在存储器组被使用时的分配图象”。

表6-18：下载对话框



备注：以下对话框在下载时出现，可在任何时候调出下载。在下载完成后本对话框自动关闭。

- 打开
- 对每个区域的解释
- 功能按钮

打开

 点击Load按钮或选择[文件（File）]菜单 -> [下载...]。

对每个区域的解释

(1) 查看 (Look in)、文件名:

此区域用于指定要加载的文件名。可以直接输入文件名或者在上面区域的文件列表中选择文件名。指定的文件名最多可以包含257个字符，并可带有扩展名。

(2) 保存类型:

此区域用于指定要下载的文件类型（扩展名）。（参见“表5-3: 可以下载的文件类型”。）

备注: 这些是默认扩展名，也可以使用别的扩展名。也可在调试器选项对话框中指定显示的负载模块的默认扩展名。

(3) 加载

此区域用于设置加载条件。

符号	指定是否读取符号信息。如果指定了非负载模块格式的文件，则此复选框显示为灰色且不可用。
对象	指定是否读取对象信息（在选定时，默认为读取）。（当加载十六进制文件时，即使本按钮被清除，还是会读取对象信息。）如果指定了非负载模块格式的文件，则此复选框显示为灰色且不可用。
擦除	指定是否在下载前擦除内部闪存中的所有内容（在选定时，默认为擦除）（当连接MINICUBE时）。

(4) 复位

此区域用于设置复位条件。

如果指定了非负载模块格式的文件，则此区域显示为灰色且不可用。

符号	指定是否读取符号信息。
CPU	指定是否复位CPU（在选定时，默认为复位）。

(5) 偏移地址

本区域用于指定在加载文件时指定使用的偏移地址（对于二进制的文件，指定起始地址）。如果指定了非负载模块格式的文件或者代码覆盖结果文件，则此区域显示为灰色且不可用。可以用符号或表达式指定地址。（参见“表5-6: 指定符号”。）输入数值的默认基数是十六进制。

功能按钮

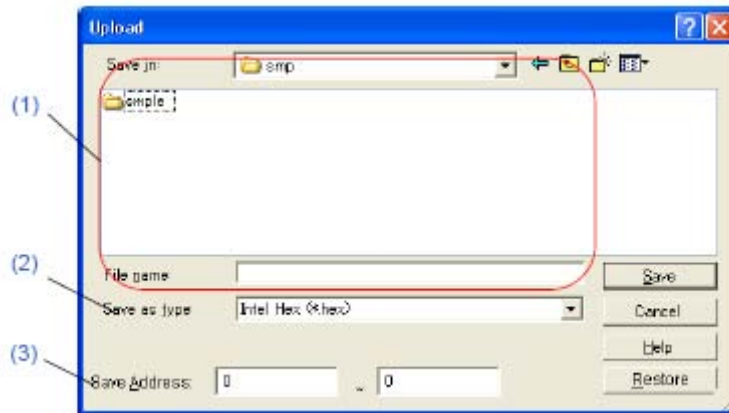
打开	加载一个选定的文件。加载后，此对话框关闭。
撤销	不执行任何操作而关闭此对话框。
帮助	显示在线帮助文件。
恢复	将输入数据恢复到原始状态。

上传对话框

此对话框用于设置要保存的文件的名称和格式，并将存储器内容保存到文件。（参见“5.2 下载功能、上传功能”。）

注意：关于在存储器组被使用时，在（2）保存类型：中选择了十六进制格式或二进制数据格式后的文件存储图象，请参见“5.2 下载功能、上传功能”。（当连接IECUBE时。）

图6-19：上传对话框



— 打开

— 对每个区域的解释

— 功能按钮

打开

选择[文件（File）]菜单 → [上传...]。

对每个区域的解释

（1）保存在、文件名：

此区域用于指定文件名。可以直接输入文件名或者在上面区域的文件列表中选择文件名。

指定的文件名最多可以包含257个字符，并可带有扩展名。

(2) 保存类型:

此区域用于指定要保存的项目文件的类型（扩展名）。

要保存的数据的格式由扩展名决定，（参见“表5-4: 可以上传的文件的类型”。）

备注: 也可以使用所列扩展名之外的扩展名。

(3) 保存地址:

此区域用于指定要保存的地址范围。

在 (2) 保存类型: 中选择了覆盖数据 (*.cvb) 时，所有的范围被保存起来（无法设置此区域）。

也可以用符号或表达式指定地址。（参见“表5-6: 指定符号”。）

输入数值的默认基数是十六进制。

备注: 如果保存文件指定的结束地址为0x10000或更高，指定下载为“十六进制格式[Bank] (*.hex;*.hxb;*.hxf)”或“二进制数据[Bank] (*.bin)”。

如果保存文件指定的结束地址低于0x10000，指定下载为“十六进制格式[64KB] (*.hex;*.hxb;*.hxf)”或“二进制数据[64KB] (*.bin)”（参见 (2) 在存储器组被使用时的存储图象）。

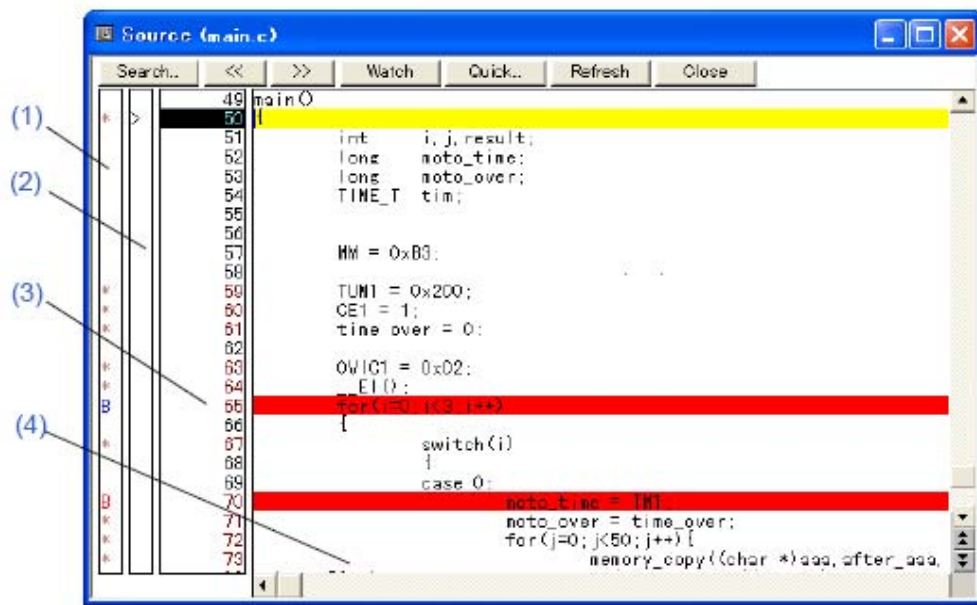
功能按钮

保存	根据设置保存文件。
撤销	不执行任何操作而关闭此对话框。
帮助	显示在线帮助文件。
恢复	恢复到此对话框打开前的状态。

源窗口

此窗口用于显示源文件或文本文件。（参见“5.3 源显示、反汇编显示功能”。）除了设置断点、显示执行覆盖测量的位置、混合显示模式（源窗口）外，在此窗口中还可以进行一系列使用右键菜单和功能按钮等的其他操作。此外，此窗口有两个状态：**活动状态**和**静止状态**。在窗口处于活动状态时，具有带联系窗口的跟踪结果（当连接IECUBE时）。另外，在本窗口中的选项带有**拖动功能**，可在另一个窗口中使用（参见“5.18 每个窗口共有的功能”）。

图6-20：源窗口



- 打开
- 对每个区域的解释
- [查看]菜单（源窗口专有的项目）
- 右键菜单
- 功能按钮
- 注意事项

打开

 点击Src按钮或者选择[浏览（Browse）]菜单 -> [源文本...].

（如果在下载了下载模块文件后相应的源文件存在，此窗口自动打开。）

对每个区域的解释

(1) 点标记区域

此区域用于显示事件设置状态（事件标记）、程序代码（*）显示和断点设置。

备注：只有在读取了负载模块文件下载的符号信息时才会显示程序代码。可以通过用鼠标点击此程序代码设置或删除断点。（如果某行未显示“*”，则在此行上方或下方处（显示有“*”者）设置断点。）

表6-7：事件设置状态（事件标记）

标记	含义
B（蓝色）	设置了软件断点。
B（红色）	设置了有效的硬件断点（在执行后）。
B（绿色）	设置了有效的硬件断点（在执行前）。 注： 对在执行前的中断设置优先级。
B（黑色）	设置了无效的硬件断点。 可在事件管理器上或中断对话框中验证硬件断点。
E	设置了事件条件。
L	设置了事件链接条件。（IECUBE）
T	设置了跟踪事件。（IECUBE）
Ti	设置了计时器事件。（IECUBE）
S	设置了截图事件。（IECUBE）
U	设置了存根事件。（IECUBE）
M	设置了DMM事件。（IECUBE）
A	设置了多个事件。

备注：如果设置了地址范围作为事件的地址条件，则显示的是范围下限的地址。此区域不反映地址条件的标记指定。

(2) 当前程序指针标记区域

在此区域显示的是表示当前程序指针值（程序指针寄存器值）的标记“>”。用鼠标点击此标记会出现显示程序指针寄存器值的弹出窗口。双击当前的程序指针标记区域可执行程序到指定的一行。（参见“[\[到这里\]](#)”。）

(3) 行号/地址显示区域

此区域显示一个源文件或文本文件的行号。

红色显示的是相应的程序代码存在的行号，黑色显示的是相应的程序代码不存在的行号。在混合显示模式（源窗口）中，反汇编显示地址以灰色显示。

此外，此区域基于代码覆盖测量信息突出显示执行过的地址（参见“5.11.3 覆盖测量执行的位置显示”）。

(4) 源文本显示区域

此区域显示源文本和文本文件。

黄色显示当前的程序指针行，红色显示设置有效断点的行。在混合显示模式（源窗口）中，源行以常规颜色显示。

此外，此区域还为指针指向的行（程序代码的起始行）和地址提供以下功能：

—[到这里]、[从这里开始]（参见“表5-9：执行的类型”）

—拖放功能

—右键菜单

注意：如果程序代码在源行不存在，则程序代码所在行上方或下方行的顶部地址由这些功能控制。

这些功能在以下情况下不能使用。此时相应的菜单会变灰并无法选择。

—如果显示是源文件以外的文件

—用户程序正在执行

[查看]菜单（源窗口专有的项目）

当源窗口处于活动状态时，在[查看]菜单中添加了以下项目。

创建中断事件	设置一个在选择的变量被访问时发生的中断事件。
在访问此变量时中断	设置一个在选择的变量被访问以进行读写时发生的中断事件。
在写到此变量时中断	设置一个在选择的变量被访问以进行写入时发生的中断事件。
在读取此变量时中断	设置一个在选择的变量被访问以进行读取时发生的中断事件。
清除	删除一个与选择的变量对应的中断事件。
事件信息	显示光标所在位置的行或者选定的变量名的事件信息。如果设置了事件，则会打开事件对话框。
混合	打开/关闭混合显示模式（源窗口）。

执行监视	进行关于代码覆盖显示的设置（当连接IECUBE时）。
清除	清除代码覆盖显示中的突出显示。
累积	在代码覆盖显示中累积突出显示。

右键菜单

移动...	移动显示位置。 打开 源文本移动对话框 。
混合	打开/关闭 混合显示模式（源窗口） 。
添加监视...	添加指定的数据到 监视窗口 。 打开 添加监视对话框 。
符号...	显示指定变量或函数的地址或指定符号的值。 打开 符号转地址对话框 。
在访问此变量时中断	设置一个在选择的变量被访问以进行读写时发生的中断事件。
在写到此变量时中断	设置一个在选择的变量被访问以进行写入时发生的中断事件。
在读取此变量时中断	设置一个在选择的变量被访问以进行读取时发生的中断事件。
清除	删除一个与选择的变量对应的中断事件。
事件信息	显示光标所在位置的行或者选定的变量名的事件信息。 如果设置了事件，则会打开 事件对话框 。
到这里	执行从当前程序指针到光标位置的程序。（参见“ 表5-7：中断类型 ”。）
改变程序指针	在光标位置到程序指针设置地址。
中断点	在光标位置设置或删除一个中断点。 备注： 对在执行前的中断（B）设置优先级。
软件中断点	在光标位置设置或删除一个软件中断点。
清除执行监视	清除代码覆盖显示中的突出显示（当连接IECUBE时）。
累积	在代码覆盖显示中累积突出显示（当连接IECUBE时）。
汇编	从光标位置的数据值指定的跳跃目标地址开始反汇编和显示。（参见“ 5.18.2 跳跃功能 ”。） 打开 汇编窗口 。 如果打开了一个活动的 汇编窗口 ，该窗口显示在最前面（以便可以对其进行操作）。
存储器	显示从光标位置的数据值指定的跳跃目标地址开始的存储器内容。（参见“ 5.18.2 跳跃功能 ”。） 打开 存储器窗口 。 如果打开了一个活动的 存储器窗口 ，该窗口显示在最前面（以便可以对其进行操作）。

功能按钮

搜索...	打开源搜索对话框并搜索源文件的字符串。如果在源文本显示区选择了一个字符串，则会打开源搜索对话框并搜索该字符串。 如果没有选择字符串，则会打开源搜索对话框但没有指定搜索的内容。 在源搜索对话框中指定搜索方法。 搜索的结果会在源窗口中突出显示。 本操作与选择[查看（View）]菜单→[搜索...]相同。
<<	从当前光标位置开始向前搜索（在屏幕中向上）满足源搜索对话框中设定条件的文本。 在搜索中本按钮会显示为<停止>。
>>	从当前光标位置开始向后搜索（在屏幕中向下）满足源搜索对话框中设定条件的文本。 在搜索中本按钮会显示为<停止>。
停止（在搜索中）	停止搜索。
监视	将源文本显示区域中选择的变量添加到监视窗口。 如果监视窗口未被打开，则会打开监视窗口。 如果在源文本显示区域中没有选择文本，只会打开监视窗口。 本操作与选择[查看（View）]菜单→[查看监视]相同。
快速...	临时显示快速监视对话框中源文本显示区域内选择的内容（如变量）。如果在源文本显示区域中没有选择文本，只会打开快速监视对话框。 本操作与选择[查看（View）]菜单→[快速监视...]相同。
刷新	用最新的数据刷新本窗口的内容。
关闭	关闭此窗口。

注意事项

- （1）如果程序代码由一个包含文件描述，而这些代码包含在多个文件中，则行编号和地址之间不为一一对应的关系。在这样一个包含文件中，显示行编号和地址之间的对应关系的函数运算不正确。
- （2）如果在负载模块文件下载后无法在源路径找到包含“main”函数的源文件，或者如果在单步执行过程中无法找到源文件，ID78K0-QB会打开一个选择源文件对话框并提示用户选择在对话框中显示的源文件的源路径。如果点击<取消>按钮，显示的文件名被记忆，ID78K0-QB一直到终止都不再询问源文件名。
- （3）最多可显示65535行的C语言和汇编语言源文件。如果源文件超过65535行，则分割源文件。
- （4）如果在源模式中进行单步执行，函数的开头（序言处理）不会被跳过。不能在序言处理中显示局部变量或堆栈跟踪以及返回执行。要跳过序言处理，则再次进行单步执行。

源搜索对话框

此对话框用于搜索源窗口中某个文件的内容。（参见“5.3.1 源显示”）。

通过设置每个项目然后点击<查找下一个>按钮即可开始搜索。通过点击<设置查找>按钮即可使用源窗口中的方向按钮（"<<"和">>"）进行搜索。

图6-21：源搜索对话框



- 打开
- 对每个区域的解释
- 功能按钮

打开

如果当前的窗口是源窗口，选择[查看（View）]菜单 -> [搜索...]或者在同一窗口点击<搜索...>按钮。

对每个区域的解释

（1）查找什么

此区域用于指定要搜索的数据（最多为256个字符）。
在默认情况下，显示的是窗口中选择的调用此对话框的串。同时也可以修改显示的字符串。
最多可记录16条输入记录。

（2）匹配大小写

可选择此选项区分大小写。

（3）方向

此区域用于指定搜索的方向。

向上	向前搜索。从当前光标位置向前（在屏幕中向上）搜索数据。
向下	向后搜索。从当前光标位置向后（在屏幕中向下）搜索数据。（默认）

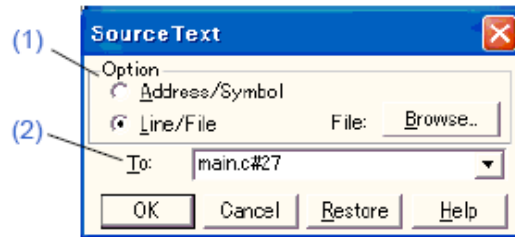
功能按钮

查找下一个	根据给定的条件搜索指定的数据。 如果找到了指定的字符串作为搜索的结果，该字符串会被突出显示。要继续搜索，再次点击此按钮。
设置查找	设置指定的条件作为搜索条件并关闭此对话框。
停止（在搜索中）	停止搜索。
关闭	关闭此对话框。 （在搜索过程中，此按钮被<停止>按钮替换。）
帮助	显示在线帮助文件。

源文本移动对话框

此对话框用于指定在源窗口中显示的文件以及文件开始显示的位置。（参见“5.3.1 源显示”。）

图6-22：源文本移动对话框



— 打开

— 对每个区域的解释

— 功能按钮

打开

如果当前的窗口是源窗口，选择[查看（View）]菜单 → [移动...]

对每个区域的解释

（1）选项

此区域用于在指定显示起始位置时选择输入模式。

（a）地址/符号

可选择此选项以通过地址（或符号）来指定。

（b）行/文件

可选择此选项以通过行号（或文件名）来指定。要搜索文件名，使用<浏览...>按钮。

（2）到：

此区域用于指定要显示的文件名或地址。

最多可记录16条输入记录。

—当选择“地址/符号”时：

指定开始显示的地址。

输入数值的默认基数是十六进制。也可以用符号或表达式指定地址。（参见“表5-6：指定符号”。）

点击<确定>按钮显示源文本即可查看与指定地址值对应的源行。

—当选择“行/文件”时：

指定开始显示的行号（或文件名）。

行号的指定规则：**[[路径名](#)][文件名](#)][#行号](#)]**。

输入数值的默认基数是十六进制

可以仅用文件名指定文件名，也可使用绝对路径和相对路径。

如果指定了文件名或者相对路径，将会搜索[调试器对话框](#)中指定的源路径中的文件。

点击<确定>按钮可显示指定的行号为第一行的文件。

当文件名被省略时，将从指定行开始显示当前显示的文件；如果行号被省略时，从第一行开始显示该文件。

功能按钮

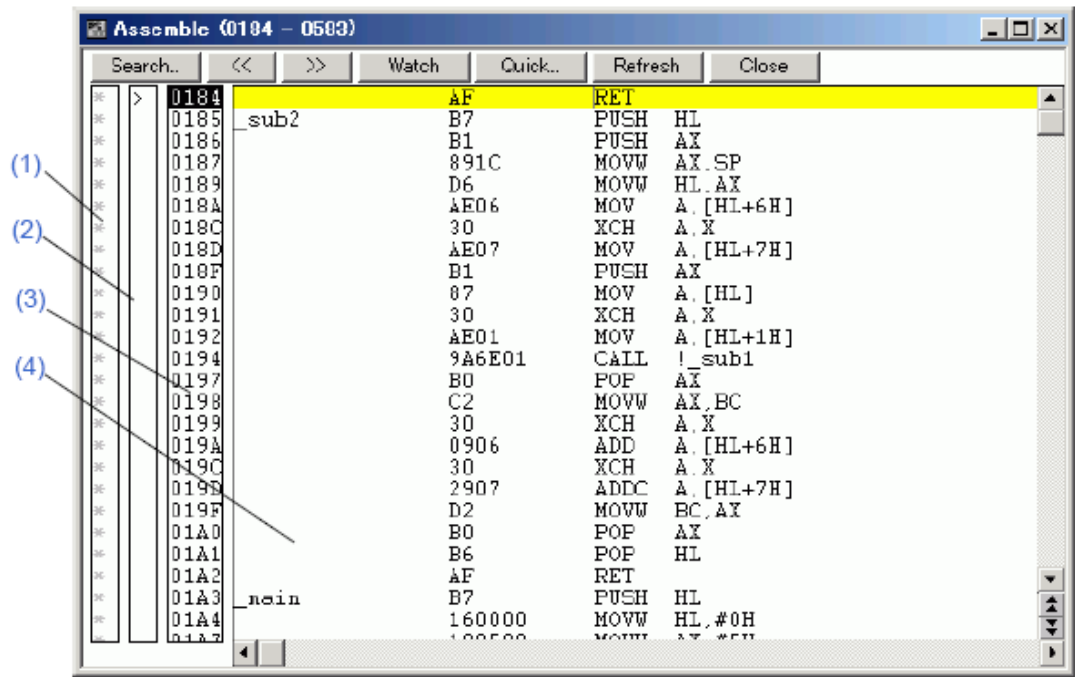
确定	从指定的位置开始显示源文件。
撤销	关闭此对话框。
恢复	恢复到此对话框打开前的状态。
帮助	显示在线帮助文件。

汇编窗口

此窗口用于反汇编和显示程序以及执行行汇编。（参见“5.3 源显示、反汇编显示功能”。）行汇编的结果也反映在存储器窗口中。

除了设置断点、显示执行覆盖测量的位置、混合显示模式（源窗口）外，在此窗口中还可以进行一系列使用右键菜单和功能按钮等的其他操作。此外，此窗口有两个状态：活动状态和静止状态。在窗口处于活动状态时，具有带联系窗口的跟踪结果（当连接IECUBE时）。另外，在本窗口中的选项带有拖放功能，可在另一个窗口中使用（参见“5.18 每个窗口共有的功能”）。

图6-23： 汇编窗口



- 打开
- 对每个区域的解释
- [查看]菜单（汇编窗口专有的项目）
- 右键菜单
- 功能按钮
- 相关操作
- 注意事项

打开

 点击Asm按钮或者选择[浏览（Browse）]菜单 -> [汇编]。

对每个区域的解释

(1) 点标记区域

此区域用于显示事件设置状态（事件标记）和断点设置。

(2) 当前程序指针标记区域

在此区域显示的是表示当前程序指针值（程序指针寄存器值）的标记“>”。用鼠标点击此标记会出现显示程序指针寄存器值的弹出窗口。双击当前的程序指针标记区域可执行程序到指定的一行。（参见“[\[到这里\]](#)”。）

(3) 地址指定区域

此区域显示反汇编开始的位置。

此外，此区域基于代码覆盖测量信息突出显示执行过的地址（参见“[5.11.3 覆盖测量执行的位置显示](#)”）。

注意：结束地址不显示。如果存储器组功能没有使用，结束地址为0xFFFF。

(4) 反汇编显示区域

此区域显示地址的标签和代码数据以及反汇编的助记符。

黄色显示当前的程序指针行，红色显示设置有效断点的行。

在助记栏中可以是行汇编。

此区域还提供以下功能：

- [\[到这里\]](#)、[\[从这里开始\]](#)（参见“[表5-9：执行的类型](#)”）
- [拖放功能](#)
- [右键菜单](#)

[查看]菜单（汇编窗口专有的项目）

当汇编窗口处于活动状态时，在[\[查看\]](#)菜单中添加了以下项目。

事件信息	显示光标所在位置的地址的事件信息。如果设置了事件，则会打开 事件对话框 。（当连接IECUBE时。）
执行监视	进行关于代码覆盖显示的设置（当连接IECUBE时）。
清除	清除代码覆盖显示中的突出显示。
累积	在代码覆盖显示中累积突出显示。

右键菜单

菜单选项对选定的行或项目有效而非鼠标指针点击的位置（与选择相同名称的主菜单操作相同）。

移动...	移动显示位置。 打开 地址文本移动对话框 。
添加监视...	添加指定的数据到 监视窗口 。 打开 添加监视对话框 。
符号...	显示指定变量或函数的地址或指定符号的值。 打开符号转地址对话框。
到这里	执行从当前程序指针到光标位置的程序。（参见“表5-7：中断类型”。）
改变程序指针	在光标位置到程序指针设置地址。
中断点	在光标位置设置或删除一个中断点。 备注： 对在执行前的中断（ B ）设置优先级。
软件中断点	在光标位置设置或删除一个软件中断点。
清除执行监视	清除代码覆盖显示中的突出显示（当连接IECUBE时）。
累积	在代码覆盖显示中累积突出显示（当连接IECUBE时）。
源文本	使用光标位置的数据值作为跳跃目标地址，显示相应的源文本和源行。（参见“5.18.2 跳跃功能”。） 如果在跳跃目标地址不存在行信息，则无法进行跳跃。 打开 源窗口 。 如果打开了一个活动的 源窗口 ，该窗口显示在最前面（以便可以对其进行操作）。
存储器	显示从光标位置的数据值指定的跳跃目标地址开始的存储器内容。（参见“5.18.2 跳跃功能”。） 打开 存储器窗口 。 如果打开了一个活动的 存储器窗口 ，该窗口显示在最前面（以便可以对其进行操作）。

功能按钮

搜索...	打开 汇编搜索对话框 并搜索助记符的字符串。 在 汇编搜索对话框 中指定搜索方法。 搜索的结果会在汇编窗口中突出显示。 本操作与选择[查看（View）]菜单→[搜索...]相同。
<<	从当前光标位置开始向前搜索（在屏幕中向上）满足 汇编搜索对话框 中设定条件的文本。 在搜索中本按钮会显示为<停止>。
>>	从当前光标位置开始向后搜索（在屏幕中向下）满足 汇编搜索对话框 中设定条件的文本。 在搜索中本按钮会显示为<停止>。

停止（在搜索中）	停止搜索。
监视	将（4）反汇编文本显示区域中选择的变量添加到监视窗口。 如果监视窗口未被打开，则会打开监视窗口。 如果在（4）反汇编文本显示区域中没有选择文本，只会打开监视窗口。 本操作与选择[查看（View）]菜单→[查看监视]相同。
快速...	临时显示快速监视对话框中（4）反汇编显示区域内选择的内容（如符号）。如果在汇编显示区域中没有选择文本，只会打开快速监视对话框。 本操作与选择[查看（View）]菜单→[快速监视...]相同。
刷新	用最新的数据刷新本窗口的内容。
关闭	关闭此窗口。

相关操作

（1）行汇编

要修改反汇编的内容，将光标移到助记栏（按Insert键切换覆盖和插入模式）。

如果在修改了助记栏中的反汇编内容后试图将光标移动到另一行，则会检查新的内容。如果新的内容是合法的，修改内容的一行上的代码数据会被显示为“*”

按Enter键将助记栏中修改的内容写入到存储器。再按Enter键检查新的内容。如果即使有一行非法，新内容不写入存储器。要放弃内容，按ESC键。

如果内容正确并且按了Enter键，内容被写入到存储器，然后光标移到助记栏的下一行，这样便可以修改下一行的数据。

注意：如果修改后新指令的字节数小于旧指令的字节数，会插入尽量多有必要的空转（NOP）指令。如果修改后新指令的字节数大于旧指令的字节数，会覆盖下一个指令。在此情况下，也会插入尽量多有必要的空转指令。这同样适用于跨源行的指令。

注意事项

- (1) 存储器组区域（0x8000到0xbfff）的符号名称（由赋到公共区域的指令助记符表示）是显示的BANK寄存器指示的存储器组的符号名。
- (2) 当程序指针指向非源行地址的一个地址时，无法进行返回执行（[运行]菜单->[返回退出]）。此外，不能连续进行返回执行，因为程序指针在返回执行后不再指向源行的地址。因此，要再次进行返回执行，应在源模式中进行单步执行，使得程序指针指向源行的地址。
- (3) 如果试图将汇编窗口在反汇编显示的起始地址向上方滚动，则显示视图可能不全（这是因为调试器无法定位反汇编显示的起始地址）。

在此情况下，点击<刷新>按钮；此时显示在顶部的地址会被作为起始地址，然后刷新反汇编显示。

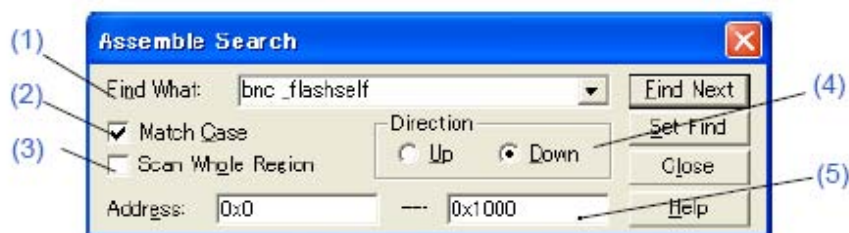
汇编搜索对话框

此对话框用于搜索汇编窗口中的内容。（参见5.3.2“反汇编显示”。）

在一个输入字符串和反汇编字符串中包含的连续字符串被比作一个空字符。

设置每个项目后按<查找下一个>按钮后即可进行开始搜索。通过点击<设置查找>按钮即可使用汇编窗口中的方向按钮("<<"和">>")进行搜索。

图6-24: 汇编搜索对话框



—打开

—对每个区域的解释

—功能按钮

打开

如果当前的窗口是汇编窗口，选择[查看（View）]菜单 -> [搜索...]或者在同一窗口点击<搜索...>按钮。

对每个区域的解释

（1）查找什么

此区域用于指定要搜索的数据（最多为256个字符）。

在默认情况下，显示的是窗口中选择的调用此对话框的串。同时也可以修改显示的字符串。最多可记录16条输入记录。

（2）匹配大小写

可选择此选项区分大小写。

（3）扫描整个区域

选择此选项可扫描整个指定的区域。

(4) 方向

此区域用于指定搜索的方向。

向上	向前搜索。从当前光标位置向前（在屏幕中向上）搜索数据。
向下	向后搜索。从当前光标位置向后（在屏幕中向下）搜索数据。（默认）

(5) 地址

此区域用于指定要搜索的地址。

输入数值的默认基数是十六进制。也可以用符号或表达式指定地址。（参见“表5-6：指定符号”。）

功能按钮

查找下一个	根据给定的条件搜索指定的数据。 如果找到了指定的字符串作为搜索的结果，该字符串会被突出显示。要继续搜索，再次点击此按钮。
设置查找	设置指定的条件作为搜索条件并关闭此对话框。
停止（在搜索中）	停止搜索。
关闭	关闭此对话框。 （在搜索过程中，此按钮被<停止>按钮替换。）
帮助	显示在线帮助文件。

地址移动对话框

此对话框用于指定以下窗口显示的起始地址：

- 存储器窗口
- 汇编窗口
- SFR窗口

注意：当一个存储器组被使用时，无法对非存储器组区域指定0x10000或更高的地址空间。（参见图6-6：内部ROM存储器组被使用时的地址空间图（存储器组ROM大小为40KB）。）

图6-25：地址移动对话框（例：当存储器窗口打开时）



- 打开
- 对每个区域的解释
- 功能按钮

打开

如果当前的窗口是目标窗口，选择[查看（View）]菜单 -> [移动...]

对每个区域的解释

（1）到：

此区域用于指定地址。在默认条件下，显示的是调用此对话框的窗口中选择的串或者当前的程序指针值。在必要时可以修改显示的字符串。输入数值的默认基数是十六进制。也可以用符号或表达式指定地址。（参见“表5-6：指定符号”。）最多可记录16条输入记录。

功能按钮

确定	从指定的位置开始显示源文件。
撤销	关闭此对话框。
恢复	恢复到此对话框打开前的状态。
帮助	显示在线帮助文件。

符号转地址对话框

此对话框用于显示指定的变量或函数的地址或者显示指定符号的值。（参见“5.3.4 转化符号（符号转地址）”。）

图6-26：符号转地址对话框



- 打开
- 对每个区域的解释
- 功能按钮

打开

选择[查看（View）]菜单 -> [符号...]

对每个区域的解释

（1）符号：

此区域用于指定要转化的变量、函数名、符号名或行号。（参见“表5-6：指定符号”。）

输入数值的默认基数是十六进制。最多可记录16条输入记录。

要修改此区域的内容，点击<确定>按钮。转化结果会显示在（2）转化结果显示区。

（2）转化结果显示区

显示（1）符号：中指定的变量、函数地址、符号值、行号地址和表达式值，还显示I/O端口名或SFR名称的地址值、寄存器名的内容以及PSW标记名的标记值。

如果指定了bit符号，符号会被转成地址.bit格式。同时，不能指定包含bit符号的等式。

(3) 基数

此区域用于选择在 (2) 转化结果显示区中显示的数据的基数。

Hex	十六进制数字
Dec	十进制数字
Oct	八进制数字
Bin	二进制数字

功能按钮

确定	如果已经对 (1) 符号: 的内容进行了修改, 则转化此符号。如果没有对 (1) 符号: 的内容进行修改, 则在转化后关闭此对话框。
关闭	关闭此对话框。
恢复	恢复输入数据到原始状态。 如果已经点击了<确定>按钮, 则恢复数据到刚按下<确定>按钮后的状态。
帮助	显示在线帮助文件。

监视窗口

此窗口用于显示和修改指定的监视数据。（参见“5.6 监视功能”。）

此窗口可像存储器窗口一样的方式实时显示广范围的监视数据（如整体变量和公用符号），即使是在程序执行过程中亦可显示。

此窗口中数据更新和重写的结果将在存储器窗口中反映。

通过在源窗口或汇编窗口中点击<监视...>按钮记录监视数据。（参见“5.6.3 记录和删除监视数据”。）

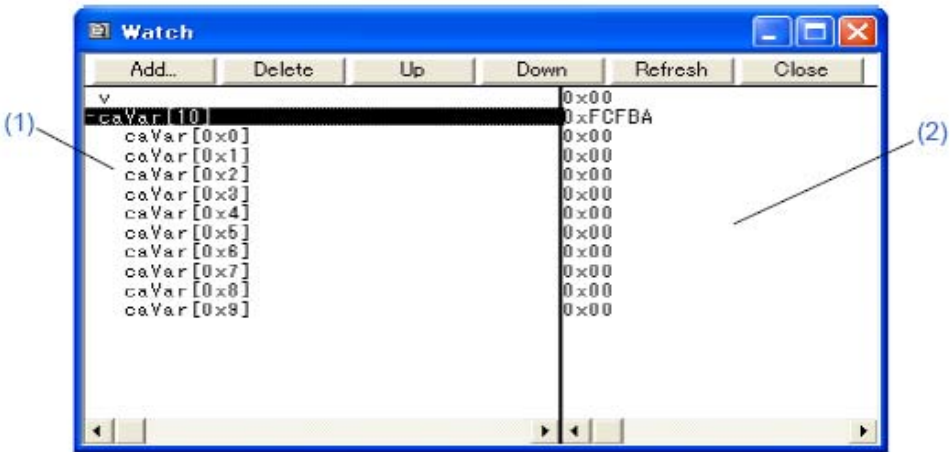
此窗口允许通过右键菜单轻松地设置断点到变量。

备注1: 如果一个局部变量和一个整体变量同名，则局部变量优先。

备注2: 在监视窗口中最多可显示10000行。

注意: 当连接MINICUBE时，在用户程序执行的过程中不能对寄存器或SFR进行DMM（写）操作。

图6-27：监视窗口



- 打开
- 对每个区域的解释
- [查看]菜单（监视窗口专有的项目）
- 右键菜单
- 功能按钮

打开

 点击Wch按钮或者选择[浏览（Browse）]菜单 -> [监视]。

对每个区域的解释

(1) 符号名称显示区

此区域用于显示变量名称、符号名称和类型以及结构或联集的标签名称。
“+”是显示的数组、指针变量、结构或联集的前缀。在双击变量时展开和显示如下：

图6-8：监视窗口显示格式（符号）

第一个符号	含义	
+	数组、指针变量或结构 可双击“+”展开显示（第一个符号由“+”变为“-”）。	
	数组	双击“+”可根据数组变量的类型显示变量的所有元素。
	指针变量	双击“+”可显示指针所示的所有数据。
	结构/联集	双击“+”可根据成员变量的类型显示结构/联集的所有成员。 如果结构或联集中定义了结构或联集，则显示内部结构或联集的名称。 也可以使用“+”展开内部结构或联集。
-	展开显示的变量。 可双击“-”取消展开显示（第一个符号由“-”变为“+”）。	

备注：如果一个数组的变量太多导致无法展开，则会显示警告信息。

可以在通过选择要修改的项目然后选择 **右键菜单**—>[修改监视...]打开的 **修改监视对话框** 中修改记录的监视数据。不能删除带有展开层级（如数组的元素和结构和联集的成员）的行。

如果对监视窗口中的变量或符号设置了一个访问断点，则符号名称显示区域以金色突出显示。

(2) 数据值显示/设置区

此区域用于显示和修改监视数据值。数据值在执行停止时被更新。

要保存一个数值，选择[文件]菜单—>[另存为...]。如果获取数据失败，本区域显示空白。

数值是通过直接输入修改的。要修改的区域以 **红色** 显示，当按下 **Enter** 键时修改的内容被写入目标存储器。可以按 **ESC** 键取消先前的数值。

注意：如果试图在用户程序执行的过程中修改数据值，可以不造成临时中断的符号^{#1}会被写入，而需要临时中断才能被写入的符号^{#2}，会出现一个对话框提示要发生临时中断，然后生成一个临时中断并写入符号。对于即使生成临时中断也无法写入的符号^{#3}，则发生错误且符号无法写入。

注1：分配到内部高速RAM或通用寄存器的变量（当连接IECUBE时）。

注2：控制寄存器、SFR等（当连接IECUBE时）。

分配到内部高速RAM或通用寄存器的变量（当连接MINICUBE并启用扩展功能时）。

注3：分配到内部高速RAM或通用寄存器的变量（当连接MINICUBE且禁用MINICUBE的扩展功能时）。

显示格式如下：

表6-9：监视窗口显示格式（数据）

显示数据	内容
整数	十六进制（0xxxxx）或（xxxxH） 十进制（xxxx）或（xxxxT） 八进制（0xxxxx）或（xxxxQ） 二进制（0bxxxxx）或（xxxxY）
字符	“字符”
枚举型	成员名称
如果指定范围	根据指定的范围显示
浮点型	支持单精度和双精度。 输入/显示格式如下： [+ -] inf [+ -] nan [+ -] integer e [+ -] exponent [+ -] integer.fraction[e [+ -] exponent
"?"	由于范围或优化编译的变化而失效的数据。

备注：可以在每个变量的右键菜单修改数值的基数。“整数”的显示格式可以在调试选项对话框中修改。

[查看]菜单（源窗口专有的项目）

当本窗口是当前窗口时，在[\[查看\]菜单](#)中添加了以下项目。

只能对选中的项目进行操作。

创建中断事件	使用如下选择的项目创建一个中断事件。
在访问此变量时中断	设置一个在变量被访问以进行读写时发生的中断事件。
在写到此变量时中断	设置一个在变量被访问以进行写入时发生的中断事件。
在读取此变量时中断	设置一个在变量被访问以进行读取时发生的中断事件。
清除	删除一个与选择的项目对应的中断事件。
事件信息	显示光标选定的变量的事件信息。如果设置了事件，则会打开 事件对话框 。
二进制（Bin）	显示二进制数字。
八进制（Oct）	显示八进制数字。
十进制（Dec）	显示十进制数字。
十六进制（Hex）	显示十六进制数字。
串（String）	显示字符串。
合适	显示每个变量的默认值。 根据 调试器选项对话框 的设置显示符号（默认）。
字节（Byte）	以8比特为单位显示。
词（Word）	以16比特为单位显示。
双词（Double word）	以32比特为单位显示。
自适应	显示每个变量的默认值（默认）。 此项目进对C语言的符号可用。 根据 调试器选项对话框 的设置显示汇编语言的符号。
上	上移一行。
下	下移一行。
强制读取	强制读取由于值会被修改而禁用的SFR、I/O端口的数据和 添加I/O端口对话框 中添加的I/O保护区。

右键菜单

菜单选项对选定的行或项目有效而非鼠标指针点击的位置（与选择相同名称的主菜单操作相同）。

在访问此变量时中断	使用选择的项目创建一个在变量被访问以进行读写时发生的中断事件。
-----------	---------------------------------

在写到此变量时中断	使用选择的项目创建一个在变量被访问以进行写入时发生的中断事件。
在读取此变量时中断	使用选择的项目创建一个在变量被访问以进行读取时发生的中断事件。
清除	删除一个与选择的项目对应的中断事件。
RRM设置...	设置RRM功能的采样范围。 打开 RRM设置对话框 （当MINICUBE连接时）。
事件信息	显示光标选定的变量的事件信息。如果设置了事件，则会打开 事件对话框 。
修改监视...	修改选定的监视数据。 打开 修改监视对话框 。
删除监视	从窗口中删除选定的监视数据。
二进制（Bin）	用二进制数字显示选择的行。
八进制（Oct）	用八进制数字显示选择的行。
十进制（Dec）	用十进制数字显示选择的行。
十六进制（Hex）	用十六进制数字显示选择的行。
串（String）	用字符串显示选择的行。
合适	以每个变量的默认值显示选择的行。 根据 调试器选项对话框 的设置显示符号（默认）。
字节（Byte）	以8比特为单位显示选择的行。
词（Word）	以16比特为单位显示选择的行。
双词（Double word）	以32比特为单位显示选择的行。
自适应	显示选择的行为每个变量的默认值（默认）。 此项目进对C语言的符号可用。 根据调试器选项对话框的设置显示汇编语言的符号。
上	将选择的行上移一行。
下	将选择的行下移一行。

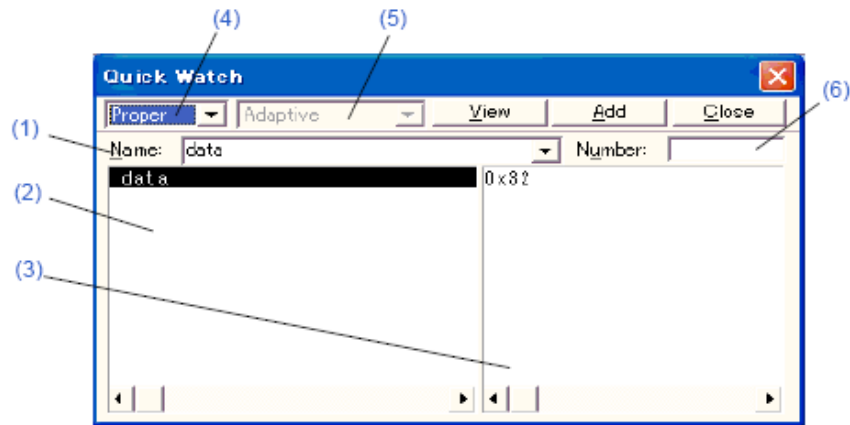
功能按钮

添加...	打开 添加监视对话框 。 如果在添加监视对话框中指定了监视数据并点击了 添加监视对话框 ，则指定的监视数据被添加到监视窗口。
删除	从窗口删除选择的监视数据。
上	将选择的行上移一行。
下	将选择的行下移一行。
刷新	用最新的监视数据更新窗口的内容。
关闭	关闭此窗口。

快速监视对话框

此对话框用于临时显示或修改指定的监视数据。（参见“5.6 监视功能”。）

图6-28：快速监视对话框



— 打开

— 对每个区域的解释

— 功能按钮

打开

如果当前的窗口是源窗口或汇编窗口，选择[查看（View）]菜单 → [快速查看...]或者点击同一窗口中的<快速...>按钮。

对每个区域的解释

（1）名称：

此区域用于指定要显示的监视数据。

在默认情况下，显示的是窗口中选择的调用此对话框的串。必要时也可以修改显示的字符串。最多可记录16条输入记录。

如果修改了此区域的内容，则可以通过点击<查看>按钮显示指定的数据。

（2）符号名称显示区

此区域用于显示监视数据（变量名、符号名称和类型、结构或联集标签名）。（参见监视窗口中的“（1）符号名称显示区”。）此区域无法编辑。

（3）数据值显示/设置区

此区域用于显示和修改数据值。（参见监视窗口中的“（2）符号名称显示区”。）

(4) 显示基数选择区

此区域用于选择显示的基数。

合适	变量：显示每个变量的默认值。 符号：以 调试器选项对话框 中设置的基数显示数据。
二进制（Bin）	以二进制数字显示。
八进制（Oct）	以八进制数字显示。
十进制（Dec）	以十进制数字显示。
十六进制（Hex）	以十六进制数字显示。
串（String）	显示为字符串。

(5) 显示大小选择区

此区域用于选择显示大小。

如果确定了显示大小（如当显示C语言或寄存器中的变量时）后，就不能再修改了。

自适应	变量：显示每个变量的默认值。 符号：以 调试器选项对话框 中设置的大小显示数据。
字节（Byte）	以8比特为单位显示。
词（Word）	以16比特为单位显示。
双词（Double word）	以32比特为单位显示。

(6) 数量

此区域用于指定要显示数据的数量（空白或者是1到256之间的数字）。

如果此区域空白，数据显示为一个简单的变量。如果指定数量为1或者以上，在[监视窗口](#)中显示的数据是一个数组变量。

如果显示数组变量，数据前显示“+”。双击此“+”，数据的所有元素会根据数据的类型展开显示。（展开的数据前显示“-”。双击此“-”可取消展开显示。）

如果确定了要显示数据的数量（如当显示C语言或寄存器中的变量时），指定的数量就不能修改了。。

功能按钮

查看	在下面的域中显示 (1) 名称: 中指定的数据。
添加	将 (1) 名称: 中指定的数据添加到监视窗口。
关闭	关闭此对话框。 没有实际写入到目标内存的数据会被取消。

添加监视对话框

此对话框用于记录在监视窗口显示的监视数据。（参见“5.6 监视功能”。）

在同一符号下可以记录多个数据。

图6-29：添加监视对话框



— 打开

— 对每个区域的解释

— 功能按钮

打开

选择[查看（View）]菜单 → [添加监视...]或者点击[监视窗口](#)中的<添加...>按钮。

对每个区域的解释

（1）名称：

此区域用于指定要添加到[监视窗口](#)的符号。

在默认情况下，显示的是窗口中选择的调用此对话框的串。

必要时也可以修改显示的字符串。

如果没有选择字符串，本区域为空白。最多可记录16条输入历史。

输入格式如下：

表6-10: 监视窗口输入格式

— C语言的变量名	
变量表达式 : 变量名	
变量表达式 [定植 变量名]	数组元素
变量表达式 . 变量名	结构/联集的实体元素
变量表达式-> 变量名	指针指示的结构/联集元素
*变量表达式	指针变量的值
&变量表达式	变量所在的地址
— 寄存器名	
— SFR名、SFR比特名	
— 标签、EQU和直接值的地址	
— 寄存器名. bit	
— SFR名. bit	
— 标签名. bit、EQU标记. bit和直接值的地址. bit	
— 比特符号	
— 范围的指定	

当指定范围时是变量如何处理如下:

表6-11: 当指定范围时是变量如何处理

范围指定	程序名	文件名	函数名	变量名
prog\$file#func#var	Prog	file	func	var
prog\$file#var	Prog	file	global	var
prog\$func#var	Prog	global	func	var
prog\$var	Prog	global	global	var
file#func#var	Current	file	func	var
file#var	Current	file	global	var
func#var	Current	current	func	var
var	Current	current	current	var

(2) 基数

此区域用于选择显示的数据基数。（参见[快速监视对话框](#)中的“（4）显示基数选择区”。）

(3) 大小

此区域用于选择显示的大小。（参见[快速监视对话框](#)中的“（5）显示大小选择区”。）

(4) 数量

此区域用于指定要显示的数据的数量。（参见[快速监视对话框](#)中的“（6）数量”。）

功能按钮

添加	添加指定数据到 监视窗口 。 此对话框保持打开。
确定	添加指定数据到 监视窗口 。 关闭此对话框。
取消	关闭此对话框。
恢复	恢复到此对话框打开前的状态。
帮助	显示在线帮助文件。

修改监视对话框

此对话框用于修改监视窗口中选定一行上的数据。（参见“5.6 监视功能”。）

不能修改具有展开层级（如数组的元素和结构和联集的成员）的行。

在修改监视数据后，选定行的内容被新的数据取代。

即使修改后的符号名与现有的数据重复，仍然可以修改符号名。

表6-11：修改监视对话框



- 打开
- 对每个区域的解释
- 功能按钮

打开

如果当前窗口是监视窗口，选择[查看（View）]菜单 -> [修改监视...]

对每个区域的解释

（1）名称：

此区域用于修改监视窗口中选定一行上的符号。（参见添加监视对话框中的“（1）名称：”。）

（2）基数：

此区域用于修改监视窗口中选定一行上的显示基数。（参见快速监视对话框中的“（4）显示基数选择区：”。）

（3）大小

此区域用于修改监视窗口中选定一行上的显示大小。（参见快速监视对话框中的“（5）显示大小选择区：”。）

（4）数量

此区域用于修改监视窗口中选定一行上显示的数据数量。（参见快速监视对话框中的“（6）数量：”。）

功能按钮

添加	不能选择。
确定	用指定的数据替换 监视窗口 中选定的一行上的数据，然后关闭此对话框。
取消	关闭此对话框。
恢复	恢复到此对话框打开前的状态。
帮助	显示在线帮助文件。

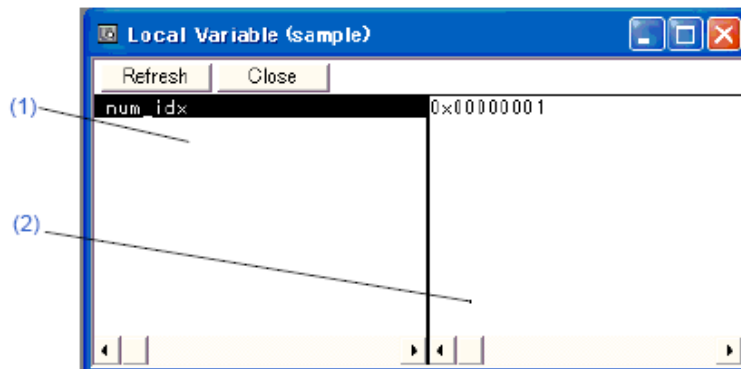
局部变量窗口

本窗口用于显示当前函数中的局部变量以及修改局部变量值。（参见“5.6 监视功能”。）

本窗口与堆栈窗口的跳转功能相连接，并在跳转到源窗口时显示跳转的函数中的局部变量。

可以在本窗口中进行一系列使用右键菜单和功能键等的其他操作。

图6-31：本地变量窗口



— 打开

— 对每个区域的解释

— 右键菜单

— 功能按钮

打开



点击Loc按钮或者选择[浏览 (Browse)]菜单 -> [局部变量...].

对每个区域的解释

(1) 局部变量名显示区

此区域显示局部变量名。（参见监视窗口中的“（1）符号名显示区”。）

可显示自动、内部静态和寄存器变量。不能对本区域进行编辑。

(2) 局部变量值显示/设置区

此区域用于显示和修改局部变量值。（参见监视窗口中的“（2）数据值显示区”。）可以通过直接输入修改数值。修改的位置以红色显示，按Enter键修改的内容写入到目标内存。但在用户程序执行的过程中无法写入修改（如果试图这样做则会发生错误）。可以按ESC键取消先前的数值。

[查看]菜单（局部变量窗口专有的项目）

当本窗口是当前窗口时，在[\[查看\]菜单](#)中添加了以下项目。

二进制（Bin）	显示二进制数字。
八进制（Oct）	显示八进制数字。
十进制（Dec）	显示十进制数字。
十六进制（Hex）	显示十六进制数字。
串	显示字符串。
合适	显示每个变量的默认值（默认）。

右键菜单

菜单选项对选定的行或项目有效而非鼠标指针点击的位置（与选择相同名称的主菜单操作相同）。

添加监视	打开 添加监视对话框 。
二进制（Bin）	用二进制数字显示选择的行。
八进制（Oct）	用八进制数字显示选择的行。
十进制（Dec）	用十进制数字显示选择的行。
十六进制（Hex）	用十六进制数字显示选择的行。
串	用字符串显示选择的行。
合适	以每个变量的默认值显示选择的行。 根据 调试器选项对话框 的设置显示符号（默认）。

功能按钮

刷新	用最新的监视数据更新本窗口的内容。
取消	关闭此对话框。

堆栈窗口

本窗口用于显示和修改用户程序的当前堆栈内容。（参见“5.6.7 堆栈跟踪显示功能”。）

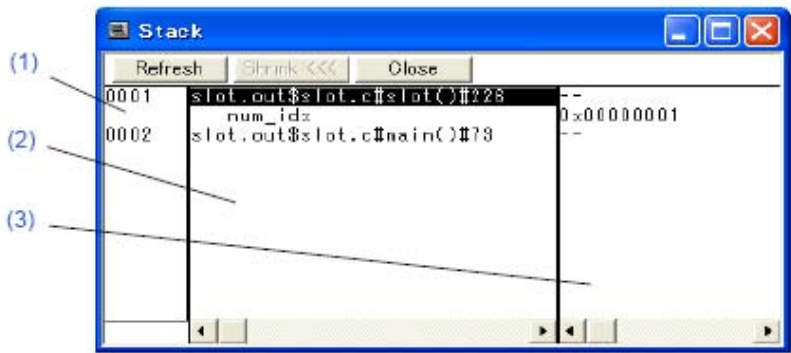
可以使用跳转功能跳转到与堆栈内容相应的窗口。

可以在本窗口中进行一系列使用右键菜单和功能按钮等的其他操作。

注意： 当有一个函数（noauto、norec函数等）没有将帧指针（HL）推到堆栈上，则堆栈跟踪显示功能会无法正常运行。

备注： 在函数的序曲处理活尾声处理过程中可能会显示[错误]。

图6-32：堆栈窗口



—打开

—对每个区域的解释

—[查看]菜单（堆栈窗口专有的项目）

—右键菜单

—功能按钮

打开



点击Stk按钮或者选择[浏览（Browse）]菜单 -> [堆栈跟踪]。

对每个区域的解释

(1) 堆栈帧编号显示区

此区域对堆栈内容设定编号并显示堆栈内容。

堆栈帧编号是从1开始的自然数。堆栈的嵌套越浅，编号数字越大。这就意味着堆栈编号大于另一个函数的某个函数是调用该函数的。

(2) 堆栈帧内容显示区

此区域显示堆栈帧内容，包括函数名和局部变量名。注意，本区域无法编辑。

(a) 如果堆栈内容包含一个函数，则如下：

[程序名\$文件名#函数名(参数列表)#行编号]

如果双击此行，操作与[跳转功能](#)的跳到源窗口相同（即程序执行已经跳转到的函数中的局部变量将显示在[局部变量窗口](#)中）。如果函数有一个局部变量，该局部变量将显示在下一行和后面的行中。

(b) 如果堆栈内容包含一个局部变量

则显示其类型和名称。（参见“[表6-8：监视窗口显示格式（符号）](#)”。）

注意，内部静态变量和寄存器变量是不显示的。

(3) 堆栈内容显示/设置区

此区域用于显示和修改堆栈内容。

(a) 如果堆栈内容包含一个函数，则显示“—”，不能修改函数。

(b) 如果堆栈内容包含一个局部变量

则显示其类型和名称。（参见“[表6-8：监视窗口显示格式（数据）](#)”。）

可以通过直接输入修改数值。修改的位置以红色显示，按Enter键修改的内容写入到目标内存。但在用户程序执行的过程中无法写入修改（如果试图这样做则会发生错误）。可以按ESC键取消先前的数值。

[查看]菜单（堆栈窗口专有的项目）

当本窗口是当前窗口时，在[\[查看\]菜单](#)中添加了以下项目。

二进制（Bin）	显示二进制数字。
八进制（Oct）	显示八进制数字。
十进制（Dec）	显示十进制数字。

十六进制 (Hex)	显示十六进制数字。
串	显示字符串。
合适	显示每个变量的默认值 (默认)。

右键菜单

二进制 (Bin)	用二进制数字显示选择的行。
八进制 (Oct)	用八进制数字显示选择的行。
十进制 (Dec)	用十进制数字显示选择的行。
十六进制 (Hex)	用十六进制数字显示选择的行。
串	用字符串显示选择的行。
合适	以每个变量的默认值显示选择的行。 根据 调试器选项对话框 的设置显示符号 (默认)。
源文本	显示从当前光标位置的数据值指定的跳转目标地址开始的相应源文本和源行。(参见“ 5.18.2 跳转功能 ”。) 如果跳转目标地址中没有行信息, 您无法跳转。 打开 源窗口 。 如果打开了活动的 源窗口 , 则该窗口会显示在最前面 (这样就可以对窗口进行操作了)。
汇编	从当前光标位置的数据值指定的跳转目标地址开始反汇编和显示。(参见“ 5.18.2 跳转功能 ”。) 打开 汇编窗口 。 如果打开了活动的 汇编窗口 , 则该窗口会显示在最前面 (这样就可以对窗口进行操作了)。
内存	显示从当前光标位置的数据值指定的跳转目标地址开始的内存内容。(参见“ 5.18.2 跳转功能 ”。) 打开 内存窗口 。 如果打开了活动的 内存窗口 , 则该窗口会显示在最前面 (这样就可以对窗口进行操作了)。

功能按钮

刷新	用最新的监视数据更新本窗口的内容。
收缩 <<<	折叠选中函数的局部变量表。
展开 >>> (在点击了<收缩 <<<>按钮时)	显示选中函数的局部变量表。
取消	关闭此对话框。

内存窗口

本窗口用于显示和修改内存内容。（参见“5.7 内存处理功能”。）可以在本窗口中进行一系列使用右键菜单和功能按钮等的其他操作。

此外，此窗口有两个状态：活动状态和静止状态。在窗口处于活动状态时，具有带联系窗口的跟踪结果（当连接IECUBE时）。（参见“5.18 每个窗口共有的功能”。）

备注1: 可以使用不同的颜色显示内存存取状态（读、写、读和写）（参见“5.7.3 存取监视功能（当连接IECUBE时）”。）

备注2: 本窗口打开时的显示起始位置如下啊：

- 第一次：从RAM区域的第一个地址开始显示。
- 第二次和以后：从活动状态的窗口关闭的地址开始显示。（如果活动状态的窗口从未被关闭，从第一次显示的起始位置开始显示。）

图6-23：内存窗口

(1) 打开

(2) 对每个区域的解释

(3) [查看]菜单（内存窗口专有的项目）

— 右键菜单

— 功能按钮

打开

 点击Mem按钮或者选择[浏览（Browse）]菜单 -> [内存]。

对每个区域的解释

（1）地址：

此区域显示内存地址。

注意：在内存组被使用时地址宽度会发生变化。

（2）+0 +1 +2....

此区域用于显示和修改内存的内容和显示存取状态（参见“5.7.3 存取监视功能（当连接IECUBE时）”。）

显示		当显示信息保存到视图文件中 时被替代的符号	含义
	绿色	R	读
	红色	W	写
	蓝色	A	读和写

内存内容是通过直接输入修改的。

要修改的位置以红色显示，当按下Enter键时修改的内容被写入目标内存。可以按ESC键取消先前的数值。一次最多可指定256个字节。

备注：要在用户程序执行的过程中修改内存内容，点击<DMM...>按钮打开DMM对话框。

（3）0 1 2 3....

此区域用于显示和修改ASCII字符的内存内容。

在选择[视图]菜单->[ASCII]时显示本区域。

可以像修改内存显示区一样修改本区域中的数据。

修改方法与（2）+0 +1 +2....相同。

备注：当显示地址被改动时，ASCII显示中的光标位置不会同步改变。

[查看]菜单（内存窗口专有的项目）

当本窗口是当前窗口时，在[\[查看\]菜单](#)中添加了以下项目。

二进制（Bin）	显示二进制数字。
八进制（Oct）	显示八进制数字。
十进制（Dec）	显示十进制数字。
十六进制（Hex）	显示十六进制数字。
半字节	以4比特为单位显示。
字节	以8比特为单位显示。（默认）
词	以16比特为单位显示。
双词	以32比特为单位显示。
ASCII	选择是否显示ASCII符号。 选定：显示 清除：不显示（默认）
小字节序	以小字节序显示。（默认）
大字节序	以大字节序显示。
存取监视	此区域用于进行存取监视功能的设置（当连接IECUBE时）。
清除	通过存取监视功能清除显示颜色。
累积	启用/禁用存取状态（内存内容改变）的累积显示。 选定：累积显示内存内容的改变。 清除：仅显示上次更新后的内存内容改变。

右键菜单

菜单选项对选定的行或项目有效而非鼠标指针点击的位置（与选择相同名称的主菜单操作相同）。

移动...	移动显示位置。 打开 地址移动对话框 。
RRM设置...	打开 RRM对话框 。（MINICUBE）
二进制（Bin）	显示二进制数字。
八进制（Oct）	显示八进制数字。
十进制（Dec）	显示十进制数字。
十六进制（Hex）	显示十六进制数字。（默认）
半字节	以4比特为单位显示。
字节	以8比特为单位显示。（默认）
词	以16比特为单位显示。
双词	以32比特为单位显示。

ASCII	选择是否显示ASCII符号。 选定：显示 清除：不显示（默认）
清除存取监视	通过存取监视功能清除显示颜色（当连接IECUBE时）。
累积	启用/禁用存取状态（内存内容改变）的累积显示。 选定：累积显示内存内容的改变。 清除：仅显示上次更新后的内存内容改变。

功能按钮

搜索...	打开 内存搜索对话框 并从显示的内存内容搜索字符串或者搜索内存内容。 选定的数据（一个内存值）在 内存搜索对话框 中显示作为要搜索的数据。如果打开内存搜索对话框而不指定数据，可以从键盘指定数据。 搜索的结果会在内存窗口中突出显示。
<<	从当前光标位置开始向前搜索（在屏幕中向上）满足 内存搜索对话框 中设定条件的内存内容。 在搜索中本按钮会显示为<停止>。
>>	从当前光标位置开始向后搜索（在屏幕中向下）满足内存搜索对话框中设定条件的内存内容。 在搜索中本按钮会显示为<停止>。
停止（在搜索中）	停止搜索。
刷新	用最新的数据更新本窗口的内容。
DMM...	打开 DMM对话框 。（在清除扩展选项对话框中的（1）使用MINICUBE扩展功能时不能选择（当连接MINICUBE时）。）
关闭	关闭此窗口。

内存搜索对话框

本对话框用于搜索内存窗口中光标所在部分的内容。（参见“5.7 内存处理功能”。）

如果光标在内存窗口中的(2)+0+1+2....中，则指定的数据被作为二进制数据串；如果光标在内存窗口中的(3)0 1 2 3....中，则指定的数据被作为ASCII字符串，并搜索这些区域的内容。

通过设置每个项目然后点击<查找下一个>按钮即可开始搜索。通过点击<设置查找>按钮即可使用内存窗口中的方向按钮（“<<”和“>>”）进行搜索。

备注1：不能搜索非映射、SFR和I/O保护区。

备注2：如果内存组被使用，对于非内存组区域，无法指定0x10000或以上的地址空间的地址。（参见“图6-6：在内部ROM内存组没有使用时的地址空间图（内存组ROM大小为40KB）”。）

图6-34：内存搜索对话框



— 打开

— 对每个区域的解释

— 功能按钮

打开

如果当前窗口是内存窗口，选择[查看（View）]菜单 → [搜索...]或者点击同一窗口中的<搜索...>按钮。

对每个区域的解释

（1）查找什么：

本区域用于搜索数据。

在默认情况下，显示的是窗口中选择的调用此对话框的串。在必要时也可以修改显示的字符串。最多可记录16条输入历史。

- (a) 当在 (2) +0 +1 +2....中搜索时：
最多可以指定16个数据项。用“空白符”分隔每个数据。
- (b) 当在 (3) 0 1 2 3....中搜索时：
最多可以指定256个字符。数据中的“空白符”被视为空白符。

(2) 单位：

本区域用于指定在 (2) +0 +1 +2....中搜索的数据的比特数：

字节	以8比特数据搜索数据。（默认）
词	以16比特数据搜索数据。
双词	以32比特数据搜索数据。

(3) 搜索整个区域

应选择此选项搜索整个指定的范围。

(4) 方向

此区域用于指定搜索的方向。

向上	向前搜索。从当前光标位置向前（在屏幕中向上）搜索数据。
向下	向后搜索。从当前光标位置向后（在屏幕中向下）搜索数据。（默认）

(5) 地址

本区域用于指定要搜索的地址。

输入数值的默认基数是十六进制。也可以用一个符号或表达式指定地址。（参见“表5-6: 指定符号”。）

功能按钮

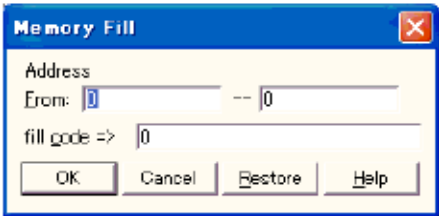
查找下一个	根据给定的条件搜索指定的数据。 如果找到了指定的字符串作为搜索的结果，该字符串会被突出显示。 要继续搜索，再次点击此按钮。
设置查找	设置指定的条件作为搜索条件并关闭此对话框。
停止（在搜索中）	停止搜索。
关闭	关闭此对话框。 （在搜索过程中，此按钮被<停止>按钮替换。）
帮助	显示在线帮助文件。

内存填充对话框

此对话框用于用指定的代码（填充代码）填充内存窗口中的内存内容。（参见“5.7 内存处理功能”。）

注意：如果内存组被使用，对0x10000或以上的地址空间无法指定地址范围（内存组区域除外）。（参见“图6-6：在内部ROM内存组没有使用时的地址空间图（内存组ROM大小为40KB）”。）

图6-35：内存填充对话框



- 打开
- 对每个区域的解释
- 功能按钮

打开

选择[编辑（Edit）]菜单 -> [内存] -> [填充...].

对每个区域的解释

（1）地址

此区域用于指定填充的范围和填充的代码。

从:	指定填充范围（起始地址 — 结束地址）。 输入数值的默认基数是十六进制。也可以用一个符号或表达式指定地址。（参见“表5-6：指定符号”。）
填充代码=>	指定在填充“从：”中指定的范围时使用的数据（填充代码）。 最多可以指定16个二进制数据串（字节数据串）。用“空白符”分隔每个数据。

功能按钮

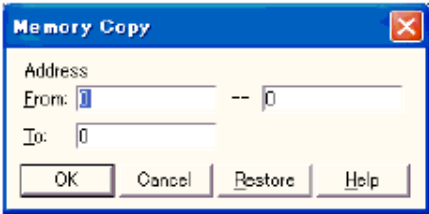
确定	根据给定的条件填充指定的数据。
停止（填充）	停止填充。
撤销	关闭此对话框。（在填充过程中，此按钮被<停止>按钮替换。）
恢复	恢复到打开此对话框前的状态。
帮助	显示在线帮助文件。

内存复制对话框

此对话框用于复制内存窗口中的内存内容。（参见“5.7 内存处理功能”。）

注意：如果内存组被使用，对0x10000或以上的地址空间无法指定地址范围和复制目标（内存组区域除外）。（参见“图6-6：在内部ROM内存组没有使用时的地址空间图（内存组ROM大小为40KB）”。）

图6-36：内存复制对话框



- 打开
- 对每个区域的解释
- 功能按钮

打开

选择[编辑（Edit）]菜单 -> [内存] -> [复制...]

对每个区域的解释

（1）地址

此区域用于指定复制源和复制目标地址。

输入数值的默认基数是十六进制。

也可以用 一个符号或表达式指定地址。（参见“表5-6：指定符号”。）

从:	指定复制来源的地址范围（起始地址 — 结束地址）。
到:	指定复制目标的起始地址。

功能按钮

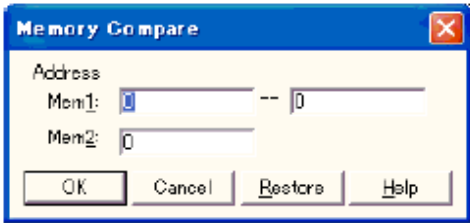
确定	根据给定的条件复制内存内容。
停止（复制）	停止复制。
撤销	关闭此对话框。（在复制过程中，此按钮被<停止>按钮替换。）
恢复	恢复到在打开此对话框前的状态。
帮助	显示在线帮助文件。

内存比较对话框

此对话框用于比较内存窗口中的内存内容。（参见“5.7 内存处理功能”。）

注意：如果内存组被使用，对0x10000或以上的地址空间无法指定地址范围和比较目标（内存组区域除外）。（参见“图6-6：在内部ROM内存组没有使用时的地址空间图（内存组ROM大小为40KB）”。）

图6-37：内存比较对话框



- 打开
- 对每个区域的解释
- 功能按钮

打开

选择[编辑（Edit）]菜单 -> [内存] -> [比较...]

对每个区域的解释

（1）地址

此区域用于指定比较源和比较目标地址。

输入数值的默认基数是十六进制。也可以用一个符号或表达式指定地址。（参见“表5-6：指定符号”。）

内存1:	指定比较源的地址范围（起始地址 — 结束地址）。
内存2:	指定比较目标的起始地址。

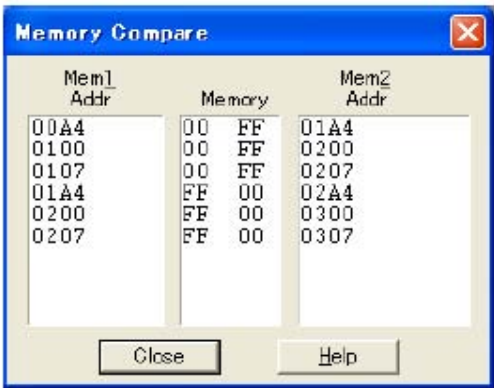
功能按钮

确定	根据给定的条件比较内存内容。如果在比较中没有发现差别，则显示“Wf200: 未发现差别”。如果发现了差别，则打开内存比较结果对话框。
停止（比较）	停止内存比较。
撤销	关闭此对话框。（在比较过程中，此按钮被<停止>按钮替换。）
恢复	恢复到打开此对话框前的状态。
帮助	显示在线帮助文件。

内存比较结果对话框

此对话框用于显示在内存比较比较对话框中进行内存比较时是否在内存内容中发现了差别。（参见“5.7 内存处理功能”。）

图6-38：内存比较结果对话框



- 对每个区域的解释
- 功能按钮

对每个区域的解释

（1）（比较结果显示区）

此区域用于显示内存的比较结果。只有在比较结果中发现的差别才会被显示。

内存1地址	显示其中发现差别的比较源地址。
内存	显示其中发现差别的数据。
内存2地址	显示其中发现差别的比较目标地址。

注意：在内存组被使用时地址宽度会发生变化。

功能按钮

关闭	关闭此对话框。
帮助	显示在线帮助文件。

DMM 对话框

此对话框用于设置DMM（动态内存修改）的地址和数据。（参见“5.16 DMM功能”。）在用户程序执行过程中通过DMM功能实时重写内存内容。

注意1: 只有在扩展选项对话框中的(1)使用MINICUBE扩展功能被选中时才能打开此对话框(当连接MINICUBE时)。

注意2: 如果内存组被使用，对于非内存组区域，无法指定0x10000或以上的地址空间的地址。（参见“图6-6: 在内部ROM内存组没有使用时的地址空间图（内存组ROM大小为40KB）”。）

注意3: 当连接MINICUBE时，在执行用户程序过程中无法对寄存器或SFR执行DMM（写）操作。

图6-39: DMM对话框



— 打开

— 对每个区域的解释

— 功能按钮

打开

选择[编辑 (Edit)]菜单 -> [DMM...], 或者在内存窗口/寄存器窗口/SFR窗口中点击<DMM...>按钮。

对每个区域的解释

(1) DMM目标选择区

此区域用于选择DMM的目标。可通过选择此选项按钮修改 (2) DMM设置区中显示的项目。

Memory	内存
Register	寄存器（当连接IECUBE时）
SFR	SFR（当连接IECUBE时）

备注：如果通过内存窗口、寄存器窗口或SFR窗口打开DMM对话框，则对应的选项按钮已经被选定。

(2) DMM设置区

本区域的显示根据 (1) DMM目标选择区中的选择而不同如下：

(a) 当选择内存时：

内存地址：	此区域用于指定数据要写入到的内存地址。 输入数值的默认基数是十六进制。也可以用一个符号或表达式指定地址。（参见“表5-6：指定符号”。）	
写入数据：	本区域用于指定写入到“内存地址：”中指定的内存地址的数据。	
数据大小：	本区域用于指定要写入的“写入数据：”中指定的数据的大小。	
	字节	写入为8比特数据。
	词	写入为16比特数据。
	双词	写入为32比特数据。

(b) 当选择寄存器时（连接IECUBE）：

寄存器名：	此区域用于指定数据要写入到的寄存器名。区分大小写，在命名中可以使用功能或绝对名。	
写入数据：	本区域用于指定写入到“寄存器名：”中指定的寄存器的数据。	
寄存器组：	指定寄存器组（0<=设置范围<=3）。 如果此区域留空，则指定当前的寄存器组。控制寄存器的指定被忽略。	

(c) 当选择SFR时（连接IECUBE）：

SFR名：	此区域用于指定数据要写入到的SFR名。区分大小写，不能指定只读的SFR。
写入数据：	本区域用于指定写入到“SFR名：”中指定的寄存器的数据。

（3）在写入时中断

在通过软件仿真（伪DMM功能）进行DMM时勾选此选项。（此选项在默认情况下不选。）

通过勾选此选项，在用户程序执行过程中通过DMM功能写入时的同时出现一个中断。

当连接IECUBE时，如果在（1）DMM目标选择区中选择了“寄存器”或“SFR”，则确定使用伪DMM。当连接MINICUBE时，则确定使用伪DMM。

功能按钮

设置	根据给定的条件写入数据。
关闭	关闭此对话框。
帮助	显示在线帮助文件。

寄存器窗口

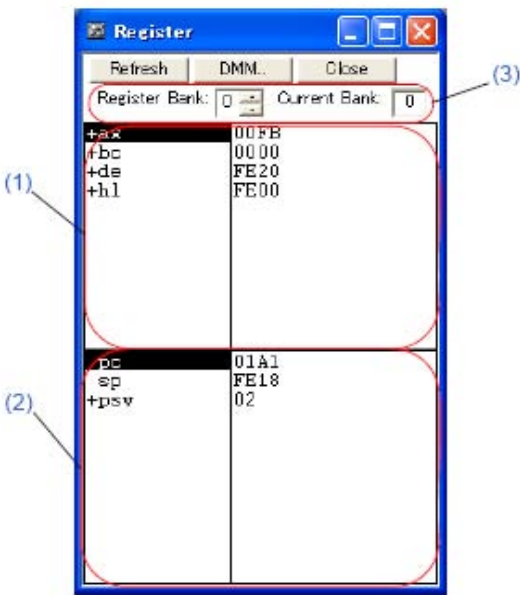
此对话框显示和修改寄存器（通用寄存器/控制寄存器）。（参见“5.8 寄存器处理功能”。）

可以在本窗口中进行一系列使用右键菜单和功能按钮等的其他操作。

本窗口中的每个区域是跳转功能的跳转指针。

注意：在用户程序执行过程中，使用扩展选项对话框中的（2）RAM监视器中设置的方法显示寄存器内容（当连接IECUBE时）。（参见“5.15 RRM功能”。）

图6-40：寄存器窗口



—打开

—对每个区域的解释

—[查看]菜单（寄存器窗口专有的项目）

—右键菜单

—功能按钮

打开

 点击Reg按钮或者选择[浏览（Browse）]菜单 -> [寄存器]。

对每个区域的解释

(1) 通用寄存器显示区

本区域用于显示和修改通用寄存器。

寄存器数值是通过直接输入修改的。要修改的区域以红色显示，当按下Enter键时修改的内容被写入目标内存。在用户程序执行的过程中，修改无法写入（如果试图写入，则会发生错误）。

可以按ESC键取消先前的数值。

(2) 控制寄存器显示区

本区域用于显示和修改控制寄存器。

双击“+”显示标签名和标签值（第一个符号从“+”变成“-”）；可双击“-”退出展开显示（第一个符号从“-”变成“+”）。

寄存器数值是通过直接输入修改的。要修改的区域以红色显示，当按下Enter键时修改的内容被写入目标内存。在用户程序执行的过程中，修改无法写入（如果试图写入，则会发生错误）。可以按ESC键取消先前的数值。

备注：当内存组被使用时，程序指针寄存器的地址宽度增加到8比特。

(3) 寄存器组设置区

此区域用于显示和指定通用寄存器组的数量。

寄存器组：	指定显示的寄存器组的数量。
当前组：	显示当前设置的寄存器组的数量。

[查看]菜单（寄存器窗口专有的项目）

当本窗口是当前窗口时，在[查看]菜单中添加了以下项目。

二进制（Bin）	显示二进制数字。
八进制（Oct）	显示八进制数字。
十进制（Dec）	显示十进制数字。
十六进制（Hex）	显示十六进制数字。（默认）
绝对名	显示寄存器名为绝对名。
功能名	显示寄存器名为功能名。（默认）

右键菜单

添加监视...	记录一个选择的字符串到监视窗口。 打开 添加监视对话框 。
二进制（Bin）	显示二进制数字。
八进制（Oct）	显示八进制数字。
十进制（Dec）	显示十进制数字。
十六进制（Hex）	显示十六进制数字。（默认）

功能按钮

刷新	用最新的数据更新窗口的内容。
DMM...	关闭 DMM对话框 （当连接IECUBE时）。
关闭	关闭此对话框。

SFR 窗口

此对话框显示和修改已经在添加I/O端口对话框中记录的SFR和I/O端口的内容。（参见“5.8 寄存器处理功能”。）

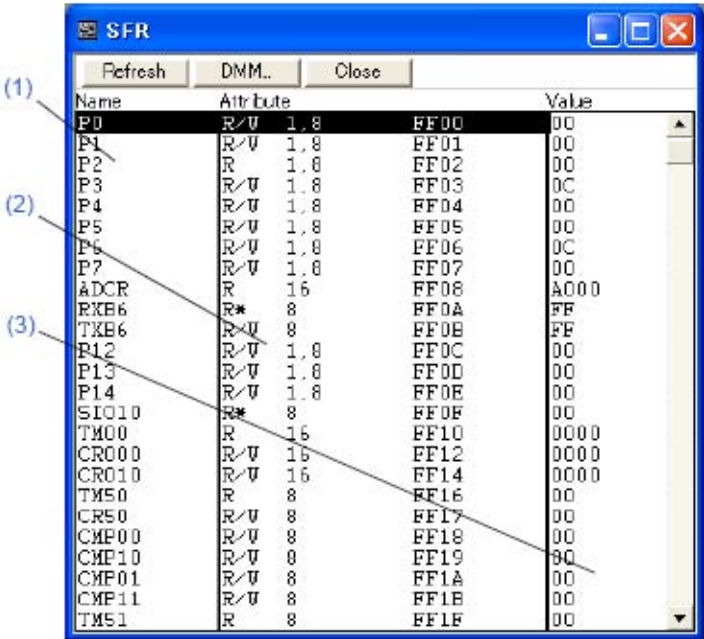
可以在本窗口中进行一系列使用右键菜单和功能按钮等的其他操作。

注意：但只读 SFR 和 I/O 端口的值是无法修改的。此外，在读取时使设备运行的 SFR 和 I/O 端口是读保护的，因此无法对其读取。要读取这些寄存器，选择一个寄存器，然后从右键菜单中选择和执行[强制读取]。

备注 1：在程序执行过程中，本窗口根据扩展选项对话框中（2）RAM 监视程序的设置显示（当连接 IECUBE 时）（参见“5.15 RRM 功能”。）

备注 2：在文件打开时显示起始位置如下：
第一次：从 SFR 最小地址开始显示。
第二次和以后：从活动状态的窗口关闭的地址开始显示。

图 6-41：SFR 窗口



- 打开
- 对每个区域的解释
- [查看]菜单（SFR窗口专有的项目）
- 右键菜单
- 功能按钮

打开



点击**SFR**按钮或者选择[浏览（Browse）]菜单 -> [SFR]。

对每个区域的解释

（1）名称

此区域用于显示 SFR 和 I/O 端口的名称。

如果没有定义 SFR 和 I/O 端口的值，I/O 端口名以淡色显示。

（2）属性

此区域用于显示 SFR 和 I/O 端口的属性。

本区域显示读/写属性、存取类型，并从左到右显示绝对地址。当显示比特 SFR 时，比特偏移值也会显示。

可以选择[查看]菜单 -> [属性]指定本区域是否显示。

读/写属性	
R	只读
W	只写
R/W	读/写
*	通过仿真寄存器读取的寄存器，在寄存器被读取时使设备停止运行。 要直接从 SFR 读取此属性，执行[查看]菜单 -> [强制读取]。 但有的设备不支持此功能。
存取类型	
1	可以比特为单位对其进行存取。
8	可以字节为单位对其进行存取。
16	可以词为单位对其进行存取。
32	可以双词为单位对其进行存取。

(3) 值

此区域用于显示和修改 SFR 和 I/O 端口的内容。

显示的内容根据属性的不同而变化：

黑色显示	只读或读/写
--	只写
**	如果被读取，值改变。

数值是通过直接输入修改的。要修改的区域以红色显示，当按下 Enter 键时修改的内容被写入目标内存。在用户程序执行的过程中，修改无法写入（如果试图写入，则会发生错误）。可以按 ESC 键取消先前的数值。

要注意的是只读 SFR 和 I/O 端口的值是无法修改的。

可以选择右键菜单 → [强制读取]读取读保护SFR和I/O端口的值。

[查看]菜单（寄存器窗口专有的项目）

当本窗口是当前窗口时，在[查看]菜单中添加了以下项目。

二进制（Bin）	显示二进制数字。
八进制（Oct）	显示八进制数字。
十进制（Dec）	显示十进制数字。
十六进制（Hex）	显示十六进制数字。（默认）
按名称排序	以字母顺序显示。
按地址排序	以地址顺序显示（默认）。
不排序（Unsort）	不进行排序。
属性	开启/关闭（2）属性的显示
选出（Pick Up）	仅显示SFR选择对话框中选择的寄存器。
选择...	打开SFR选择对话框。
强制读取	强制读取由于值会被修改而禁止读取的SFR，或者强制读取I/O端口和添加I/O端口对话框中添加的I/O保护区域的数据。

右键菜单

移动...	打开 地址移动对话框 。
添加监视...	打开 添加监视对话框 。
添加I/O端口...	打开 添加I/O端口对话框 。
二进制（Bin）	显示二进制数字。
八进制（Oct）	显示八进制数字。
十进制（Dec）	显示十进制数字。
十六进制（Hex）	显示十六进制数字。（默认）
按名称排序	以字母顺序显示。
按地址排序	以地址顺序显示（默认）。
不排序（Unsort）	不进行排序。
属性	开启/关闭 （2）属性 的显示
选出（Pick Up）	仅显示SFR选择对话框中选择的寄存器。
选择...	打开SFR选择对话框。
强制读取	强制读取由于值会被修改而禁止读取的SFR，或者强制读取I/O端口和 添加I/O端口对话框 中添加的I/O保护区域的数据。

功能按钮

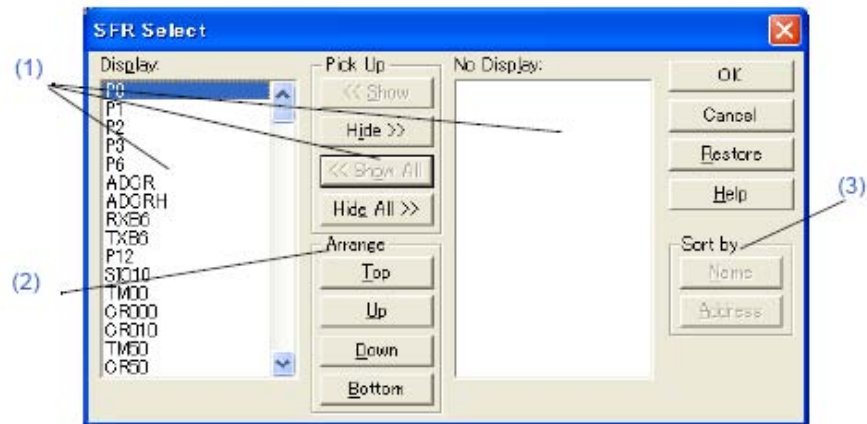
刷新	用最新的数据更新窗口的内容。
DMM...	关闭 DMM对话框 （当连接IECUBE时）。
关闭	关闭此对话框。

SFR 选择对话框

此对话框用于选择不在 [SFR 窗口](#) 中显示的 SFR 和 I/O 端口。（参见“5.8 寄存器处理功能”。）

此对话框还可用于指定寄存器和端口显示的顺序。

图 6-42: SFR 选择对话框



— 打开

— 对每个区域的解释

— 功能按钮

打开

当 [SFR 窗口](#) 是当前窗口时，选择[查看（View）]菜单 -> [选择...]

对每个区域的解释

（1）显示：、选出（Pick Up）、无显示：

此区域用于选择在 [SFR 窗口](#) 中显示和不显示的 SFR 和 I/O 端口。

（a）显示：

在 [SFR 窗口](#) 中显示 SFR 或 I/O 端口。

（b）不显示：

在 [SFR 窗口](#) 中不显示 SFR 或 I/O 端口。

(c) 选出 (Pick Up)

使用以下按钮改变在 **SFR 窗口** 中显示的 SFR 或 I/O 端口。

可以点击以上按钮同时按住 Ctrl 或 Shift 键移动两个或以上的寄存器。

<<显示	将选中的SFR或I/O端口从 (b)不显示: 移到 (a)显示: 。
>>隐藏	将选中的SFR或I/O端口从 (a)显示: 移到 (b)不显示: 。
<<显示全部	将所有的SFR或I/O端口移到 (a)显示: 。
>>隐藏全部	将所有的SFR或I/O端口移到 (b)不显示: 。

(2) 安排

以下按钮用于改变**(a)显示:** 中的显示顺序。

如果要改变显示安排, 不能同时选择多行。每次只能选择一行。

顶部	将选中的 SFR 或 I/O 端口移到列表的顶部。
上	将选中的 SFR 或 I/O 端口上移一行。
下	将选中的 SFR 或 I/O 端口下移一行。
底部	将选中的 SFR 或 I/O 端口移到列表的底部。

(3) 排序

以下按钮用于改变**(b)不显示:** 中的显示顺序。

按名称	以字母顺序显示。
按地址	以地址顺序显示。

功能按钮

确定	在 SFR窗口 中反映此对话框的选择并关闭此对话框。
取消	取消修改并关闭此对话框。
恢复	恢复到此对话框打开前的状态。
帮助	显示此对话框的在线帮助文件。

添加 I/O 端口对话框

此对话框用于记录添加到 [SFR 窗口](#) 的 I/O 端口。（参见“[5.8 寄存器处理功能](#)”。）

图 6-43: 添加 I/O 端口对话框



— 打开

— 对每个区域的解释

— 功能按钮

打开

选择[选项 (Option)]菜单 → [添加I/O端口...]

对每个区域的解释

(1) I/O 端口列表:

此区域列出当前已经记录的 I/O 端口。

如果记录了新的 I/O 端口，端口被加入到此列表。可以用[功能按钮](#)选择和修改或删除已经记录的 I/O 端口。

(2) 名称:

此区域用于指定要添加的 I/O 端口名（最大长度 15 个符号）。

(3) 地址:

此区域用于指定要添加的 I/O 端口的地址。

输入数值的默认基数是十六进制。也可以用一个符号指定地址。（参见“[表 5-6: 指定符号](#)”。）

可以在此区域内设置的地址可以是目标区域地址或 SFR 区域地址。

(4) 存取

此区域用于选择要添加的 I/O 端口的存取大小。

字节	8比特单位（默认）。
双词	16比特单位。 （只有在（2）名称：中指定SFR或外部SFR时才可选择。）

(5) 读/写

此区域用于指定要添加的 I/O 端口的存取属性。

在默认情况下，所有的属性被清除（即 I/O 端口既可读也可写）。

功能按钮

确定	在SFR窗口中反映添加的结果并关闭此对话框。
取消	取消修改并关闭此对话框。
恢复	恢复到原来的状态。
帮助	显示此对话框的在线帮助文件。
添加	添加指定地址的I/O端口。
修改	修改（1）I/O端口列表：中选择的I/O端口的设置。
删除	删除（1）I/O端口列表：中选择的I/O端口。

计时器对话框

IECUBE

此对话框用于记录和设置计时器事件的条件和显示执行时间测量的结果。（参见“5.12 事件功能”和“5.9 计时器功能（当连接 IECUBE 时）”。）

可点击<始终查看>按钮始终显示 (3) 执行事件显示区作为计时器结果对话框。

可以通过设置此对话框中的每个项目（最多为 256 个项目）然后点击<确定>按钮记录和设置计时器事件条件。计时器事件条件由事件管理器管理。

可同时使用（有效）的计时器事件条件数量是有限的。（参见“5.12.4 每个事件条件的启用事件数”。）

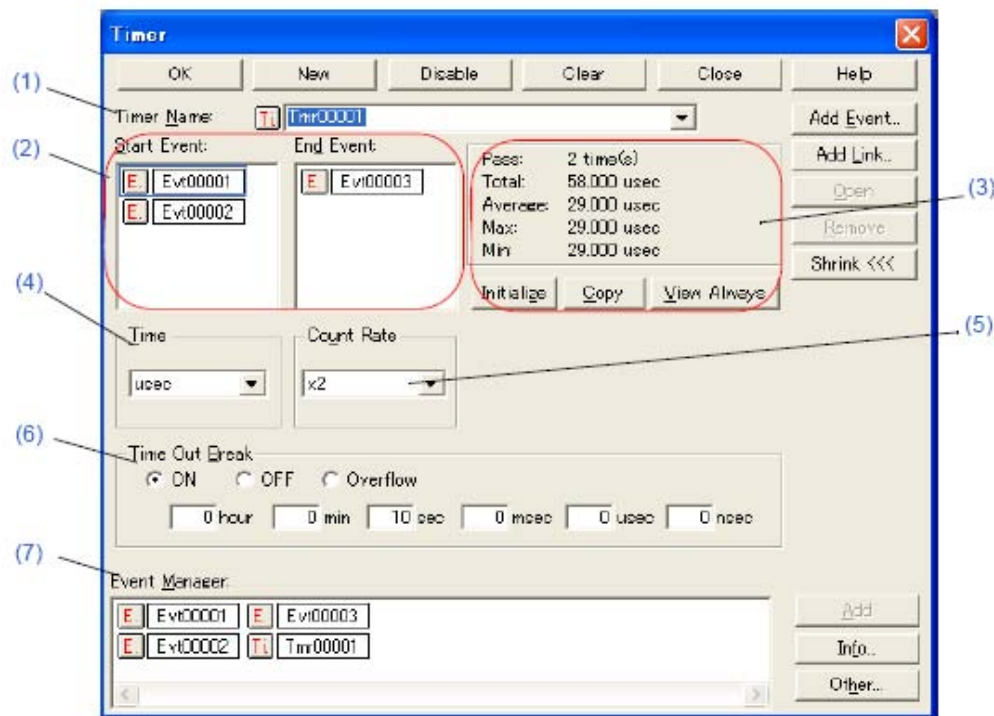
当选择了设置计时器事件条件时，显示执行事件测量的结果。

即使在用户程序执行的过程中仍然可以设置、删除、启用和禁用计时器事件条件。

备注 1：即使在用户程序执行的过程中，也会在每个 RRM 功能的采样时间更新测量结果显示的内容。

备注 2：即使在用户程序执行的过程中也可进行计时器事件条件的设置/启用/禁用/删除。

图 6-44：计时器对话框



— 打开

— 对每个区域的解释

— 功能按钮（相关事件功能）

打开

 点击Tim按钮或者选择[事件（Event）]菜单 -> [计时器...].

对每个区域的解释

（1）计时器名称：

此区域用于设置一个计时器事件的名称。

可以直接输入最多八个字符的数字字母串作为计时器的名称。

要显示已经创建的事件条件的内容，可以从下拉列表中选择。

要显示从用户程序执行到中断的事件，可以指定“运行—中断”（参见“5.9.2 运行—中断事件”）。

本区域左边的标记显示事件的使用状态（参见“5.19 事件图标”）。灰色标记表示事件条件正在被编辑且还没有被记录。

点击左边的标记可以启用或禁用一个事件条件。

（2）开始事件：、结束事件：

此区域用于为计时器设置一个事件条件。

关于本区域中可设置的项目数量，可参见“表 6-17：可设置的事件数量”。可以通过从事件管理器区中拖动要设置事件的图标并放到本区域中轻松地设置事件条件。关于详细信息，可参见“5.12.3 设置事件条件”。

（3）执行事件显示区

此区域显示程序执行时间的测量结果（参见“表 6-12：可测量的值”）。

通过	通过的次数。
合计	由开始事件和结束事件条件指定的测量区间中的合计执行时间。
平均值	平均执行时间。
最大值	最大执行时间。
最小值	最小执行时间。

注意： 由于计数器溢出而不可信任的测量结果以红色显示。

表 6-12：测量值

连接的IE	可测量的执行事件	可测量的执行次数
IECUBE	最大1分25秒（分辨率=20纳秒） 最大48小时50分钟（2K分，分辨率=41微秒）	最多4294967295次 （32 bit）

(a) 按钮

初始化	清除测量结果。
复制	以文本形式复制测量结果到剪贴板。
始终查看	打开 计时器结果对话框 。

(4) 时间

此区域用于选择显示计时器测量结果的单位。

当创建一个新事件时显示的是“Nsec”。

nsec	纳秒（默认）
usec	微秒
msec	毫秒
sec	秒
min	分钟

(5) 计数率

此区域用于设置执行时间测量的计时器计数率数值。

可以为每个计时器事件条件设置计时器计数率数值。

可以从下拉列表中选择×1 到×2048 的计数率数值。

(6) 超时中断

本区域用于设置 **(2) 开始事件：**、**结束事件：** 中指定的阶段测量时间（从计时器开始事件的建立到计时器结束事件的建立）的超时中断。

开	如果阶段测量事件超出指定的超时时间，则发生超时中断（执行终止）。 在文本框中指定超时时间，可指定达到最大可测量时间的值。
关	不发生超时中断（默认）。
溢出	如果阶段测量时间超过最大可测量时间，则发生超时中断（执行终止）（参见 “表6-12：可测量的值” ）。

注意：对运行一中断事件不能选择“开”；不能设置超时中断来指定超时时间（在选择了“溢出”时可以设定超时中断）。

(7) 事件管理器：

此区域用于显示记录的事件列表。（参见 [“表 5-19：事件图标”](#) [“（4）事件管理器区中的操作”](#)。）

功能按钮

参见[事件管理器](#)中的 [“功能按钮（相关事件功能）”](#)。

计时器结果对话框

IECUBE

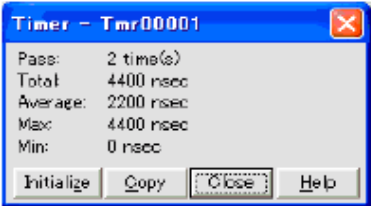
此对话框用于显示执行时间测量的结果。（参见“5.9 计时器功能（当连接 IECUBE 时）”。）

点击[计时器对话框](#)中的<始终查看>按钮，打开此对话框会一一对应地针对一个计时器事件条件。可同时打开两个或更多的计时器结果对话框。

最多可打开 256+1（运行一中断事件）个计时器结果对话框，同时可以测量的有效事件数量是“5.12.4 每个事件条件的启用事件数”+1（运行一中断事件）。

备注：即使在用户程序执行的过程中，也会在每个 [RRM 功能](#) 的采样时间更新测量结果显示的内容。

图 6-45：计时器结果对话框



- 打开
- 对每个区域的解释
- 功能按钮（相关事件功能）

打开

在[计时器对话框](#)中选择一个计时器事件条件，点击<始终查看>按钮。

对每个区域的解释

（1）执行时间显示区

与[计时器对话框](#)相同。参见“（3）执行时间显示区”。

功能按钮

初始化	清除测量结果。
复制	以文本形式复制测量结果到剪贴板。
关闭	关闭对话框。
帮助	显示此对话框的在线帮助文件。

跟踪视图窗口

IECUBE

此对话框用于显示跟踪结果。（参见“5.10 跟踪功能（当连接 IECUBE 时）”。）

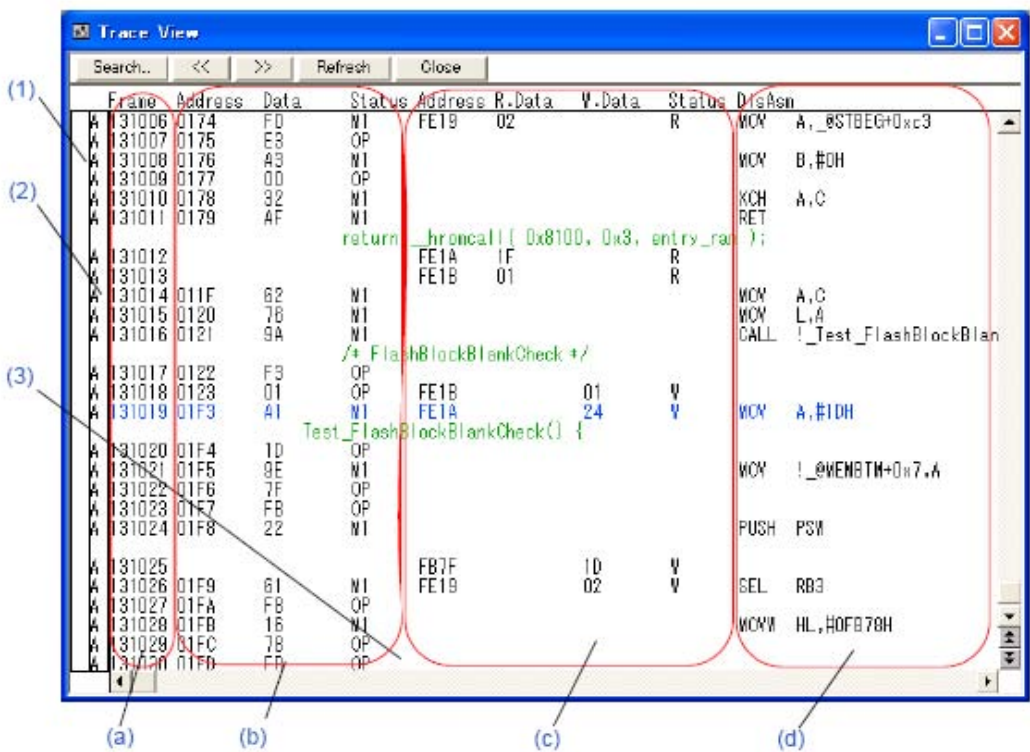
在中断或单步执行过程中进行显示更新。

此窗口具备混合显示模式（跟踪视图窗口）。

同时，窗口具备“5.18.3 具有联系窗口的跟踪结果（当连接 IECUBE 时）”。

在此窗口中可进行执行右键菜单和一系列使用功能按钮的其他操作。

图 6-45：跟踪视图窗口



- 打开
- 对每个区域的解释
- [查看]菜单（跟踪视图窗口专有的项目）
- 右键菜单
- 功能按钮

打开

点击TrW按钮或者选择[浏览（Browse）]菜单 -> [浏览]。

对每个区域的解释

(1) 点标记显示区

此区域显示事件设置状态（事件标记）。

如果在相应跟踪地址设置了执行事件或存取事件，则会显示与事件类型相对应的标记。

显示的标记不是在跟踪过程中的标记而是在跟踪结果显示时设置的事件标记。

(2) 跟踪模式显示区

此区域显示跟踪模式的类型。

A	无条件跟踪或阶段跟踪的帧
Q	认证跟踪的帧
S	单步执行帧
T	显示触发器帧

(3) 跟踪结果显示区

此区域显示跟踪的结果。

在 ID78K0-QB 中，显示一条水平线（块信息）表示程序执行的结束。

此外，还显示“表 6-13：当跟踪器停止时的中断原因”和“表 6-14：跟踪运行过程中复位的原因”中的停止跟踪器的原因和跟踪运行过程中复位的原因。如果有多个中断的原因，则显示所有的原因。

表 6-13：当跟踪器停止时的中断原因

原因	含义
事件中断	由事件造成的中断。
跟踪满中断	由于跟踪内存满而造成的中断。
跟踪延迟中断	由于跟踪延迟而造成的中断。
非映射中断	访问了非映射区域。
读保护	试图从读保护区域读取。
写保护	试图写入写保护区域。
SFR非法	SFR被非法访问。
SFR读保护	试图从禁止读取的SFR读取。
SFR写保护	试图写入禁止写入的SFR。
步进中断	单步执行中断。
人工中断	人工中断。
堆栈溢出	由堆栈溢出造成的中断。
堆栈下溢	由堆栈下溢造成的中断。
未初始化堆栈指针	由未进行堆栈指针初始化造成的中断。

原因	含义
软件中断	由软件造成的中断。
未初始化内存读取	读取了为初始化的内存。
超时中断	由于执行时间超时而造成的中断。
未指定非法	其他。
获取保护	获取保护中断。
跟踪停止	跟踪停止。
IMS (IXS) 非法	IMS (IXS) 非法中断。
闪存非法	闪存非法中断。
重试计数 (Retry Over)	重试计数中断。
周边中断	来自周边的中断。
执行前	执行前

表 6-14: 跟踪运行过程中复位的原因

原因	含义
内部复位	由周边芯片复位。
外部复位	由目标复位。
伪仿真复位	由于伪仿真而造成的复位。
POC复位	由于POC而造成的复位。

可在跟踪数据选择对话框中选择以下子区是否显示。

(a) 帧

此区域显示跟踪的帧编号。

(b) 地址数据状态 (助记显示)

此区域显示获取程序的结果。

(i) 状态

有以下几种状态。

程序获取显示

M1	获取指令的第一个字节。 如果获取地址是符号的开始，则第一行以蓝色突出显示。 (BRM1和M1帧也包括在内。)
OP	获取第二和以后字节的代码。
IF	无效获取或状态位置。

截取显示

SNAP	截取显示。
------	-------

(ii) 地址数据

此区域显示地址和数据。

地址	显示获取地址。
数据	显示获取数据。

(c) 地址 R 数据和 W 数据（数据存取显示）

此区域显示存取数据的结果。

VECT	矢量读取
RW	数据读/写
R	数据读
W	数据写

(d) DisAsm（助记显示）

此区域显示汇编的结果。

当(i)状态只是 M1，显示此区域。

[查看]菜单（跟踪视图窗口专有的项目）

当本窗口是当前窗口时，在[查看]菜单中添加了以下项目。

选择...	选择要显示的内容。 打开 跟踪数据选择对话框 。
选出	进行选出显示的设置。
关	不选出和显示（默认）。
搜索	选出和显示满足搜索条件的帧。
截取（Snap）	选出和显示截取帧。
混合	指定是否以混合显示模式显示源文件或者不显示。 选择：混合显示 清除：不显示（默认）
窗口同步	将跟踪视图窗口连接到以下窗口：“ 5.18.3 具有联系窗口的跟踪结果(当连接IECUBE时) ”） 选择的窗口被连接。
源文本	连接 源窗口 。
汇编	连接 汇编窗口 。
内存	连接 内存窗口 。

右键菜单

移动...	移动显示位置。 打开 跟踪移动对话框 。
跟踪清除	清除跟踪数据
选择...	选择要显示的内容。 打开 跟踪数据选择对话框 。
选出	进行选出显示的设置。
关	不选出和显示（默认）。
搜索	选出和显示满足搜索条件的帧。
截取（Snap）	选出和显示截取帧。
混合	指定是否以混合显示模式显示源文件或者不显示。 选择：混合显示 清除：不显示（默认）
窗口同步	将跟踪视图窗口连接到以下窗口：（参见“ 5.18.3 具有联系窗口的跟踪结果（当连接 IECUBE 时） ”）
源文本	连接 源窗口 。
汇编	连接 汇编窗口 。
内存	连接 内存窗口 。
源文本	显示从当前光标位置的数据值指定的跳转目标地址开始的相应源文本和源行。（参见“ 5.18.2 跳转功能 ”。）如果跳转目标地址中没有行信息，您无法跳转。打开 源窗口 。如果打开了活动的源窗口，则该窗口会显示在最前面（这样就可以对窗口进行操作了）。
汇编	从当前光标位置的数据值指定的跳转目标地址开始反汇编和显示。（参见“ 5.18.2 跳转功能 ”。）打开 汇编窗口 。如果打开了活动的汇编窗口，则该窗口会显示在最前面（这样就可以对窗口进行操作了）。
内存	显示从当前光标位置的数据值指定的跳转目标地址开始的内存内容。（参见“ 5.18.2 跳转功能 ”。）打开 内存窗口 。如果打开了活动的内存窗口，则该窗口会显示在最前面（这样就可以对窗口进行操作了）。

功能按钮

搜索...	打开 跟踪搜索对话框 并搜索或选出跟踪结果。 搜索的结果会在视图窗口中突出显示。 本操作与选择[查看（View）]菜单->[搜索...]相同。
<<	从当前光标位置开始向前搜索（在屏幕中向上）满足 跟踪搜索对话框 中设定条件的文本。 在搜索中本按钮会显示为<停止>。
>>	从当前光标位置开始向后搜索（在屏幕中向下）满足 跟踪搜索对话框 中设定条件的文本。 在搜索中本按钮会显示为<停止>。
刷新	用最新的数据更新本窗口的内容。
关闭	关闭此对话框

跟踪搜索对话框

IECUBE

此对话框用于在跟踪视图窗口中搜索。（参见“5.10 跟踪功能（当连接 IECUBE 时）”。）

通过设置每个项目然后点击<查找下一个>按钮即可开始搜索。

通过点击<设置查找>按钮即可使用跟踪视图窗口中的方向按钮（“<<”和“>>”）进行搜索。

备注 1: 通过选择[查看]—>[选出]—>[关]打开此对话框进行搜索跟踪数据；通过选择[查看]—>[选出]—>[搜索]打开此对话框选出和显示跟踪数据。

注意 1: 在使用菜单条或者在跟踪数据选择对话框中指定程序分支后，如果选出第一个M1获取帧（BRM1），则无法调用本对话框。

注意 2: 如果内存组被使用，对于非内存组区域，无法指定 0x10000 或以上的地址空间的地址。（参见“图 6-6: 在内部 ROM 内存组没有使用时的地址空间图（内存组 ROM 大小为 40KB）”。）

图 6-47: 跟踪搜索对话框



—打开

—对每个区域的解释

—功能按钮

打开

如果跟踪视图窗口是当前窗口，选择[查看]菜单—>[搜索...]或者点击同一窗口中的<搜索...>按钮。

对每个区域的解释

(1) 事件状态

此区域用于选择一个状态条件。如果省略了状态条件，则会搜索所有的帧（状态）。

所有状态	所有的帧（默认）
M1获取	M1获取（包括读、写）
R/W	数据读/写（包括BRM1）
R	数据读
W	数据写

(2) 存取大小

本区域用于选择存取大小条件。通过指定一个存取大小条件，可以确定由一个存取事件要检测的数据条件的宽度。

字节	搜索以8比特宽度的数据条件（仅在8比特存取的过程中）。
比特	搜索以1比特宽度的数据条件（仅在8比特存取的过程中） ^{注1、2} 。

注 1：如果指定一个访问事件为状态条件，则不显示比特的替换选项。如果指定比特或者 1，则发生错误。

注 2：在此情况下，则搜索以 1 比特宽度的数据条件。由于仿真器的运行，则不直接检测对比特的存取；通过内部设置以下地址条件和数据条件，仿真器搜索空白比特的存取：

输入实例 跟踪搜索的设置

地址：FE20.1 —> 地址：FE20

数据：1 数据：00000010B

掩码：11111101B

如果同一地址的另一个比特被存取或者同一地址的所有 8 比特被存取，若地址和比特匹配[地址. bit]的指定值，则根据指定的状态搜索跟踪数据。

备注：如果没有指定存取大小条件，则自动根据地址条件和数据条件判断，并如下设置：

- 如果地址条件设为以比特为单位：比特；
- 如果数据条件设为以 8 比特为单位：字节。

(3) 地址：、掩码：（地址设置区）

此区域用于指定地址条件。

输入数值的默认基数是十六进制。也可以用一个符号或表达式指定符号。（参见“表 5-6：指定符号”。）可如下设置：

表 6-15：地址条件可设置的范围（跟踪）

可设置的范围	条件
0 <= 地址值 <= 0xFFFF	当未使用组时。
0 <= 掩码值 <= 0xFFFF	
0 <= 地址值 <= (n<<16) 0xFFFF	当使用组时。 (n=使用组的最大值)
0 <= 掩码值 <= (n<<16) 0xFFFF	

(a) 地址：

设置一个地址条件（低位地址—高位地址）（可省略）。

可如下设置：

(i) 设置为一个点

仅为低位地址设置一个值或者为低位地址和高位地址设置同样的值。

(ii) 设置为一个范围

仅为低位地址设置一个值或者为低位地址和高位地址设置同样的值。

(iii) 设置为一个比特

仅为低位地址设置一个值或者为低位地址和高位地址设置同样的值。

以“地址.bit”的形式指定一个值。不能设置掩码。

显示比特位置的比特值必须满足：0 <= 比特 <= 7。

(b) 掩码：

为一个地址值设置掩码值（仅当(i)设置为一个点时）。

可以省略掩码。

掩码值是 1 的比特的地址值可以是 0 或 1。

例 1：

地址	0x4000到0x4000
掩码	0xFF

当如此设置时，地址 0x4000 到 0x40FF 满足条件。

例 2：

地址	0x4000到0x4000
掩码	0x101

当如此设时，地址 0x4000、0x4001、0x4100 和 0x4101 满足条件。

(4) 数据：、掩码：（数据设置区）

此区域用于设定数据条件。

输入数值的默认基数是十六进制。

可设置的范围如下，取决于 **(2) 存取大小：** 中指定的存取大小条件。（参见“[表 6-21：数据条件的可设置范围。](#)”）

(a) 数据：

设置一个数据值作为数据条件。也可以用符号指定数据。（参见“[表 5-6：指定符号。](#)”）

(b) 掩码：

为一个数据值设置掩码值。

在设置掩码后，掩码值是 1 的比特的掩码值可以是 0 或 1。

例 1：

数据	0x4000
掩码	0xFF

当如此设置时，地址 0x4000 到 0x40FF 满足条件。

例 2：

数据	0x4000
掩码	0x101

当如此设时，地址 0x4000、0x4001、0x4100 和 0x4101 满足条件。

(5) 扫描整个区域

选择此选项可扫描整个指定的区域。

(6) 帧：

此区域用于指定要搜索的帧编号。

输入数值的默认基数是十进制。也可以用帧编号指定格式指定符号。

(7) 方向

此区域用于指定搜索的方向。

向上	向前搜索。 从当前光标位置向前（在屏幕中向上）搜索数据。
向下	向后搜索。（默认） 从当前光标位置向后（在屏幕中向下）搜索数据。

功能按钮

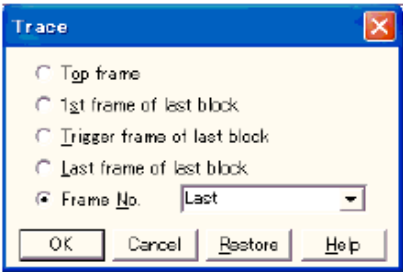
查找下一个	根据给定的条件搜索指定的数据。 如果找到了指定的帧作为搜索的结果，该帧会被突出显示。要继续搜索，再次点击此按钮。
设置查找	设置指定的条件作为搜索条件并关闭此对话框。
选出 (在搜索中为停止)	根据指定的数据搜索条件选出。 如果找到了指定的帧作为搜索的结果，该帧会选出。要选出一个满足不同条件的帧，再次点击此按钮。
关闭	关闭此对话框。
帮助	显示在线帮助文件。

跟踪移动对话框

IECUBE

此对话框用于指定跟踪视图窗口开始显示的位置。（参见“5.10 跟踪功能（当连接 IECUBE 时）”。）

图 6-48：跟踪移动对话框



- 打开
- 对每个区域的解释
- 功能按钮

打开

如果跟踪视图窗口是当前窗口，选择[查看]菜单->[移动...]

对每个区域的解释

(1) 帧选择区

此区域用于指定目标所在的帧。

顶部帧	移动显示起始位置到跟踪数据的第1帧。
上一块的第1帧	移动显示起始位置到跟踪数据的最新块帧中的第1帧。
上一块的触发帧	移动显示起始位置到跟踪数据的最新块帧中的触发帧。
上一块的最后1帧	移动显示起始位置到跟踪数据的最后1帧。
帧编号	移动显示起始位置到指定的帧编号。（参见“表6-16：帧编号指定格式”） 在默认条件下，选择窗口中选择的调用此对话框的对话框或“最后”。 输入数值的默认基数是十进制。 如果指定帧编号为0，显示起始位置被移到跟踪数据的第1帧。最多可记录16条输入历史。

备注：“块帧”是一个从跟踪器运行开始到结束跟踪的帧的单元。在跟踪视图窗口中一个被水平线分割的区域（块信息）是块帧。

表 6-16：帧编号指定格式

指定	缩写	内容
+数值	无	将显示起始位置从光标所在位置的帧向后（在屏幕中向下）移动指定数目（数值）的帧。
-数值	无	将显示起始位置从光标所在位置的帧向前（在屏幕中向上）移动指定数目（数值）的帧。
顶部	O	与“顶部帧”相同。
第一	S	与“上一块的第1帧”相同。
触发	T	与“上一块的触发帧”相同。
最后	L	与“上一块的最后1帧”相同。

功能按钮

确定	从指定位置开始跟踪显示。
取消	关闭此对话框。
恢复	恢复输入数据到原来的状态。
帮助	显示此对话框的在线帮助文件。

跟踪数据选择对话框

IECUBE

此对话框用于选择在跟踪视图窗口中显示的项目。（参见“5.10 跟踪功能（当连接 IECUBE 时）”。）

图 6-49：跟踪数据选择对话框



- 打开
- 对每个区域的解释
- 功能按钮

打开

如果跟踪视图窗口是当前窗口，选择[查看]菜单->[选择...]

对每个区域的解释

(1) 项目

本区域用于选择在跟踪视图窗口中显示的项目。可以选择显示或不显示以下项目，选择的域将显示。

帧	(a)帧栏。
指令获取地址	(b)地址数据状态（助记显示）中的地址栏。
指令获取数据	(b)地址数据状态（助记显示）中的数据栏。
指令获取状态	(b)地址数据状态（助记显示）中的状态栏。
内存获取地址	(c)地址读数据和写数据（数据存取显示）中的地址栏。
内存获取数据	(c)地址读数据和写数据（数据存取显示）中的数据栏。
内存获取状态	(c)地址读数据和写数据（数据存取显示）中的状态栏。
反汇编	(d)反汇编（助记显示）栏。

(2) 基数

本区域用于选择数据显示的基数。可以选择显示或不显示以下项目。

指令获取数据	(b)地址数据状态（助记显示）中的数据栏。
内存获取数据	(c)地址读数据和写数据（数据存取显示）中的数据栏。

十六进制（Hex）	以十六进制数字显示（默认）。
十进制（Dec）	以十进制数字显示。
八进制（Oct）	以八进制数字显示。
二进制（Bin）	以二进制数字显示。

(3) 选出

此区域用于选择选出条件。

选出关闭	不选出显示（默认）。
选出搜索帧	选出和显示满足搜索条件的帧。
选出截取帧	选出和显示截取帧。

功能按钮

确定	在跟踪视图窗口中反映此对话框中的选择结果。
取消	关闭此对话框。
恢复	恢复原来的状态。
帮助	显示此对话框的在线帮助文件。

跟踪对话框

IECUBE

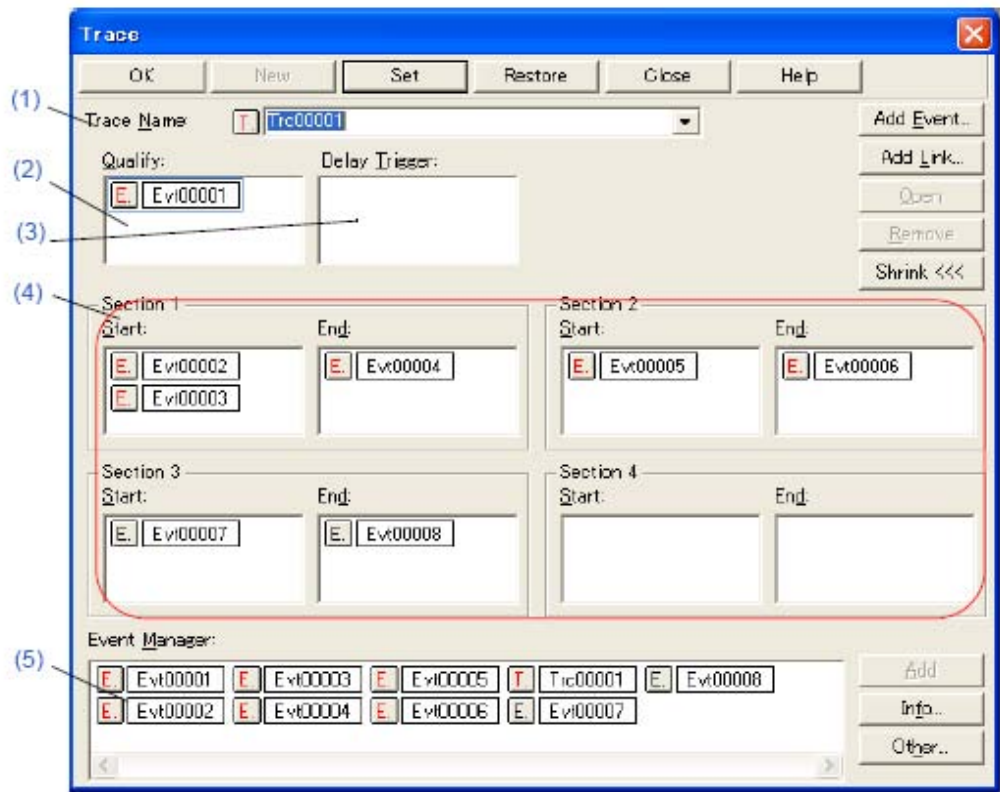
此对话框用于记录、设置和显示跟踪事件条件。（参见“5.12 事件功能”和“5.10 跟踪功能（当连接 IECUBE 时）”。）

在此对话框中指定执行有条件的跟踪时的跟踪事件条件。（参见“表 5-14：有条件的跟踪的类型”。）

记录和设置跟踪事件条件时通过设置此对话框中的每个项目然后点击<确定>按钮完成的。记录的跟踪事件条件时通过事件管理器管理的。

可同时使用（有效）的计时器事件条件数量是有限的。（参见“5.12.4 每个事件条件的启用事件数”。）

图 6-50：跟踪对话框



- 打开
- 对每个区域的解释
- 功能按钮（相关事件功能）

打开

 点击Trc按钮或选择[事件]菜单—>[跟踪...]

对每个区域的解释

(1) 跟踪名称:

此区域用于显示跟踪事件名。

直接输入不超过 8 个字符的数字字母串作为名称。

要显示已经创建的一个事件条件的内容，可以从下拉列表中选择。

此区域左边的标记表示事件的使用状态（参见“表 5-19: 事件图标”）。灰色标记表示事件条件正在被编辑且尚未被记录。

点击左边的标记可以启用或禁用事件条件。

(2) 认证

此区域用于设置认证跟踪的事件条件（参见“5.10.5 设置条件跟踪”）。

如果设置了两个或以上的事件，当每个事件发生时都进行跟踪。

在此区域中可如下设置事件条件的数量：

表 6-17: 可设置事件的数量

连接的IE	事件条件 合计（执行/存取）	事件联系条件
IECUBE	18 (8/10)	2

可以通过从事件管理器区中拖动要设置事件的图标并放到本区域中轻松地设置事件条件。关于详细信息，可参见“5.12.3 设置事件条件”。

(3) 延迟触发器:

此区域用于为延迟触发设置一个事件条件（参见“5.10.5 设置有条件的跟踪”）。

关于本区域中可设置的项目数量，可参见“表 6-17: 可设置的事件数量”。

可以通过从事件管理器区中拖动要设置事件的图标并放到本区域中轻松地设置事件条件。关于详细信息，可参见“5.12.3 设置事件条件”。

(4) 选择开始: 、选择结束:

可同时设置 4 个时段。

此区域用于为开始和停止一个时段跟踪的事件条件（参见“5.10.5 设置有条件的跟踪”）。

关于本区域中可设置的项目数量，可参见“表 6-17: 可设置的事件数量”。

可以通过从事件管理器区中拖动要设置事件的图标并放到本区域中轻松地设置事件条件。关于详细信息，可参见“5.12.3 设置事件条件”。

(5) 事件管理器:

此区域用于显示记录的事件列表（参见“表 5-19 时间图标”、“（4）事件管理区中的操作。”）。

功能按钮

参见事件管理器中的“功能按钮（相关事件功能）”。

延迟计数对话框

IECUBE

此对话框用于设置和显示延迟计数值。（参见“5.10 跟踪功能（当连接 IECUBE 时）”。）

通过设置一个延迟计数值，可以在满足跟踪对话框中设置的延迟触发事件的条件后将跟踪执行延时计数值指定的次数。（参见“5.10.5 设置有条件的跟踪”。）

图 6-51：延迟计数对话框



- 打开
- 对每个区域的解释
- 功能按钮（相关事件功能）

打开

选择[事件]菜单->[延迟计数...].

对每个区域的解释

（1）延迟计数

可选择以下项目。

第一	将触发指针放到第一个跟踪数据，跟踪所有的帧，然后停止跟踪器。
中间	将触发指针放到跟踪数据的中间，跟踪所有的帧，然后停止跟踪器。
最后	将触发指针放到跟踪数据的最后并立即停止跟踪器。

功能按钮

确定	确认设置并关闭此对话框。
恢复	恢复原来的设置。
取消	关闭此对话框。
帮助	显示此对话框的在线帮助文件。

代码覆盖窗口

IECUBE

此对话框显示代码覆盖测量的结果。（参见“5.11 代码测量功能（当连接 IECUBE 时）”。）

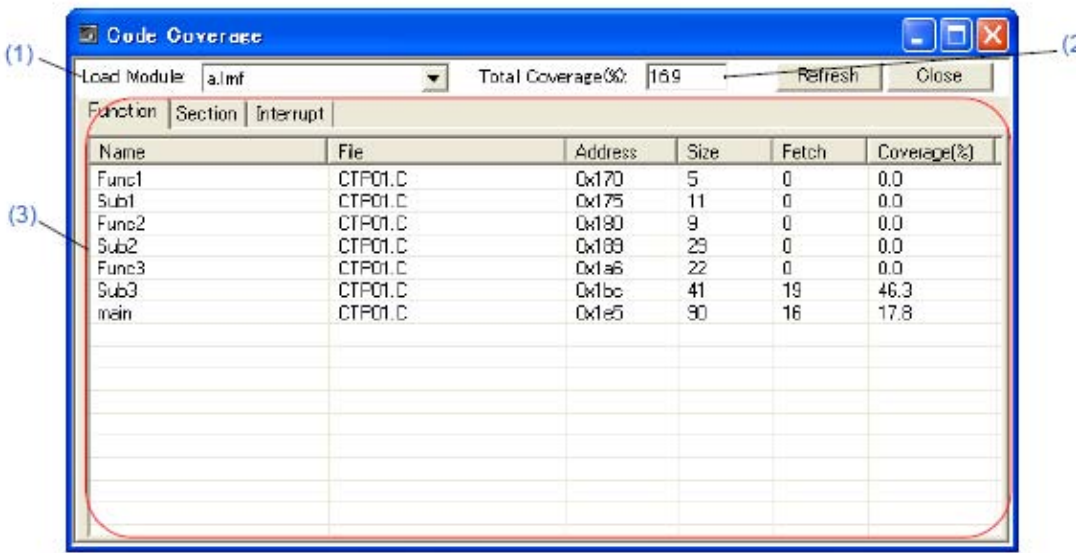
可以在源窗口或汇编窗口中查看用户程序已经执行的行和未执行的行。

代码覆盖测量的结果在中断时更新（在用户程序执行的过程中不会自动更新）。

注意：如果通过闪存自编程仿真替换片上内存数据，则代码覆盖测量的结果是不准确的。（参见“闪存选项对话框”。）

在此窗口中可进行执行右键菜单和一系列使用功能按钮的其他操作。

图 6-52：代码覆盖窗口



- 打开
- 对每个区域的解释
- 右键菜单
- 功能按钮

打开

 点击Cov按钮或选择[浏览]菜单->[代码覆盖]。

对每个区域的解释

(1) 负载模块

此区域用于选择已经下载的负载模块。

在没有下载的负载模块时，此区域是空白的。

(2) 总覆盖 (%)

此区域显示已经测量代码覆盖的区域的覆盖。

总覆盖 = 已执行（获取）的函数的总大小 / 函数的总大小（不包括覆盖测量范围外的部分）

注意：即使执行了所有的代码，覆盖也不会是 100%。比如，一个 switch 语句中 case 的跳转表是数据而不是要执行的代码，则对该跳转表要进行代码覆盖测量。

(3) 测量结果显示区

此区域显示，每个标签（函数、段、中断处理程序）的代码覆盖。

在没有下载的负载模块时，此区域是空白的。

备注：可以点击标题（在标签上）对显示的项目排序（在每次点击是切换升序/降序）。

(a) [函数] 标签

名称	函数名（在汇编程序源文件中以分段单位显示功能）。
文件	定义函数的文件名称。
地址	函数的起始地址。
大小	函数的大小（单位：字节）。
获取	执行（获取）的字节数。
总覆盖 (%)	函数的覆盖（0-100%） ---：当函数不在覆盖测量的范围内。

(b) [段] 标签

名称	段的名称。
类型	段的类型（代码和数据）。
地址	段的起始地址。
大小	段的大小（单位：字节）。
获取	获取（或执行）的字节数。
总覆盖 (%)	段的覆盖（0-100%） ---：当段不在覆盖测量的范围内。

(c) [中断]标签

名称	中断请求名称。
类型	中断类型（不可掩模、掩模、软件、安全ID、闪存掩模选项）。
状态	在程序中的使用状态 ---: 未知。
地址	中断处理程序的起始地址。
大小	中断处理程序的大小（单位：字节）。 非“使用”状态的最大大小。
获取	获取的字节数。
总覆盖（%）	中断处理程序的覆盖（0-100%） ---: 当中断处理程序不在覆盖测量的范围内。

通过使用选定行的起始地址值作为跳转指针，显示从此标签跳转到[源窗口](#)或[汇编窗口](#)。跳转目标地址将从跳转指针显示。

可以通过选择一个跳转源行，然后选择[跳转]菜单中的[源文本/汇编]然后执行跳转功能。也可以通过双击跳转源行进行跳转。

备注：在用户程序执行过程中使用<刷新>按钮更新数据。

右键菜单

源文本	用当前光标位置的数据值作为跳转目标地址，显示对应的源文本和源行。（参见“ 5.18.2 跳转功能 ”。） 如果跳转目标地址中没有行信息，您无法跳转。 打开 源窗口 。 如果打开了活动的 源窗口 ，则该窗口会显示在最前面（这样就可以对窗口进行操作了）。
汇编	从数据值指定的跳转目标地址开始反汇编和显示。（参见“ 5.18.2 跳转功能 ”。） 打开 汇编窗口 。 如果打开了活动的 汇编窗口 ，则该窗口会显示在最前面（这样就可以对窗口进行操作了）。
清除	清除覆盖测量结果。

功能按钮

刷新	用最新的监视数据更新本窗口的内容。
关闭	关闭此窗口。

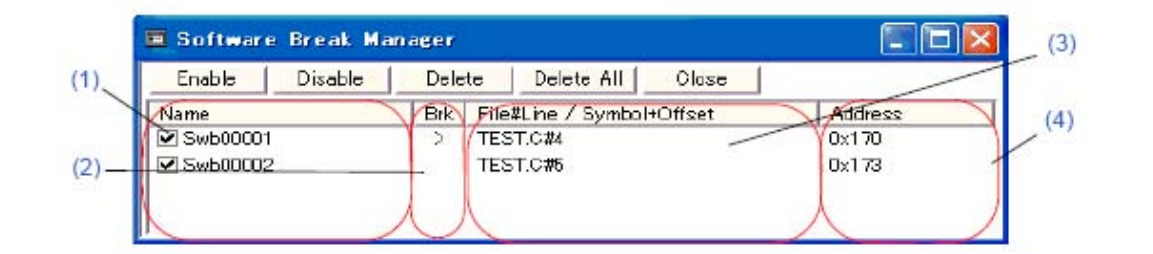
软件中断管理器

此窗口用于显示、启用、禁用和删除软件中断。（参见“5.4.4 硬件和软件中断”。）

可在此窗口设置软件断点；也可在源窗口或汇编窗口。（参见“5.4.2 断点设置”。）

备注：可以点击标题（在标签上）对显示的项目排序（在每次点击是切换升序/降序）。

图 6-53：软件断点管理器



- 打开
- 对每个区域的解释
- 功能按钮

打开

选择[事件]菜单—>[软件中断管理器]。

对每个区域的解释

(1) 名称

此区域显示记录的事件名称以及显示每个事件是否启用的复选框。

在默认条件下，事件名称以“Swb+[数字]”的形式显示。可以将其改为最多达 256 个字符的数字字母串。要更改一个事件名称，选择并点击事件名称，然后对其直接进行编辑。要设置编辑，按 Enter 键。

在启用事件时，则选择复选框；要禁用事件，则清空复选框。

此外，如果事件名称与源行对应，则双击事件名跳转到源窗口；如果事件名称不与源行对应，则双击事件名跳转到汇编窗口。

(2) Brk

显示的“>”标记表示在当前程序指针位置设置的软件中断事件（这样就可以轻松地识别造成中断的软件中断事件）。

(3) 文件#行 / 符号+偏移

此区域显示设置软件中断事件的位置如下：

- 程序\$文件名/#行号（如果事件名称与源行对应）
- 程序\$文件名/#符号+偏移（如果事件名称不与源行对应）

当符号重新下载时基于此评估事件。

(4) 地址

此区域显示设置软件中断事件的地址。

功能按钮

启用	启用选择的事件。
禁用	禁用选择的事件。
删除	删除选择的事件。
删除全部	删除所有设置的软件中断事件。
关闭	关闭此窗口。

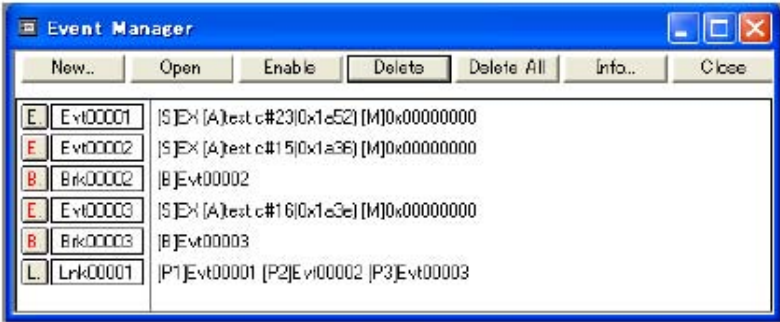
事件管理器

此窗口用于管理事件条件，显示、启用、禁用和删除各种事件条件。（参见“5.12 事件功能”。）

在此窗口中可进行执行右键菜单和一系列使用功能按钮（相关事件功能）的其他操作。

事件图标是跳转功能的跳转指针。

图 6-54：事件管理器（详细显示模式）



- 打开
- 对每个区域的解释
- [查看]菜单（事件管理器专有的项目）
- 右键菜单
- 功能按钮（相关事件功能）

打开

 点击**MGR**按钮或者选择[事件]菜单 -> [事件管理器]。

对每个区域的解释

（1）事件显示区

此区域显示记录的**各种事件条件**的图标（事件图标）。

选择右键菜单->[详细]，可显示详细信息。

(a) [在显示的列表中]



显示图标（参见“表 5-19：事件图标”）。

事件图标是跳转指针。（参见“5.18.2 事件功能”。）

(b) [详细显示模式]

[E]	Evi00001	[S]E×[A]test.c#23(0x1a52)[M]0x00000000
[E]	Evi00002	[S]E×[A]test.c#15(0x1a35)[M]0x00000000
[B]	Brk00002	[B]Evi00002
[E]	Evi00003	[S]E×[A]test.c#16(0x1a3e)[M]0x00000000
[B]	Brk00003	[B]Evi00003
[L]	Lnk00001	[P1]Evt00001 [P2]Evi00002 [P3]Evt00003

事件内容详细信息的显示可以使用以下关键信息作为分隔符。

表 6-18：显示事件详细信息分隔符

关键信息	内容
事件条件	
[S]	状态条件
[Z]	存取大小条件
[AR]	地址范围条件
[A]	地址条件 符号或表达式：（实际地址）
[D]	数据条件 符号或表达式：（实际地址）
[DR]	数据范围条件
[P]	通过计数条件（当连接IECUBE时）
[M]	掩模条件
事件联系条件（当连接IECUBE时）	
[P1] - [P4]	在第N行上的事件联系条件
[D]	禁用条件
[P]	通过计数条件
中断条件	
[B]	中断条件
跟踪条件（当连接IECUBE时）	
[M]	跟踪器控制模式
[T]	延迟触发条件
[D]	延迟计数
[S1] - [S4]	段跟踪起始条件
[E1] - [E4]	段跟踪结束条件
[Q]	认证跟踪条件
计时器条件（当连接IECUBE时）	
[S]	计时器测量起始条件
[E]	计时器测量结束条件

关键信息	内容
[R]	计时器标度比
[U]	计时器测量单位
[B]	超时中断条件
截取条件（当连接IECUBE时）	
[SN]	截取条件
[R]	寄存器条件
[B]	寄存器组条件
[M]	内存条件 符号或表达式：（实际地址）
[Z]	存取大小条件
[F]	SFR条件
存根条件（当连接IECUBE时）	
[SU]	存根条件
[A]	跳转条件 符号或表达式：（实际地址）
事件DMM条件（当连接IECUBE时）	
[DM]	事件DMM条件
[E]	SFR条件
[D]	经修改的数据
[M]	内存条件
[A]	数据修改地址
[Z]	经修改后数据的大小

[查看]菜单（事件管理器专有的项目）

当本窗口是当前窗口时，在[\[查看\]菜单](#)中添加了以下项目。

选择全部事件	选择所有记录的事件。
删除事件	选择一个选定的事件。
根据名称排序	按事件名称的顺序显示图标。
根据类型排序	按事件类型的顺序显示图标。
不排序	不排列图标（默认）。
详细	显示详细信息。
总览	列表显示（默认）。

右键菜单

根据名称排序	按事件名称的顺序显示图标。
根据类型排序	按事件类型的顺序显示图标。
不排序	不排列图标（默认）。
详细	显示详细信息。
总览	列表显示（默认）。
源文本	用选定事件的位置作为跳转目标地址，显示对应的源文本和源行。（参见“5.18.2 跳转功能”。） 如果跳转目标地址中没有行信息，您无法跳转。 打开源窗口。 如果打开了活动的源窗口，则该窗口会显示在最前面（这样就可以对窗口进行操作了）。
汇编	从作为跳转目标地址的选定事件位置开始显示汇编窗口。（参见“5.18.2 跳转功能”。） 打开汇编窗口。 如果打开了活动的汇编窗口，则该窗口会显示在最前面（这样就可以对窗口进行操作了）。
内存	从作为跳转目标地址的选定事件位置开始显示内存内容。（参见“5.18.2 跳转功能”。） 打开内存窗口。 如果打开了活动的内存窗口，则该窗口会显示在最前面（这样就可以对窗口进行操作了）。

功能按钮（相关事件功能）

描述左右的功能以及相关的事件对话（事件管理器、事件对话框和各种事件设置对话框）（参见“表 5-17：各种事件条件”）。

确定	（事件对话框、事件联系对话框） 自动记录正在被编辑的事件条件（如果有的话）并关闭此对话框。 在选择模式下： 选择一个事件条件，并再次显示调用事件联系对话框的设置对话框（在标题条中显示）。如果已经关闭了调用对话框，选择模式回到正常模式，事件对话框不关闭。否则，对话框不会关闭。
	（非以上对话框） 自动记录正在被编辑的事件条件（如果有的话）并关闭此对话框。每个事件条件一旦被记录立即生效。
新建...	（事件管理器） 打开此对话框并创建新的事件。 点击每个按钮，会打开相应的事件设置对话框并将新事件名设定好。在事件设置对话框打开后，此对话框关闭。 点击<撤销>按钮返回到事件管理器。
	（非以上对话框） 在此对话框中新创建一个事件条件。

设置	(事件对话框、事件联系对话框) 记录各种事件条件。由于对话框在新事件注册后不会关闭,因此可记录新的事件条件。 在选择模式下: 选择一个事件条件。如果有事件正在被编辑,会自动记录和选择该事件。
	(非以上对话框) 记录各种事件条件。由于对话框在新事件注册后不会关闭,因此可记录新的事件条件。 每个事件条件一旦被记录立即生效。
启用/禁用	启用或经用选定的事件条件。 但不能启用或禁用事件条件或事件联系条件。 与点击事件图标的标记操作相同。
清除	清除事件条件的内容。
恢复	恢复一个经编辑的事件条件的内容。如果显示了没有记录的事件条件,所有事件名栏外的栏都是空的或者设置了默认值。
撤销/关闭	关闭此对话框。如果事件条件正在被编辑,在关闭对话框时事件条件不会被记录。
帮助	显示本窗口的帮助窗口。
事件联系...	打开 事件联系对话框 (当连接IECUBE时)。
中断...	打开 中断对话框 。
跟踪...	打开 跟踪对话框 (当连接IECUBE时)。
截取...	打开 截取对话框 (当连接IECUBE时)。
存根...	打开 存根对话框 (当连接IECUBE时)。
计时器...	打开 计时器对话框 (当连接IECUBE时)。
事件DMM...	打开 事件DMM对话框 (当连接IECUBE时)。
管理器	打开 事件管理器 。
添加事件...	以选择模式打开 事件对话框 ,然后选择或新建一个要设置的事件条件。在按下<添加事件...>按钮后,事件条件会被添加到选定的区域。
添加联系...	以选择模式打开 事件联系对话框 ,然后选择或新建一个要设置的事件联系条件。在按下<添加联系...>按钮后,事件条件会被添加到选定的区域。
打开	根据选择的事件条件打开各个事件设置对话框。每个设置对话框显示选定事件条件的内容。 与双击事件图标或按Enter键的操作相同。
移除/删除	删除选择的事件。 当要删除一个事件条件或事件联系条件时,如果一个事件用作多个事件条件,则会出现错误且无法删除该事件条件或事件联系条件。
删除全部	删除软件中断事件外的所有事件。
展开>>>/缩回<<<	打开(展开>>>)或关闭(缩回<<<)事件管理器区的显示。 展开或缩小对话框的大小。

添加	添加事件管理器区的事件条件或事件联系条件到设置区并显示为焦点。
信息...	打开选择显示信息对话框。 此对话框用于改变显示模式和重新安排事件名称。 图6-55: 选择显示信息对话框  <按名称排序>.....按名称顺序排列事件。 <按类型排序>.....按类型顺序排列事件。 <不排序>.....按事件记录的顺序不排序显示事件。 <详细>.....设置详细显示模式。 <总览>.....设置列表显示模式。 <撤销>.....关闭此对话框（与ESC键相同）。
其他...	打开此对话框选择事件类型。 点击每个按钮，会打开相应的事件设置对话框并将新事件名设定好。在事件设置对话框打开后，此对话框关闭。 <管理器...>.....关闭事件管理器。 <撤销>.....关闭此对话框以创建事件条件。

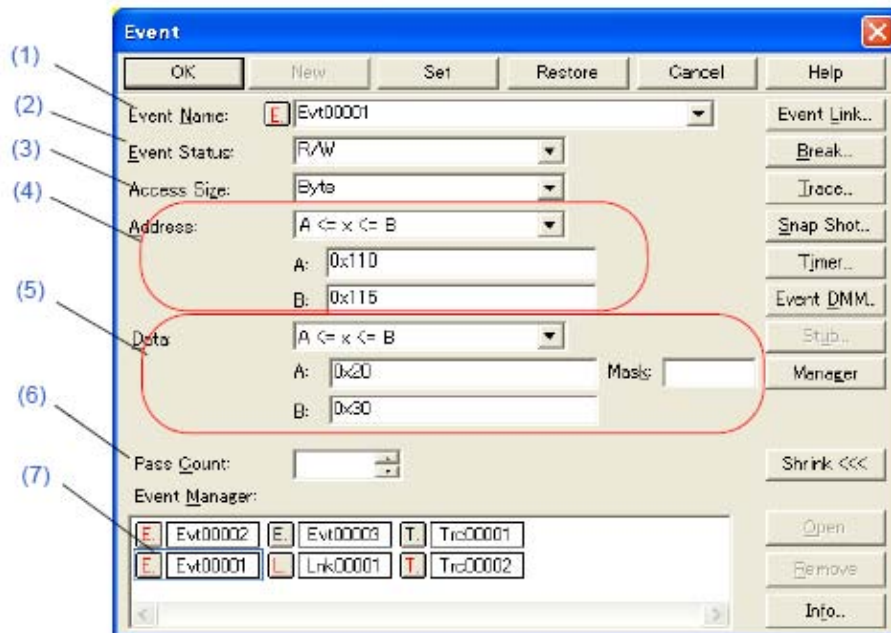
事件对话框

此对话框用于记录和显示事件条件。（参见“5.12 事件功能”。）

可以在本对话框中设置每个功能然后点击<确定>按钮设置事件条件。可以通过事件管理器管理已经记录的事件条件。

可以为各种事件条件设置一个事件条件。可同时使用的计时器事件条件数量是有限的。（参见“5.12.4 每个事件条件的启用事件数”。）

图 6-56：事件对话框



—打开

—对每个区域的解释

—功能按钮（相关事件功能）

打开

一般模式

如果如下打开事件对话框，可以记录事件条件而不指定目的。



点击Evm按钮或者选择[事件]菜单 -> [事件...]。

选择模式

如果在事件对话框已经如下打开时按下<确定>按钮，则可以在打开此对话框的设置对话框中记录一个事件条件。（该对话框显示在标题条上。）

在每个各种事件设置对话框中，按<添加事件...>按钮。

对每个区域的解释

(1) 事件名称

此区域用于设置事件名称。

可以直接输入不超过 8 个字符的数字字母串作为事件的名称。

要显示一个已经创建的事件条件的内容，可以从下拉列表中选择。

在选择模式，可以在调用事件对话框的设置对话框的事件条件设置区设置选中的事件条件。

此区域左边的标记显示事件的使用状态（参见“5-19：事件图标”）。

灰色的 E 标记表示事件条件正在被编辑且还未被记录。

(2) 事件状态：

此区域用于选择状态条件。

通过指定一个状态条件，可以确定一个执行事件和存取事件的类型。（如果指定了一个执行事件，则不能向 (3) 存取大小： 和 (5) 数据： 、掩码： 中输入内容。）

可以指定的状态条件列出如下。

图 6-19： 状态条件

状态	缩写	含义
执行事件		
执行	EX	程序执行（当连接IECUBE时）。
执行前	EX-B	程序执行（执行前中断） ^注 （当连接IECUBE时）。
存取事件		
读/写	RW	内存读/写
读	R	内存读取
写	W	内存写入

注： 仅对中断条件可用。

(3) 存取大小：

此区域用于选择存取大小的条件。

通过从下拉列表中选择存取大小的条件，可以确定一个存取事件要探测的数据条件的存取宽度。

字节	探测以8比特宽度的数据条件（仅在8比特存取的过程中）。
词	探测以16比特宽度的数据条件（仅在16比特存取的过程中）。（IECUBE）

比特	探测以1比特宽度的数据条件（仅在8比特存取的过程中）。 在此情况下，则检测以 1 比特宽度的数据条件。由于仿真器的运行，则不直接检测对比特的存取；通过内部设置以下地址条件和数据条件，仿真器搜索空白比特的存取： <table><tr><td>输入实例</td><td>跟踪搜索的设置</td></tr><tr><td>地址：FE20.1 —> 地址：FE20</td><td></td></tr><tr><td>数据：1</td><td>数据：00000010B</td></tr><tr><td></td><td>掩码：11111101B</td></tr></table> 如果同一地址的另一个比特被存取或者同一地址的所有 8 比特被存取，若地址和比特匹配[地址. bit]的指定值，则根据指定的状态搜索跟踪数据。 当写入数据到一个比特，所有的 8 比特都会被读写。若指定读或读/写为状态，如果[地址. bit]的指定值匹配，此时进行读取操作，则事件发生。	输入实例	跟踪搜索的设置	地址：FE20.1 —> 地址：FE20		数据：1	数据：00000010B		掩码：11111101B
输入实例	跟踪搜索的设置								
地址：FE20.1 —> 地址：FE20									
数据：1	数据：00000010B								
	掩码：11111101B								

备注：如果没有指定存取大小条件，则自动根据地址条件和数据条件判断，并如下设置：

- 如果地址条件设为以比特为单位：比特；
- 如果数据条件设为以 8 比特为单位：字节；
- 如果数据条件设为以 16 比特为单位：词。

(4) 地址：

此区域用于指定地址条件（地址值）。

注意：如果内存组被使用，对于非内存组区域，无法指定 0x10000 或以上的地址空间的地址。（参见“图 6-6：在内部 ROM 内存组没有使用时的地址空间图（内存组 ROM 大小为 40KB）”。）

表 6-20：地址条件（事件）的可设置范围

可设置的范围	条件
当连接IECUBE时（未使用组）：	
0 <= 地址值 <= 0xFFFF	—
0 <= 掩码值 <= 0xFFFF	—
当连接IECUBE时（使用组）：	
0 <= 地址值 <= (n<<16) 0xFFFF	n=使用组的最大值
0 <= 掩码值 <= (n<<16) 0xFFFF	n=使用组的最大值
当连接MINICUBE时：	
0xF800 <= 地址值 <= 0xFFFF	—

(a) 地址：

从下拉列表选择地址范围（指定方法）并在 **A：** 和 **B：** 区域指定地址值。

当连接 MINICUBE 时，仅能选择 $x == A$ 。

输入数值的默认基数是十六进制。也可以用一个符号或表达式指定地址。（参见[指定符号](#)。）

$x == A$	规定地址值为 A ：指定的地址值。
$x >= A$	规定地址值高于为 A ：指定的地址值。
$x <= B$	规定地址值低于为 B ：指定的地址值。
$A <= x <= B$	规定地址值为 A ：到 B ：范围之间的地址值。 在此情况下，当指定了执行前事件外的项目时，则使用两个事件。
$x < A \ \&\& \ B < x$	规定地址值为 A ：到 B ：范围之外的地址值。

（5）数据：、掩码：

此区域用于设定数据条件（数据值、掩码值）。

输入数值的默认基数是十六进制。

可设置的范围如下，取决于（3）[存取大小](#)：中指定的存取大小条件。

表 6-21：数据条件的可设置范围

可设置的范围	条件
字节	$0 \leq \text{数据值} \leq 0xFF$ $0 \leq \text{掩码值} \leq 0xFF$
词	$0 \leq \text{数据值} \leq 0xFFFF$ $0 \leq \text{掩码值} \leq 0xFFFF$
比特	数据值=0或1 掩码值=无法指定

（a）数据：

从下拉列表选择地址范围（指定方法）并在 **A**：和 **B**：区域指定数据值。

当连接 MINICUBE 时，仅能选择“无条件”或“ $x == A$ ”。

输入数值的默认基数是十六进制。也可以用一个符号或表达式指定地址。（参见[指定符号](#)。）

无条件	不在乎（指定所有的数据值）。
$x \neq A$	规定数据值为不匹配为 A ：指定的数据值。
$x == A$	规定数据值为匹配 A ：指定的数据值。
$x >= A$	规定数据值高于为 A ：指定的数据值。
$x <= B$	规定数据值低于为 B ：指定的数据值。
$A <= x <= B$	规定数据值为 A ：到 B ：范围之间的数据值。 在此情况下，则使用两个事件。
$x < A \ \&\& \ B < x$	规定数据值为 A ：到 B ：范围之外的数据值。

(b) 掩码:

为一个数据值设置掩码值。

在设置掩码后，掩码值是 1 的比特的掩码值可以是 0 或 1。

例 1:

数据	0x4000
掩码	0xFF

当如此设置时，地址 0x4000 到 0x40FF 满足条件。

例 2:

数据	0x4000
掩码	0x101

当如此设时，地址 0x4000、0x4001、0x4100 和 0x4101 满足条件。

(6) 通过计数

通过计数指定在一个指定的条件满足前（当连接 MINICUBE 时）的用户程序执行过程中事件条件必须满足的次数。如果没有指定通过计数，则设为 1（一旦事件条件满足则条件满足）。如果在 (2) 事件状态: 中指定了执行前，此区域不可用。

表 6-22: 通过计数

装置	范围
IECUBE	1-255

(7) 事件管理器:

此区域用于显示记录的事件列表。（参见“表 5-19: 事件图标”“(4) 事件管理器区中的操作”。）

功能按钮

参见事件管理器中的“功能按钮（相关事件功能）”。

事件联系对话框

IECUBE

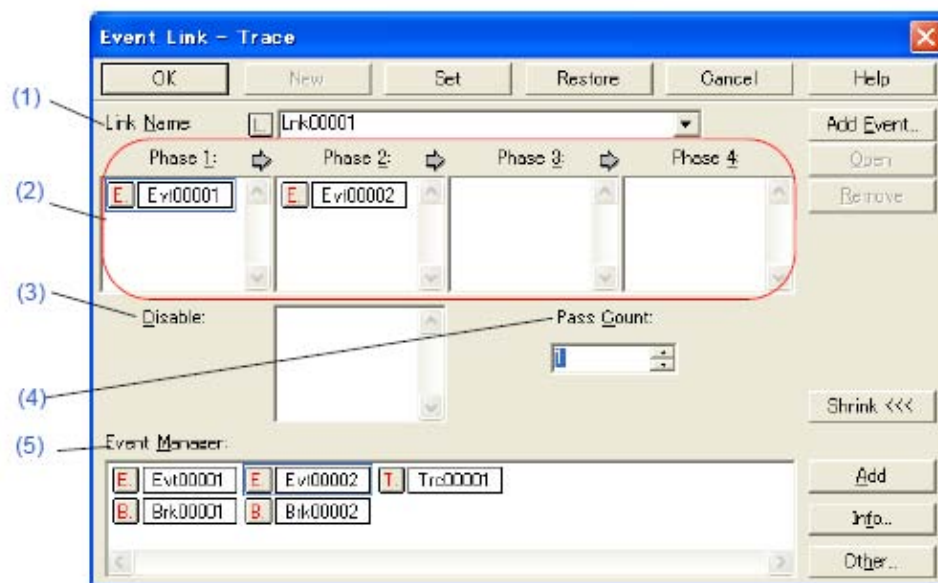
此对话框用于记录和显示事件联系条件。（参见“5.12 事件功能”。）

可以通过在此对话框中设置每个项目（最多 256 个），然后点击<确定>按钮来记录事件联系条件。记录的事件联系条件由事件管理器管理的。

可同时使用的计时器事件联系条件数量是有限的。（参见“5.12.4 每个事件条件的启用事件数”。）

注意：可在事件联系对话框中指定的事件条件是：非执行前事件，通过计数值为 0 或 1。（不能使用通过计数值为 2 或更大的事件条件。）

图 6-57：事件联系对话框



—打开

—对每个区域的解释

—功能按钮（相关事件功能）

打开

一般模式

如果如下打开事件联系对话框，可以记录事件联系条件而不指定目的。

选择[事件]菜单 → [事件联系...]

选择模式

如果在事件联系对话框已经如下打开时按下<确定>按钮，则可以在打开此对话框的设置对话框中记录一个事件联系条件。

在每个各种事件设置对话框中，按<添加联系...>按钮。（打开事件联系对话框的设置对话框显示在标题条上。）

对每个区域的解释

（1）联系名称

此区域用于设置事件联系名称。

可以直接输入不超过 8 个字符的数字字母串作为名称。

要显示一个已经创建的事件联系条件的内容，可以从下拉列表中选择。

在选择模式，可以在调用事件联系对话框的设置对话框的事件联系条件设置区设置选中的事件条件。

此区域左边的标记显示事件联系条件的使用状态（“5-19：事件图标”）。灰色的 L 标记表示事件联系条件正在被编辑且还未被记录。

（2）阶段 1： 、阶段 2： 、阶段 3： 、阶段 4：

此区域用于指定事件条件和事件检测的顺序。

最多可指定四个顺序。如果在程序执行的过程中检测到一个禁用条件，则初始化到目前已经满足的事件条件，对事件条件从第一个开始进行检测。如果在同时检测到一个联系条件和禁用条件，则禁用条件优先。

设置顺序为阶段 1→阶段 2→阶段 3→阶段 4。按此顺序，不需要对阶段 4 进行设置。在此情况下，当为最后一个阶段设置的事件条件被检测到时，事件发生。只能为阶段 1 设置事件条件，或者可以为两个或以上的阶段设置同样的事件条件。

可以为本区域的每个阶段设置的事件条件的数量如下：

表 6-23： 事件联系对话框中的事件条件数量

连接的IE	每个阶段	禁用区域 （执行/存取）	合计 （执行/存取）
IECUBE	18	18	18 (8/10)

(3) 禁用：

此区域用于设置禁用到目前已经满足的事件条件。（表 6-23：事件联系对话框中的事件条件数量。）

可以通过从事件管理器区中拖动要设置事件的图标并放到本区域中轻松地设置事件条件。关于详细信息，可参见“5.12.3 设置事件条件”。

(4) 通过计数

此区域用于设置通过计数条件（可设置范围：0—255）。

通过计数指定在一个指定的条件满足前的用户程序执行过程中事件条件必须满足的次数。

如果没有指定通过计数，则设为 1（一旦事件条件满足则条件满足）。

(5) 事件管理器：

此区域用于显示记录的事件列表。（参见“表 5-19：事件图标”“（4）事件管理器区中的操作”。）

功能按钮

参见事件管理器中的“功能按钮（相关事件功能）”。

中断对话框

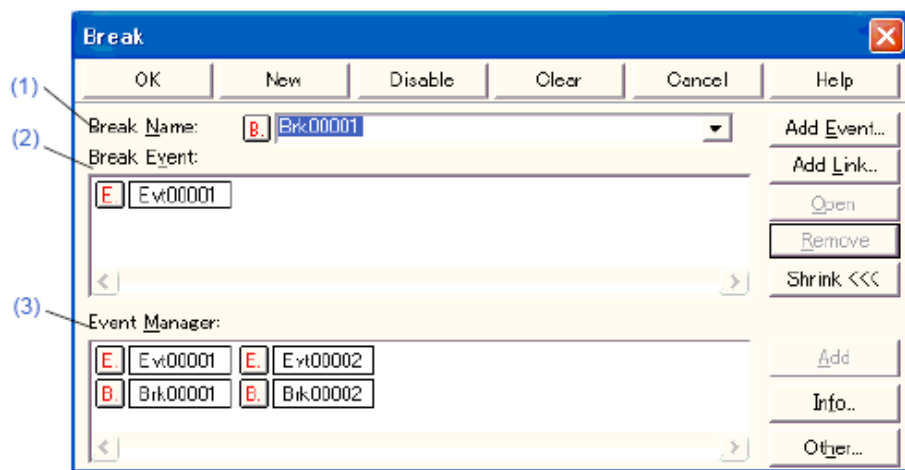
此对话框用于记录、设置和显示中断事件条件。（参见“5.12 事件功能”和“5.4 中断功能”。）

可以通过在此对话框中设置每个项目（最多 256 个），然后点击<确定>按钮来记录事件联系条件。记录的事件联系条件由事件管理器管理的。

可同时设置（启用）的中断事件条件数量是有限的。（参见“5.12.4 每个事件条件的启用事件数”。）

注意：中断事件条件的设置/启用/禁用/删除操作可以在用户程序执行的过程中进行。

图 6-58：中断对话框



— 打开

— 对每个区域的解释

— 功能按钮（相关事件功能）

打开



点击Brk按钮或者选择[事件]菜单 → [中断...]

对每个区域的解释

(1) 中断名称：

此区域用于设置中断事件名。直接输入不超过 8 个字符的数字字母串作为名称。

要显示已经创建的事件条件的内容，可从下拉列表中选择。

此区域左边的标记表示事件的使用状态（参见“表 5-19：事件图标”）。灰色标记表示事件条件正在被编辑且尚未被记录。可以点击左边的标记启用或禁用事件条件。

(2) 中断事件：

此区域用于设置中断的事件条件。

可在此区域设置数个事件条件和事件联系条件。

表 6-24：可以在条件设置区设置的事件的数量

连接的IE	总事件条件 (执行前 / 执行后 / 存取)	事件联系条件
IECUBE	34 (16 / 8 / 10)	2
MINICUBE	2 (1 ^a / 0 / 1)	—

a. 当使用软件中断时，该数字为 0。

可以通过从事件管理器区中拖动要设置事件的图标并放到本区域中轻松地设置事件条件。关于详细信息，可参见“5.12.3 设置事件条件”。

(3) 事件管理器：

此区域用于显示记录的事件列表。（参见“表 5-19：事件图标”“（4）事件管理器区中的操作”。）

功能按钮

参见事件管理器中的“功能按钮（相关事件功能）”。

截取对话框

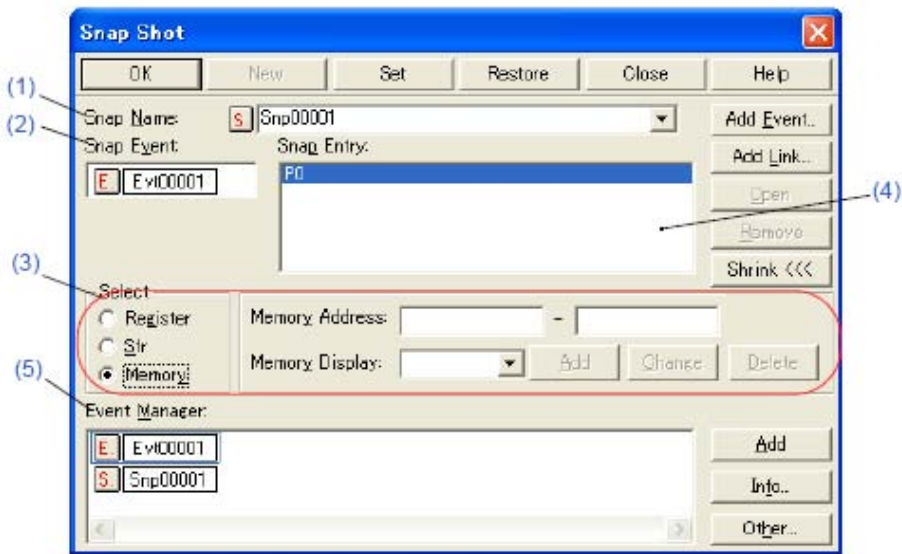
IECUBE

此对话框用于记录、设置和显示截取事件条件。（参见“5.12 事件功能”和“5.13 截取功能（当连接 IECUBE 时）”。）

可以通过在此对话框中设置每个项目（最多 256 个），然后点击<确定>按钮来记录和设置截取事件条件。记录的截取事件条件由事件管理器管理的。

可同时设置（启用）的截取事件条件的数量是有限的。（参见“5.12.4 每个事件条件的启用事件数”。）

图 6-59：截取对话框（当选择“寄存器”时）



- 打开
- 对每个区域的解释
- 功能按钮（相关事件功能）

打开

选择[事件]菜单 -> [截取...].

对每个区域的解释

(1) 截取名：

此区域用于设置截取事件名。直接输入不超过 8 个字符的数字字母串作为名称。要显示已经创建的事件条件的内容，可从下拉列表中选择。

此区域左边的标记表示事件的使用状态（参见“表 5-19：事件图标”）。灰色标记表示事件条件正在被编辑且

尚未被记录。可以点击左边的标记启用或禁用事件条件。

（2）截取事件：

此区域用于显示截取的事件条件。

关于此区域可以设置的事件条件和事件联系条件，参见“表 6-17：可设置事件的数量”。

可以通过从事件管理器区中拖动要设置事件的图标并放到本区域中轻松地设置事件条件。关于详细信息，可参见“5.12.3 设置事件条件”。

（3）选择：

此区域用于记录、修改和删除截取数据以及选择要截取的数据的类型。

通过选择一个数据类型，要在右边的截取数据设置区显示的项目根据选择的区域而变化。

作为截取数据，每个最多可以记录 16 个寄存器、SFR 和内存地址，因此总共可以记录 48 个寄存器、SFR 和内存地址。

在此区域内记录、修改或删除的截取数据反映在（4）截取条目：中。如果在（4）截取条目：中选择了截取数据，则选择的截取数据的内容在此区域内显示。

（a）寄存器

要设置寄存器作为截取数据，应选择此项目。

（i）寄存器名：（寄存器名设置区）

要指定一个寄存器名，可以直接输入到文本框或者从下拉列表中选择。

可以指定功能名或者绝对名为寄存器的名称。

备注 1：选择“全部”来指定所有的寄存器。

备注 2：可以指定通用寄存器和控制寄存器的名称作为寄存器名。

备注 3：跟踪视图窗口中显示的截取数据统一为小写字母并且为绝对名格式。

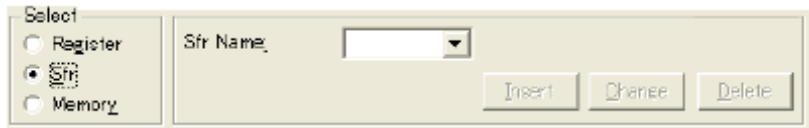
（ii）寄存器组：（寄存器组设置区）

要指定一个寄存器组，可以直接输入到文本框或者从下拉列表中选择（0-3）。

备注：要指定当前的组，选择“当前”。如果指定被省略，则会指定当前的组。

(b) Sfr

要设置 SFR 作为截取数据，应选择此项目。



(i) SFR 名：（SFR 名设置区）

要指定一个 SFR 名，可以直接输入到文本框或者从下拉列表中选择。

只能指定可以被读取的 SFR。区分大小写。

要注意的是，在跟踪视图窗口中寄存器名称是以大写字符显示的。

(c) 内存

要设置内存作为截取数据，应选择此项目。



(i) 内存地址：（内存地址设置区）

起始地址—结束地址

此区域用于指定内存的地址范围。如果仅指定了起始地址而省略了结束地址，则设定为结束地址与起始地址相同。如果地址范围无法被存取大小整除，则该范围会被调整为可以被存取大小整除的值。

也可以用一个符号或表达式指定地址。（参见“表 5-6：指定符号”。）输入数值的默认基数是十六进制。

如果通过使用表达式或者符号记录或修改此区域，则转换后的地址值与指定的表达式和符号一起显示在（4）截取条目：中。

在跟踪视图窗口，仅显示转换后的地址值。

表 6-25：地址可设置的范围（截取对话框）

范围
$0 \leq \text{地址值} \leq 0xFFFF$

(ii) 内存显示：（内存显示大小设置区）

要指定存取大小，可以直接输入到文本框或者从下拉列表中选择。区分大小写。

字节	B	以8比特为单位截取内存。
词	W	以16比特为单位截取内存。
双词	DW	以32比特为单位截取内存。

(d) 处理截取数据的按钮

以下按钮用于记录、修改和删除截取数据。

插入	记录截取数据。 插入记录的截取数据并显示在 (4) 截取条目：中选定的位置。
修改	将 (4) 截取条目：中选定的截取数据的内容修改为此区域中指定的截取数据的内容。
删除	删除 (4) 截取条目：中选定的截取数据。 DEL键可进行相同的操作。

(4) 截取条目：

此区域像是一个记录的截取数据列表。

当截取事件发生时，记录的截取数据被写入到跟踪器。

如果在此区域中选择了截取数据，则在每个设置区中显示选择的截取数据的内容。

截取数据显示如下：

表 6-26：截取数据显示格式

显示实例	内容
寄存器截取数据： RP0[0] RP[Current] All[2]	寄存器名[组号或‘Current（当前）’]
SFR截取数据： PM0 PM1	SFR名称
内存截取数据： 0xFE20<byVar>,B0xFE22<wVar>,W0xFE30<szVar> - 0xFE2F<szVar+0x10>,B 起始地址<符号表达式>-结束地址<符号表达式>, 存取大小	

(5) 事件管理器：

此区域用于显示记录的事件列表。（参见“表 5-19：事件图标”“（4）事件管理器区中的操作”。）

功能按钮

参见事件管理器中的“功能按钮（相关事件功能）”。

存根对话框

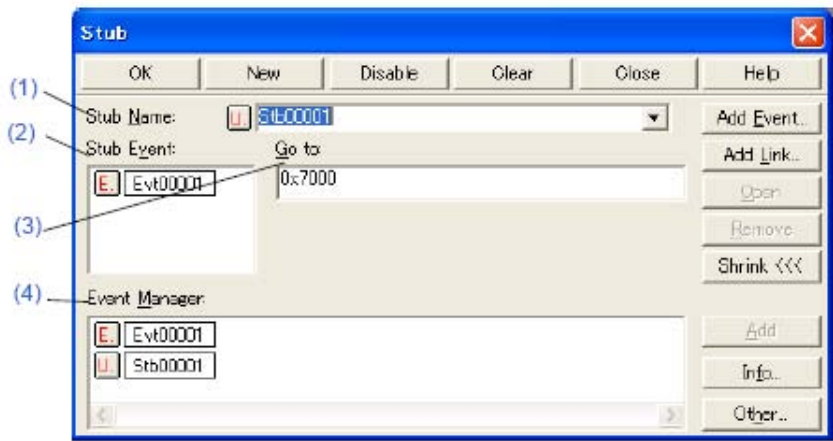
IECUBE

此对话框用于记录、设置和显示存根事件条件。（参见“5.12 事件功能”和“5.14 存根功能（当连接 IECUBE 时）”。）

可以通过在此对话框中设置每个项目（最多 256 个），然后点击<确定>按钮来记录和设置存根事件条件。记录的存根事件条件由[事件管理器](#)管理的。

可同时设置（启用）的存根事件条件的数量是有限的。（参见“5.12.4 每个事件条件的启用事件数”。）

图 6-60：存根对话框



- 打开
- 对每个区域的解释
- 功能按钮（相关事件功能）

打开

选择[事件]菜单 -> [存根...].

对每个区域的解释

（1）存根名：

此区域用于设置存根事件名。直接输入不超过 8 个字符的数字字母串作为名称。

要显示已经创建的事件条件的内容，可从下拉列表中选择。

此区域左边的标记表示事件的使用状态（参见“表 5-19：事件图标”）。灰色标记表示事件条件正在被编辑且尚未被记录。可以点击左边的标记启用或禁用事件条件。

（2）存根事件：

此区域用于显示存根的事件条件。

关于此区域可以设置的事件条件和事件联系条件，参见“表 6-17：可设置事件的数量”。

可以通过从事件管理器区中拖动要设置事件的图标并放到本区域中轻松地设置事件条件。关于详细信息，可参见“5.12.3 设置事件条件”。

（3）转到：

此区域用于指定当存根事件发生时指定的功能的起始地址。（参见“表 5-20：要执行的功能的起始地址（存根功能）”。）

输入数值的默认基数是十六进制。也可以用一个符号或表达式指定地址。（参见“表 5-6：指定符号”。）

（4）事件管理器：

此区域用于显示记录的事件列表。（参见“表 5-19：事件图标”“（4）事件管理器区中的操作”。）

功能按钮

参见[事件管理器](#)中的“功能按钮（相关事件功能）”。

事件 DMM 对话框

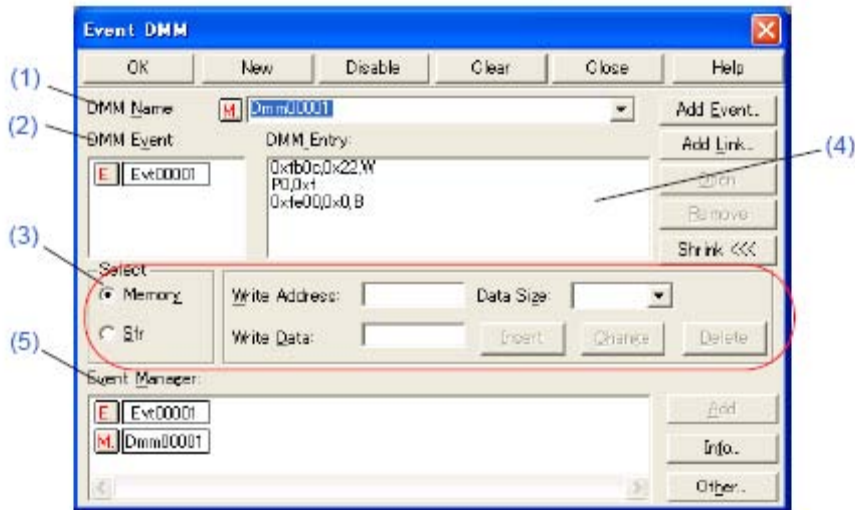
此对话框用于记录、设置和显示事件 DMM 条件。（参见“5.12 事件功能”和“5.16.1 事件 DMM 条件（当连接 IECUBE 时）”。）

可以通过在此对话框中设置每个项目（最多 256 个），然后点击<确定>按钮来记录和设置事件 DMM 条件。记录的事件 DMM 条件由[事件管理器](#)管理的。

可同时设置（启用）的事件 DMM 条件的数量是有限的。（参见“5.12.4 每个事件条件的启用事件数”。）

备注：每个事件 DMM 最多可记录 100 个 SFR 数据项目和内存数据项目。

图 6-61：事件 DMM 对话框（当选择“内存”时）



— 打开

— 对每个区域的解释

— 功能按钮（相关事件功能）

打开

选择[事件]菜单 -> [事件DMM...].

对每个区域的解释

(1) DMM 名：

此区域用于设置事件 DMM 名。直接输入不超过 8 个字符的数字字母串作为名称。要显示已经创建的事件条件的内容，可从下拉列表中选择。

此区域左边的标记表示事件的使用状态（参见“表 5-19：事件图标”）。灰色标记表示事件条件正在被编辑且尚未被记录。可以点击左边的标记启用或禁用事件条件。

(2) DMM 事件：

此区域用于设置事件 DMM 的事件条件。

在此区域内可设置的事件条件和事件联系条件数如下：

- 事件条件：18（执行：8、存取：10）
- 事件联系条件：2

可以通过从事件管理器区中拖动要设置事件的图标并放到本区域中轻松地设置事件条件。关于详细信息，可参见“5.12.3 设置事件条件”。

(3) 选择：

此区域用于在一个事件建立后记录、修改和删除 DMM 的目标数据（事件 DMM 数据）。

内存或 SFR 可以记录为事件 DMM 数据，对每个事件 DMM 最多可设置 100 个事件 DMM 项目。

通过选择一个数据类型，要在右边的截取数据设置区显示的项目根据选择的区域而变化。

在此区域内记录、修改或删除的截取数据反映在 (4) 截取条目：中。如果在 (4) 截取条目：中选择了事件 DMM 数据，则选择的事件 DMM 数据的内容在此区域内显示。

(a) 内存

此区域用于指定起始地址、要写入的数据值和一个事件建立后在写入数据到内存时的数据大小。

Select

Memory

Sfr

Write Address:

Data Size:

Byte

Write Data:

Insert

Change

Delete

写入地址：	指定起始地址。 也可以用一个符号或表达式指定地址。（参见“表5-6：指定符号”。）
写入数据：	指定数据值。
数据大小：	指定数据大小（字节、词）

(b) Sfr

本区域用于指定 SFR 名称和一个事件建立后在写入数据到 SFR 的数据值。

SFR名称：	指定SFR名称。
写入数据：	指定数据值。

(c) 操作事件 DMM 数据的按钮

以下按钮用于记录、修改和删除事件 DMM 数据。

插入	记录为事件DMM数据。 插入记录的事件DMM数据并显示在（4）截取条目：中选定的位置。
修改	将（4）截取条目：中选定的事件DMM数据的内容修改为此区域中指定的事件DMM数据的内容。
删除	删除（4）截取条目：中选定的事件DMM数据。 DEL键可进行相同的操作。

（4）截取条目：

此区域像是一个记录的事件 DMM 数据列表。

如果在此区域中选择了事件 DMM 数据，则在每个设置区中显示选择的事件 DMM 数据的内容。

截取数据显示如下：

表 6-27：事件 DMM 数据显示格式

显示实例	内容
内存数据： 0xFE20<byVar>, 0xFF, B 0xFE22<wVar>, 0xFF00FF00, W	起始地址<符号表达式>, 写入数据, 存取大小, 存取大小 （如果用符号或表达式指定，则用“<>”标识。）
SFR数据： P0, 0xFF TMS, 0x0000	SFR名称, 写入大小

（5）事件管理器：

此区域用于显示记录的事件列表。（参见“表 5-19：事件图标”“（4）事件管理器区中的操作”。）

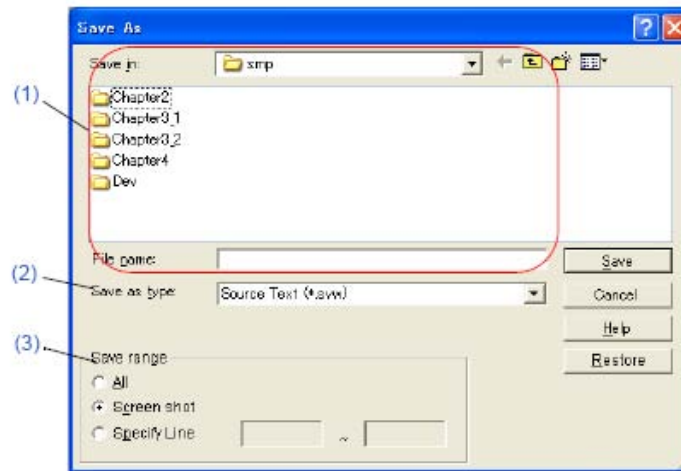
功能按钮

参见事件管理器中的“功能按钮（相关事件功能）”。

视图文件保存对话框

此对话框用于将当前窗口的当前显示信息保存到一个视图文件。（参见“5.17.2 窗口显示信息（视图文件）”。）

图 6-62：视图文件保存对话框



— 打开

— 对每个区域的解释

— 功能按钮

打开

当此窗口是当前窗口时，选择[文件]菜单 → [另存为...]

对每个区域的解释

（1）保存在（Save in）、文件名：

此区域用于指定要保存的文件名。可以直接通过键盘输入文件名或者在上面区域的文件列表中选择文件名。

指定的文件名最多可以包含257个字符，并可带有扩展名。

（2）保存类型：

此区域用于指定要保存的项目文件的类型（扩展名）。（参见“表5-23：视图文件的类型”。）

窗口会显示与当前窗口相应的文件扩展名。

(3) 保存范围

指定要保存的数据的范围

如果当前要保存的窗口如下，则会显示此区域：

—汇编窗口

—内存窗口

—代码覆盖窗口

—源窗口

—跟踪视图窗口

(a) 全部

选择此选项保存整个范围，从第一行到最后一行。

(b) 截屏

选择此选项保存在屏幕上可以看到的区域，从屏幕最顶部的一行到最底部的一行。如果源窗口为混合显示模式，会从包含屏幕中可视区域的源行中保存窗口内容。

(c) 指定行 / 指定帧 / 指定地址

选择此选项指定要保存区域的起始行和结束行。

如果省略了起始行和结束行，则认为起始行和结束行分别为第一行和最后一行。

如果指定了 100 行 / 100 帧 / 256 字节或以上的范围，会显示消息对话框以显示保存进度。要中途停止保存，点击消息对话框中的<停止>按钮。

根据当前的窗口显示以下其中一项：

指定行	指定要保存的行号范围。 输入数值的默认基数是十六进制。 如果源窗口为混合显示模式，还会保存指定行的混合显示部分。
指定帧	指定要保存的跟踪帧的范围。（参见“6-16：帧编号指定格式”。）输入数值的默认基数是十六进制。
指定地址	指定要保存的地址的范围。 可以用符号或表达式指定地址。（参见“6-16：帧编号指定格式”。） 输入数值的默认基数是十六进制。

功能按钮

保存	保存当前窗口的显示信息到选定的文件。在保存后关闭此对话框。
取消	关闭此对话框而不执行任何操作。
帮助	显示此对话框的在线帮助文件。
恢复	恢复到此对话框打开前的状态。

视图文件加载对话框

此对话框用于读取视图文件。（参见“5.17.2 窗口显示信息（视图文件）”。）

当加载一个视图文件时，会打开一个参照窗口（静止状态下的源窗口）并显示保存时的信息。

要打开的窗口及其状态变化如下，其主要取决于要加载的文件。

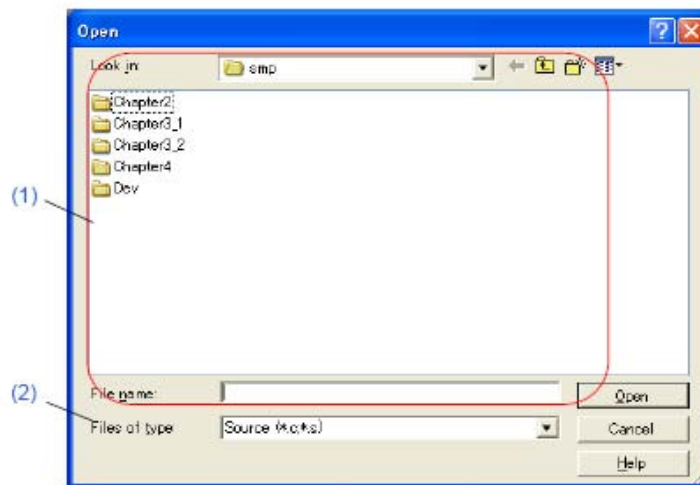
一加载已经读取符号信息到的源文件：

如果有一个活动状态的源窗口，则窗口以静态打开；否则源窗口以活动状态打开。

一加载还未读取符号信息到的源文件或者是视图文件：

以静止状态打开文本格式文件的源窗口。

图 6-63：视图文件加载对话框



一打开

一对每个区域的解释

一功能按钮

打开



打开Open按钮选择[文件]菜单 -> [打开...]

对每个区域的解释

(1) 查看 (Look in)、文件名:

此区域用于指定要加载的文件名。可以直接输入文件名或者在上面区域的文件列表中选择文件名。

指定的文件名最多可以包含257个字符，并可带有扩展名。

(2) 保存类型:

此区域用于指定要保存的项目文件的类型（扩展名）。（参见“表5-23：视图文件的类型”。）

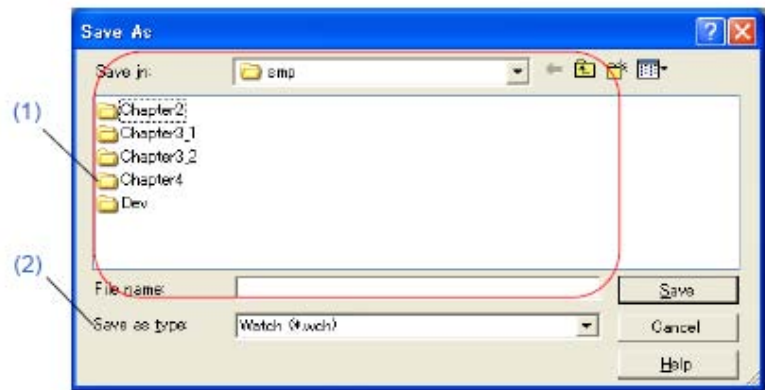
功能按钮

打开	加载一个选定的文件。加载后，此对话框关闭。
撤销	不执行任何操作而关闭此对话框。
帮助	显示在线帮助文件。

环境设置文件保存对话框

此对话框用于将当前窗口的设置内容保存到一个设置文件。（参见“5.17.3 窗口设置信息（设置文件）”。）

图 6-64：环境设置文件保存对话框



- 打开
- 对每个区域的解释
- 功能按钮

打开

当要保存的窗口是当前窗口时，选择[文件]菜单 -> [环境...]-> [另存为...]

对每个区域的解释

（1）保存在（Save in）、文件名：

此区域用于指定要保存的文件名。可以直接通过键盘输入文件名或者在上面区域的文件列表中选择文件名。指定的文件名最多可以包含257个字符，并可带有扩展名。

（2）保存类型：

此区域用于指定要保存的项目文件的类型（扩展名）。（参见“表5-24：设置文件的类型”。）窗口会显示与当前窗口相应的文件扩展名。

功能按钮

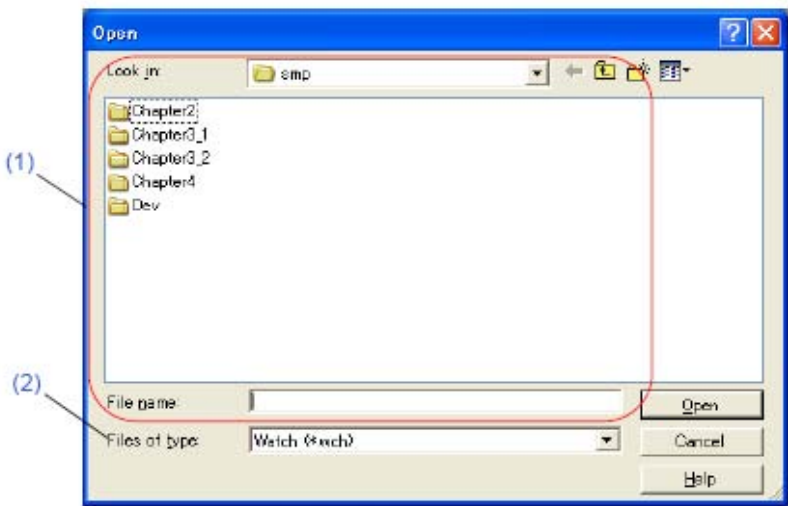
保存	保存当前窗口的设置信息到选定的文件。在保存后关闭此对话框。
取消	关闭此对话框而不执行任何操作。
帮助	显示此对话框的在线帮助文件。

环境设置文件加载对话框

此对话框用于加载一个设置文件。（参见“5.17.3 窗口设置信息（设置文件）”。）

当加载一个设置文件时，会打开目标窗口并恢复保存的设置信息。

图 6-65：环境设置文件加载对话框



- 打开
- 对每个区域的解释
- 功能按钮

打开

选择[文件]菜单 -> [环境...] -> [打开...]

对每个区域的解释

(1) 查看（Look in）、文件名：

此区域用于指定要加载的文件名。可以直接输入文件名或者在上面区域的文件列表中选择文件名。指定的文件名最多可以包含257个字符，并可带有扩展名。

(2) 保存类型：

此区域用于指定要保存的项目文件的类型（扩展名）。（参见“表5-24：设置文件的类型”。）

功能按钮

打开	加载一个选定的文件。加载后，此对话框关闭。
撤销	不执行任何操作而关闭此对话框。
帮助	显示在线帮助文件。

复位调试器对话框

此对话框用于初始化 ID78K0-QB、CPU 和符号信息。

图 6-66：复位调试器对话框



- 打开
- 对每个区域的解释
- 功能按钮

打开

选择[文件]菜单 -> [调试器复位...].

对每个区域的解释

(1) 复位主体选择区

此区域用于指定要初始化的主体。初始化选定的项目。

调试器	初始化ID78K0-QB（符号）。
符号	初始化符号信息。
目标CPU	初始化CPU。

功能按钮

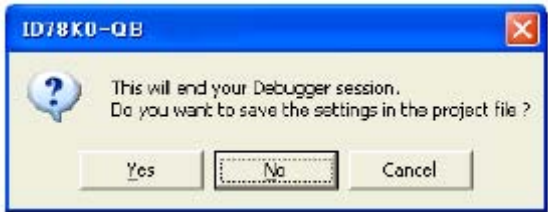
确定	根据设置初始化。
取消	关闭此对话框而不执行任何操作。
帮助	显示此对话框的在线帮助文件。

退出调试器对话框

此对话框用于选择是否要在退出 ID78K0-QB 前将当前调试环境保存到一个项目文件。（参见“5.17 调试环境（项目文件）”。）

可以在[调试器选项对话框](#)中指定 ID78K0-QB 不打开此确认对话框而直接退出。

图 6-67：退出调试器对话框



- 打开
- 功能按钮

打开

- 选择[文件]菜单 -> [退出]。
- 如果已经执行了任务列表中退出窗口的强制退出，如退出应用程序。

功能按钮

是	将当前的调试环境保存到一个项目文件，关闭所有的窗口，然后退出ID78K0-QB。 如果没有指定文件名，则打开 项目文件保存对话框 。如果选择了 项目文件保存对话框 中的<撤销>按钮，则不保存环境到项目文件也不退出ID78K0-QB。 （如果在调试器运行过程中读取或者加载项目文件，此按钮为默认焦点。）
否	关闭所有的对话框并退出ID78K0-QB。 （如果在调试器运行过程中读取或者加载项目文件，此按钮为默认焦点。）
取消	关闭此对话框而不执行任何操作。

关于对话框

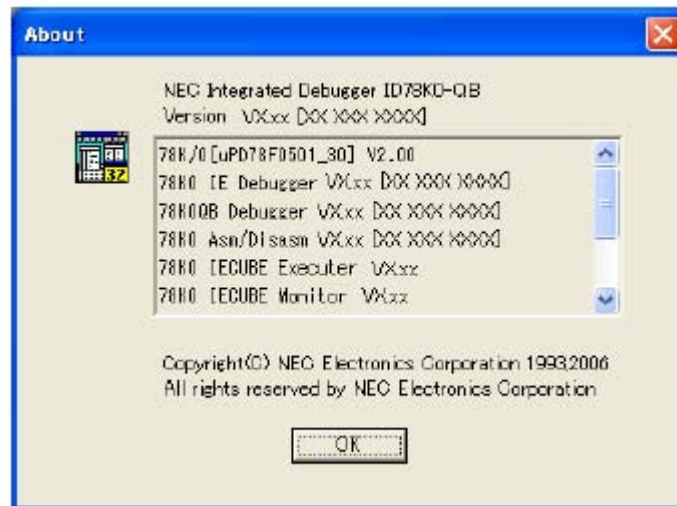
此对话框用于显示 ID78K0-QB 的版本信息（年份用四位数字显示）。

备注：无法通过选择右键菜单中的[全选+复制]从本对话框中复制版本信息到剪贴板。

显示的版本信息如下：

- ID78K0-QB 的产品版本
- 设备文件的版本
- GUI 的版本
- 调试器 DLL 的版本
- 汇编程序 DLL 的版本
- 执行程序的版本
- 监视程序的版本
- Tcl/Tk 的版本
- 产品 ID 和仿真器的产品版本
- OCD 控制代码版本（MINICUBE）

图 6-68：关于对话框



—打开

—功能按钮

打开

选择[帮助]菜单 -> [关于...]或者点击配置对话框中的<关于...>按钮。

功能按钮

确定	关闭此对话框。
----	---------

控制台窗口

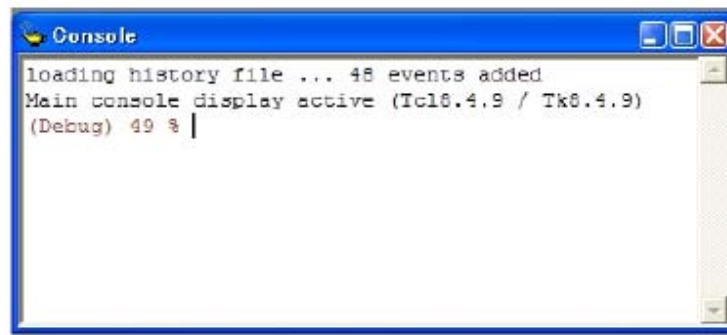
此对话框用于输入控制 ID78K0-QB 的命令。

由于键组合是与 Emacs 类似的，如果控制台窗口处于活动状态，快速键不被确认。而 F1 键可以显示控制台窗口在线帮助。

当控制台窗口打开时，在窗口中显示一个带有<确认>按钮的错误消息对话框。

关于命令的说明，请参见“第七章：命令参考”。

图 6-69：控制台窗口



—打开

—对每个区域的解释

打开

选择[浏览]菜单 -> [控制台]。

对每个区域的解释

关于命令的说明，请参见“第七章：命令参考”。

浏览对话框

此对话框用于选择在源文本移动对话框中设置的文件。

备注：如果在系统刚启动后第一次打开，则显示由源路径第一次指定的文件夹。当第二次以及以后打开对话框时，则记录和显示先前显示的文件夹。如果点击了<撤销>按钮，则不记录先前显示的文件夹。

图 6-70：浏览对话框



— 打开

— 对每个区域的解释

— 功能按钮

打开

在目标对话框中点击<浏览...>按钮。

对每个区域的解释

(1) 查看 (Look in)、文件名：

此区域用于指定要打开的文件名。可以直接输入文件名或者在上面区域的文件列表中选择文件名。

指定的文件名最多可以包含257个字符，并可带有扩展名。

(2) 保存类型：

此区域用于指定要打开的文件的类型（扩展名）。（参见“表5-5：可以显示的文件的类型”。）

功能按钮

打开	设置一个选定的文件。设置后，此对话框关闭。
撤销	不执行任何操作而关闭此对话框。
帮助	显示在线帮助文件。

第7章 命令参考

本章解释ID78K0-QB命令功能的细节。

- 命令行规则
- 命令目录
- 别名目录
- 变量目录
- 包目录
- 键绑定
- 扩展窗口
- 调回程序
- 钩程序
- 相关文件
- 注意事项
- 命令解释

7.1 命令行规则

- 命令行说明书有如下规则:
- 指定命令行的命令名、命令选项和命令意见。
 - 分开单词，使用空格（空格键或TAB键）
 - 在行的末尾，使用换行字符或分号。
 - 当命令名或选项输入到识别点，它们被认证。
 - 在脚本中，命令名必须要完整的输入。

命令格式

命令一选项 *arg1 arg2 arg3 ...*

7.2 命令目录

表7-1 Debugger控制指令目录

命令名	功能
地址	地址表达式的赋值
组装	分解/行组装
批	批执行（在屏幕上显示批处理命令行）
断点	设置/删除断点
Dbgexit	终结 ID78K0-QB
下载	文件下载
擦除	内部闪存的删除
Extwin	扩展窗口的创建
完成	从功能中返回
Go	持续执行
帮助	显示帮助
钩	设置钩
检查	符号检查
跳转	跳转到窗口
映射	设置/删除存储器交换
多文档界面	扩展窗口的设置
存储器	存储器的显示/设置
模块	文件和功能的显示
下一步	程序步骤
刷新	窗口屏幕刷新
寄存器	寄存器值和 SFR 值的显示/设置

复位	复位
运行	CPU 复位和运行
步骤	逐步执行
停止	停止执行
上传	上传信息
译本	译本的显示
监视	变量显示/设置
位置	栈跟踪
Wish	Tclet 的起动
X 覆盖	覆盖操作 (IECUBE)
X 计时	定时器操作 (IECUBE)
X 跟踪	跟踪操作(IECUBE)

表7-2 控制台/Tcl命令目录

命令名	功能
alias	其它名称的创建
cd	目录的改变
clear	屏幕刷新
echo	Echo 在屏幕上显示批处理命令行
exit	关闭/结束
history	历史显示
ls	文件显示
pwd	目录检查
source	批执行
time	命令测量时间
tkcon	控制台控制
unalias	删除其它名称
which	显示命令路径或不同的名称
Others	基于 Tcl/Tk 8.4

备注: 对于 Tcl 命令的细节, 选择 [启动] 菜单 -> [所有程序] -> [NEC 电子学工具] -> [ID78K0-QB] -> [VX.xx] -> [ID78K0-QB VX.xx Tcl8.4 帮助].

7.3 别名目录

命令可以和"bin/ldtcl/aliases.tcl"文件中定义的别名一起指定。
默认的别名显示如下。
这个文件可以用一个编辑器编辑。

表 7-3 别名.tcl 文件内容

别名集合
别名 b 断点
别名 g go
别名 i step -i
别名 j 跳转
别名 l 下载
别名 m 存储器
别名 n 下一步
别名 r 运行
别名 s step?
别名 w 观察

7.4 变量目录

表 7-4 变量目录

变量	功能
dcl (芯片)	只读芯片名称
dcl (项目文件)	只读项目文件名称
dcl (资源路径)	只读资源路径
dcl (ieid)	只读 IE 类型
dcl (iestat)	只读 IE 状态
dcl (bkstat)	只读中断状态
env (LANG)	语言
dcl_version	只读 Dcl 译本

7.5 包目录

表 7-5 包目录

包	功能
tcltest	复位?检验
cwind	自动窗口控制
BWidget	工具箱
tcllib	Tcl 库
mclistbox	多层柱目录框
combobox	组合框

7.6 键绑定

- tcsh + Emacs like
- 命令名补充 [Tab]
- 文件名补充 [Tab]
- HTML帮助 [F1]

7.7 扩展窗口

扩展窗口可以使用Tk创建。

在扩展窗口中，Widget被分配了“.dcl”代替“.”作为根。

当下列的脚本文件被分配到bin/ide/tcl/tools/中，在选择[浏览] 菜单 -> [其它]上增加了扩展窗口。已经增加了多文档界面、一个扩展窗口的执行命令。

```
# Sample.tcl
wm protocol .dcl WM_DELETE_WINDOW { exit }
mdi geometry 100 50
button .dcl.b -text Push -command exit
pack .dcl.b
```

注意事项:在扩展窗口中，因为MDI窗口的限制，所以Tk菜单命令不能被使用。

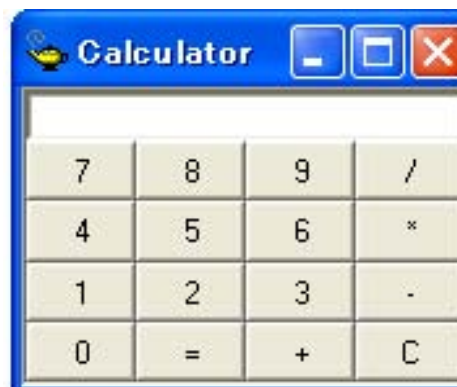
7.7.1 示例（计算器脚本）

描述计算器脚本的扩展窗体的脚本及其执行屏幕显示如下：

扩展窗口的脚本

```
# Calculator.tcl
mdi geometry 100 100
set top .dcl
entry $top.e -relief sunken -textvariable v
frame $top.f -height 120 -width 120; pack $top.e -fill x; pack $top.f -fill both -expand 1
set i 0; set v {}; set r 0.25
foreach n {7 8 9 / 4 5 6 * 1 2 3 - 0 = + C} {
    if {$n == "=" || $n == "C"} {
        button $top.f.b$n -text $n
    } else {
        button $top.f.b$n -text $n -command "$top.e insert end $n"
    }
    place $top.f.b$n -relx [expr ($i%4)*$r] -rely [expr ($i/4)*$r] -relw $r -relh $r
    incr i
}
bind $top.f.bC <1> {$top.e delete 0 end}
bind $top.f.b= <1> {catch {expr $v} v}
```

图7-1 执行屏幕



7.8 调回程序

扩展窗口可以用异步消息调用 dcl_asyncproc 程序。

```
proc dcl_asyncproc {mid} {
  if {$mid == 19} {
    redraw
  }
}
```

异步消息 ID 被看作是 dcl_asyncproc 程序的依据点。
这些消息 ID 显示如下：

表 7-6 消息 ID

消息 ID	意义
9	改变配置后
10	寄存事件后
11	删除事件后
12	执行前
13	断开后
14	CPU 复位后
15	ID78K0-QB 复位后
17	改变扩展选项后
18	改变调试器选项后
19	下载后
20	改变寄存器存储后
36	启动跟踪前(IECUBE)
37	停止跟踪后(IECUBE)
40	起动定时器前(IECUBE)
41	停止计时器后(IECUBE)
42	刷新跟踪后
45	复位符号后

7.9 钩程序

利用钩程序可以将一个钩设置进 ID78K0-QB。

钩程序显示如下：

- 下载前（钩在下载前）
- 下载后（钩在下载后）
- CPU 复位后（钩在 CPU 复位后）
- CPU 运行前（钩在起动作执行前）
- CPU 停止后（钩在 CPU 停止后）

通过使用钩程序，寄存器值可以在下载程序前或 CPU 复位后改变。

一个程序的实例显示如下。直到 ID78K0-QB 关闭钩都是有效的。

(1) [当钩用 ID78K0-QB 控制命令设置时]

- 1) 用编辑器创建脚本文件 a.。
- 2) 启动 ID78K0-QB，选择[浏览] 菜单 -> [控制台]，并打开[控制台窗口](#)。
- 3) 如果脚本文件在下列的窗口中执行，脚本文件中的钩被设置。
%hook test.tcl

(2) [当钩在下载的项目文件上设置时]

- 1) 用编辑器注释？文本创建脚本文件 a.。
- 2) 启动 ID78K0-QB 并读 test.prj。在脚本文件中的钩被设置。

```
proc BeforeDownload {} {
    register MM 0x7
    register PMC8 0xff
    register PMC9 0xff
    register PMCX 0xe0
}

proc AfterCpuReset {} {
    register MM 0x7
    register PMC8 0xff
    register PMC9 0xff
    register PMCX 0xe0
}
```

注意：确保脚本文件名和项目文件名相同。

例如：

脚本文件符合 test.prj 是 test.tcl。

在相同的文件夹中分配 test.prj, test.pri, and test.tcl

7.10 相关文件

表 7-7 相关文件目录

文件名	功能
aliases.tcl	当打开 aliases.tcl 控制台时执行。设置默认别名等。
项目文件名	当项目文件名.tcl 项目被打开时执行。 可以使用下列的钩。 下载前 下载后 CPU 复位后 CPU 运行前 CPU 停止后
加载模块文件名.tcl	当加载的模块名称.tcl 加载模块文件被下载时执行。 可以使用下列的钩。 下载前 下载后 CPU 复位后 CPU 运行前 CPU 停止后

7.11 注意事项

- 文件和路径的离析器被大量削减 (/)
- 当打开控制台，错误消息输出到控制台。
- 强制终止命令，关闭控制台。
- 外部命令的执行（DOS 命令）通过默认关闭。

7.12 命令解释

这部分中，每个命令使用如下显示的格式解释。

命令名

命令名描述

输入格式

命令的输出格式描述。
在下列解释中，斜体字表示用户提供的意见，当意见附上了"?"时可能被省略。
当命令名和一个选项输入到识别点时，它们被确认。

功能

命令功能的解释

使用样例

显示命令使用的样例。

地址

地址-地址表达式的赋值

输入格式

地址表达式

功能

通过指定的表达式将地址表达式转换成地址。

使用实例

(IDCON) 1 %地址主要部分 -----将主要部分转换成地址
0xaa
(IDCON) 2 % 地址主要部分+1 -----将主要部分+1 转换成地址
0xab

集合

集合-分解/行集合

输入格式

集合 ? 选项? 地址? 代码?

功能

字符串行集合有来自地址指定的地址代码说明。
当‘.’被地址说明时，它可以理解为来自立即分解的地址持续。
当代码省略时，它从指定的地址中分解。
如下是选项：由于行集合它们被忽视。

-代码	也显示命令代码。忽视行集合。
-数字 number	显示号码行。忽视行集合。

使用样例

(IDCON) 1 % assemble -n 5 main -----从主要部分指定的地址中分解 5 行。
0x000002ad PUSH HL
0x000002ae PUSH AX
0x000002af PUSH AX
0x000002b0 MOVW AX, SP
0x000002b2 MOVW HL, AX
(IDCON) 2 % assemble main mov a,b -----从主要部分中指定的地址中将 a,b 移到行集合。
(IDCON) 3 % assemble . mov a,b -----从序列地址中将 a,b 移入行集合。

批

批-执行批（在屏幕上显示批处理命令行）

输入格式

批 *脚本名称*

功能

用在屏幕上由*脚本文件*指定的显示文件在批中执行。
可以嵌套。

使用实例

(IDCON) 1 % clear -----清除控制台窗口。
(IDCON) 2 % batch bat_file.tcl ----- 为批文件.tcl 执行批处理。
(IDCON) 3 % tkcon 保存 a:/log.txt all
 -----将所以控制台窗口的缓冲数据保存到 a:/log.文本中。

断点

断点-设置/删除断点

输入格式

断点 ? 选项 ? 地址 1 ? 地址 2 ?
断点-删除中断号
断点-启动中断号
断点-废除中断号
断点-信息

功能

断点操作由选项和地址指定。
如果断点设置正确，中断号被返回。
下列是选项:

-软件	软件中断被详细说明。
-硬件	硬件中断被详细说明（默认）
-执行	设置地址执行中断（默认）
-执行前	在地址设置执行前中断
-读	设置地址数据读中断
-写	设置地址数据写中断
-访问	设置地址数据访问中断
-规格	设置访问规格（8，16 位）。(IECUBE)
-数据 值	设置数据条件
-数据掩码 值	设置数据掩码
-通道 值	设置通道数
-信息	显示断点目录
-删除	删除指定号码的断点
-废除	废除指定号码的断点
-启动	启动指定号码的断点

使用实例

(IDCON) 1 % 中断点主要部分-----设置主要部分的断点。

1

(IDCON) 2 % 断点-i -----显示断点目录。

1 Brk00001 enable rammon.c#17

(IDCON) 3 %中断点 -软件 sub -----设置软件 sub 的断点。

2

(IDCON) 4 % 中断点 -i -----显示中断点目录。

1 Brk00001 enable rammon.c#17

2 Brk00001 enable rammon.c#8

(IDCON) 5 % 中断点 -废除 2 -----废除 2 号中断点。

(IDCON) 6 % 中断点 -I -----t.显示中断点目录。

1 Brk00001 enable rammon.c#17

2 Brk00001 disable rammon.c#8

(IDCON) 7 % 中断点 -删除 1 -----删除 1 号中断点。

2 Brk00001 废除 rammon.c#8

dbgexit

dbgexit – 使 ID78K0-QB 终结

输入格式

dbgexit ?选项?

功能

终止 ID78K0-QB。
下列是选项:

-保存项目	项目保存在终止 ID78K0-QB 上。
-------	----------------------

使用实例

(IDCON) 1 % dbgexit -saveprj -----当 ID78K0-QB 被终结时保存项目。

下载

下载-文件的下载

输入格式

下载? 选项? 文件名? 偏移量?

功能

按照选项下载带指定文件名的文件。
如果指定偏移量，地址向上偏移（如果数据是二进制格式，加载启动地址是偏移量指定的）。

-二进制	下载二进制格式数据
-覆盖	下载覆盖数据
-非符号	执行下载。不读符号信息。
-只有符号	读符号信息。
-擦除	下载前，内部闪存的内容全部擦除（只有带内部闪存的产品 0
-复位	下载后，CPU 复位。
-信息	显示下载信息。
-64kb	64 位或更低的内存容量文件用这种格式下载。 这个选项是为下载二进制或十六进制格式文件指定的。
-bank	内存 bank? 文件用这种格式下载。

使用实例

(IDCON) 1 % 下载 test.lmf ----- 下载 test.lmf。

擦除

擦除-删除内部闪存。

输入格式

擦除

功能

擦除内部闪存。

使用实例

(IDCON) 1 %擦除 ----- 擦除内部闪存。

Extwin

Extwin-扩展窗口的创建

输入格式

Extwin *脚本文件*。

功能

创建带*脚本文件*的扩展窗口。

使用实例

(IDCON) 1 % extwin d:/foo.tcl -----使用 d:/foo.tcl 创建一个扩展窗口。

完成

完成-从功能处返回。

输入格式

完成

功能

直到返回到调用的当前功能的程序才执行。

使用方法

(IDCON) 1 %完成

----- 继续执行当前功能直到执行返回到调用程序。

转到

转到-继续执行

输入格式

转到 ? 选项?

功能

不断地执行程序。如果-等待中断被指定，命令一直等到程序停止。
下列是*选项*:

-忽视中断	中断被忽视。
-等待中断	命令一直等到程序停止。

使用实例

(IDCON) 1 % go -w ----- 等待直到程序停止。

钩

钩-钩设置

输入格式

钩 脚本文件

功能

设置带脚本文件的钩程序。
当项目文件被加载时和当 ID78K0-QB 被复位时钩设置被初始化。

使用实例

(IDCON) 1 % 钩 d:/foo.tcl -----使用 d:/foo.tcl 设置钩程序。

检查

检查-符号检查

输入格式

检查 ? 选项? 程序名模式

功能

利用表达式 模式搜寻和显示带程序名的加载模块符号。
可以用下列表达式。

?	匹配字符 1
*	除了 0 外，其它字符都匹配
[chars]	匹配字符型字符。（数列规格像[a-z/0-9]也可以。）
\x	匹配字符 x. (? * []\ 规格也可以。)

下列是选项:

-不区分大小写	区分大小写
-地址	成双地显示地址符号。

使用实例

(IDCON) 1 % 检查 test1.lmf {[a-z]*}
-----使用表达式 {[a-z]*} 搜寻和显示指定带 test1.lmf 的字符。

跳转

跳转-跳转到窗口

输入格式

跳转-资源-行 文件名? 行?
跳转? 选项? 地址

功能

由选项指定显示窗口。

-资源	从地址指定的地址显示源窗口。
-集合	从地址指定的地址显示集合窗口。
-存储器	从地址指定的地址显示存储器窗口。
-行	将命令移入指定的行。
-焦点	焦点移到显示窗口。

使用实例

(IDCON) 1 % jump -s main -----显示来自主要部分指定地址的源窗口。
(IDCON) 2 % jump -s -l mainfile.c 10 -----移到 主文件.c 的第 10 行。
(IDCON) 3 % jump -m array -----显示由数组指定地址的存储器窗口。

映射

映射 - 存储器映射的设置/删除

输入格式

映射选项 地址1 地址2 ?accsize?

功能

设置、删除和显示存储器映射。
Accsize? 指定通路规格为 8, 16, 或 32 (单位:字节, 默认是 8).
下列是选项:

- 对象	映射对象区域。
-堆栈	映射堆栈区域。(IECUBE)
-保护	映射 I/O 保护区域。
-rrm	设置 RRM 区域启动地址。 当 MINICUBE 被连接时, RRM 区域被划分。 在这种情况下, 启动地址和大小在下列的目录格式中被成双的指定。 {地址大小} {地址大小} {地址大小} ...} 大小是 1 到 16, 并且全部大小可以达到 16。
-刷新	删除所有的映射设置。
-信息	提交映射设置。

使用实例

(IDCON) 1 % map -i ----- 参考映射设置。
1: 0 0x7fff 8 {IROM}
2: 0x8000 0x87ff 8 {对象 RRM}
3: 0x8800 0x9fff 8 {对象}
4: 0xa000 0xf7ff 8 {NonMap}
5: 0xf800 0xfaff - {NonMap}
6: 0xfb00 0xfedf 8 {IRAM}
7: 0xfe0 0xfeff 8 {存储器}
8: 0xff00 0xffff 8 {SFR}

多文档界面

多文档界面-设置扩展窗口。

输入格式

多文档界面几何学?x y? 宽度 高度
多文档界面标题 字符串

功能

设置扩展窗口的大小和标题名。
只使用来自扩展窗口的命令。

使用实例

(IDCON) 1 % 多文档界面几何学 0 0 100 100
-----设置一个扩展窗口的位置和大小，匹配 0,0, 100-像素宽度，和 100-像素高度。
(IDCON) 2 % 多文档标题 foo ----- 设置标题"foo" 到扩展窗口。

存储器

存储器-显示/设置存储器。

输入格式

存储器?选项?地址?值?
存储器?选项?-填充地址1地址2值
存储器?选项?-复制地址 1 地址 2 地址 3

功能

根据选项设置存储器中由地址指定的地址值。
如果值省略，显示由地址指定的存储器地址值。
如果-指定填充，来自地址 1 和地址 2 的数据的被作为填充值。
如果-复制被指定，来自地址 1 和地址 2 的数据复制到地址 3。
下列是选项:

-字节	显示/设置一个-字节单位（默认）。
-单词	显示/设置单词单位。
-填充	数据被填充进来。
-复制	数据被复制。
-不校验	在写时校验不执行。

注释: 在用户执行程序时如果下列一个操作被执行，CPU 即刻被停止和继续执行。

- 除了 RRM 区域参考一个存储器区域 (MINICUBE)
- 在 RRM 区域中参考存储器区域 (MINICUBE)
- 除了内部 ROM/RAM 区域设置存储器区域 (IECUBE)
- 设置存储器区域 (MINICUBE)

使用实例

(IDCON) 1 %存储器 100 ----- 显示存储器中的地址值 0x100。
0x10
(IDCON) 2 %存储器 100 2 ----- 设置存储器中地址 0x100 到 2。
(IDCON) 3 %存储器 100 ----- 显示存储器中 0x100 地址的值。
0x02
(IDCON) 4 %存储器-填充 0 1ff 0 ----- 在存储其中用 0 填充地址 0x0 到 0x1ff 。

模块

模块

模块-显示文件和功能目录。

输入格式

模块 *程序名? 文件名?*

功能

显示程序名指定的文件目录和功能加载模块。
如果文件名没被指定，显示文件。
如果文件名被指定，显示指定文件的功能目录。

使用实例

(IDCON) 1 % 模块 rammon.lmf ----- 在 rammon.lmf 中显示文件目录。
1: rammon.c
(IDCON) 2 % 模块 rammon.lmf rammon.c ----- 在 rammon.lmf 中的 rammon.c 中显示文件目录。
1: rammon.c sub1
2: rammon.c main

下一步

下一步-程序进程

输入格式

下一步? 选项?

功能

执行程序步骤。如果功能被调用，执行功能后步骤停止。
下列是选项:

-资源	在资源行单元中执行命令（默认）。
-说明	在命令单元中执行命令。

使用实例

(IDCON) 1 % 下一步 -i -----在命令单元中用逐步的方式执行程序。
(IDCON) 2 % 下一步 -s ----- 在资源行单元中用逐步的方式执行程序。

刷新

更新-刷新窗口

输入格式

刷新

功能

刷新窗口并更新数据。

使用实例

(IDCON) 1 %批 foo.tcl	----- foo.tcl 批执行。
(IDCON) 2 % 刷新	----- 刷新窗口并更新数据。

寄存器

寄存器-显示/设置寄存器的值和 SFR 值

输入格式

寄存器? *选项?* *寄存器名?* *值?*

功能

在指定寄存器名的寄存器中设置值。
如果值省略，显示指定寄存器名的寄存器值。
下列是*选项*:

-强制	执行必须的读或写。
-存储号 <i>bankno</i>	指定存储号。

备注: 在用户执行程序时如果下列操作中的一个在执行，CPU立即被停止和继续执行。(当 IECUBE被连接时。)
- 设置寄存器或 SFR
-参考一个控制寄存器

使用实例

(IDCON) 1 % 寄存器 pc ----- 显示 PC 值。
0x100
(IDCON) 2 % register pc 200 ----- 设置 PC 到 0x200。
(IDCON) 3 % register pc ----- 显示 PC 值。
0x200

复位

复位-复位

输入格式

复位 ? 选项?

功能

复位 ID78K0-QB , CPU, 符号或事件。
如果选项被省略, CPU 被复位。
下列是选项:

-cpu	Cpu 被复位（默认）。
-调试器	ID78K0-QB 被复位。
-符号	符号被复位。
-事件	所有事件和软件中断被复位。

使用实例

(IDCON) 1 % 复位 ----- 复位 CPU。

运行

运行-CPU 复位和执行

输入格式

运行 ? 选项?

功能

复位程序并执行。
如果 -没有指定等待中断，直到程序停止才会等待命令。
下列是选项:

-等待中断	等待程序停止命令。
-------	-----------

使用实例

(IDCON) 1% 运行	----- 复位程序并执行。
(IDCON) 2 %运行 -w	----- 复位程序,执行直到程序停止。

步骤

--

步骤-步骤执行

输入格式

--

步骤 ? 选项 ?

功能

--

执行步骤。
如果功能被调用，命令在功能头部停止。
下列是选项:

-资源	执行命令行单元中的命令（命令）。
-说明	在说明单元中执行命令。

使用实例

--

(IDCON) 1 % 步骤 -i -----在资源行单元中逐行执行。
(IDCON) 2 % 步骤 -s -----在命令单元中逐步执行。

停止

停止-停止执行

输入格式

停止

功能

强制停止程序。

使用实例

(IDCON) 1 %运行 ---- 复位程序并执行。
(IDCON) 2 % 停止 ----强制停止程序。

上传

上传-上传

输入格式

上传 ? 选项? 文件名地址 1 地址 2
上传-覆盖文件名

功能

保存指定文件序列中的存储器数据。
当保存覆盖数据时，所有指定覆盖数据的数列被保存到文件中（不需要启动/结束说明书）。

注释: 当任何一个选项指定地址时数据用下列格式保存数据-binary, -intel, -motorola, or -tektronix用存储器存储合并的设备指定。

当末地址低于0x10000时:
或更低的存储器容量用这种格式保存。
当末地址低于0x10000时:
存储器存储用这种格式保存。

下列是选项:

- 二进制	数据用二进制格式保存。
-覆盖	保存覆盖数据。
-intel	数据用 Intel 十六进制保存（默认）。
-motorola	数据用 Motorola 十六进制格式保存。
-tektronix	数据用 Tektronix 十六进制格式保存。
-force	文件被覆盖。

使用实例

(IDCON) 1 %上传 -b foo.hex 0 0xffff -----从 0x0 到 foo.hex 用二进制格式保存。

译本

译本-显示译本信息

输入格式

译本

功能

显示 ID78K0-QB 译本。

使用实例

(IDCON) 1 %译本 -----显示 ID78K0-QB 译本。
GUI : VX.XX [XX XXXX 200X]
设备文件 : 78K/0[uPD78FXXXX] VX.XX
调试器 : 78K0-QB 调试器 VX.XX [XX XXXX 200X]
实行 : 78K0 MINICUBE Executer VX.XX
Micro program : OCD Control Code VX.XX
汇编程序 : 78K0 Asm/Disasm VX.XX[XX XXXX 200X]
Tcl/Tk : 8.4.X

监视

监视-显示/设置变量

输入格式

监视 ? 选项? 变量值?

功能

显示和设置变量。
下列是选项:

-二进制	用二进制数字显示值。
-八进制	用八进制数字显示值。
-十进制	用十进制数字显示值。
-十六进制	用十六进制数字显示值。
-字符串	用字符串显示值。
- sizeof	用十进制数字显示值或变量。
- 编码名称	在制定用字符串显示时编码。 默认, 使用系统编码。

使用实例

(IDCON) 1 % 监视 伏安 -----显示伏安值。
0x10
(IDCON) 2 %监视-d 伏安 ----- 用十进制符合显示伏安值。
16
(IDCON) 3 %监视{array[0]} 0xa ----- 设置 array[0] to 0xa.

位置

位置-堆栈跟踪

输入格式

位置

功能

执行堆栈的后面跟踪。

使用实例

(IDCON) 1 % 位置	----- 跟踪后面堆栈
1: test2.c#sub2(int i)#13	
2: test.c#num(int i)#71	
3: test.c#main()#82	

请求

请求-启动 Tclet

输入格式

请求 *脚本文件名称*

功能

利用 Tk (Tclet)启动脚本。
用 Tclet 创建扩展窗口。

使用实例

(IDCON) 1 % 请求 test.tcl ---- 启动 test.tcl。

X 覆盖

当 IECUBE 被连接

X 覆盖-覆盖操作

输入格式

X 覆盖 选项

功能

操作覆盖。
下列是选项:

-清除	清楚覆盖记忆。
-----	---------

使用实例

(IDCON) 1 % x 覆盖 -清除 ----- 清除覆盖存储。

X 计时

X 计时-定时器的操作

输入格式

X 计时 选项

功能

操作定时器。
下列是选项:

-启动	执行程序时定时器启动。(当 IECUBE 被连接)
-停止	执行程序时定时器停止。(当 IECUBE 被连接)
-中断	从进入到中断用 nsec 显示时间。

使用实例

(IDCON) 1 % x 计时 -启动 ----在程序执行时启动定时器。
(IDCON) 2 % x 计时 -停止 ----在程序执行时停止定时器。

X 跟踪

X 跟踪-跟踪器操作

输入格式

X跟踪 -倾卸 ?-添加? 帧数 ? 文件名称?
X跟踪-启动
X跟踪-停止
X跟踪-清除
X 跟踪-模式 ? 模式?

功能

操作跟踪器。
下列是选项:

-启动	程序执行时启动跟踪器。
-停止	程序执行时停止跟踪器。
-清除	清除跟踪存储。
-倾卸	跟踪数据被倾卸。(默认) 倾卸结果方向改变到控制台窗口。 如果指定文件名称, 倾卸结果写入文件。
- 添加	倾卸结果添加到文件。
-模式? 模式?	选择跟踪控制模式 (任何一个、全部、cond、不断的、 全部停止、全部中断、延迟停止, 延迟中断)。 当模式省略时, 显示最新模式。

使用实例

(IDCON) 1 % xtrace -启动 ----- 在程序执行时启动跟踪器。
(IDCON) 2 % xtrace -停止 ----- 在程序执行时停止跟踪器。
(IDCON) 3 % xtrace -dump 3 ----- 倾卸 3 帧跟踪数据。
_A 000015 0023B 9D M1 BNC \$_sub_全局的+0x2b
_A 000016 0023C 1F OP
_ 软件中断
(IDCON) 4 % x 跟踪 -清除 ----- 清除跟踪记忆。

tkcon

tkcon-控制台控制

输入格式

tkcon cmd ?arg?

功能

控制控制台窗口。
控制台/Tcl 中的一个命令（参考“表 7-2 List of 控制台/Tcl 命令目录”。）

tkcon 缓冲 ?大小?	设置和参考控制台最小缓冲大小（行数）。 如果超过指定的缓冲大小，超过的行从最先顺序中删除。
tkcon 关闭 tkcon 损坏	关闭控制台窗口。
tkcon 字体 ?字体名称?	在控制台窗口中设置和参考字体。
tkcon 获得	执行输入标准像 Stdin。 打开一个对话框。
tkcon 历史 ?-新行?	显示历史命令。
tkcon 保存 ?文件名称? ?类型	将控制台窗口缓冲数据作为一个文件保存。 当文件名和文件类型省略时，打开一个对话框。 从全部、history、 stdin、 stdout、 和 stderr 中选择类型。
tkcon 译本	显示控制台译本。

使用实例

(IDCON) 1 % tkcon 保存 c:/temp/logfile.txt all
-----将所有控制台窗口的缓冲数据内容保存入 c:/temp/logfile.txt。

附录A扩展窗口

- 概述
- 扩展窗口目录样例
- 激活
- 每个样例窗口的解释

A.1 概述

用 ID78K0-QB, 用户除了现有的窗口还可以创建用户窗口。
在ID78K0-QB中Tcl（工具命令语言）解释程序和控制调试器命令被执行。用户可以利用Tcl创建窗口。
ID78K0-QB由下列扩展窗口样例提供。

A.2 扩展窗口目录样例

表 A-1 扩展窗口目录(样例)

窗口名称	功能
目录窗口	显示资源目录和功能。
Grep 窗口	寻找字符串。
钩窗口	设置钩程序。
符号查找窗口	通过符号描述的目录查找

A.3 激活

扩展窗口可以通过选择[浏览]菜单上的[其它]中的目录、Grep、钩、符号查找来激活。

注释: 每个 .tcl 文件被安装进 NEC 电子工具\ID78K0-QB\Vx.xx\bin\idctl\tools

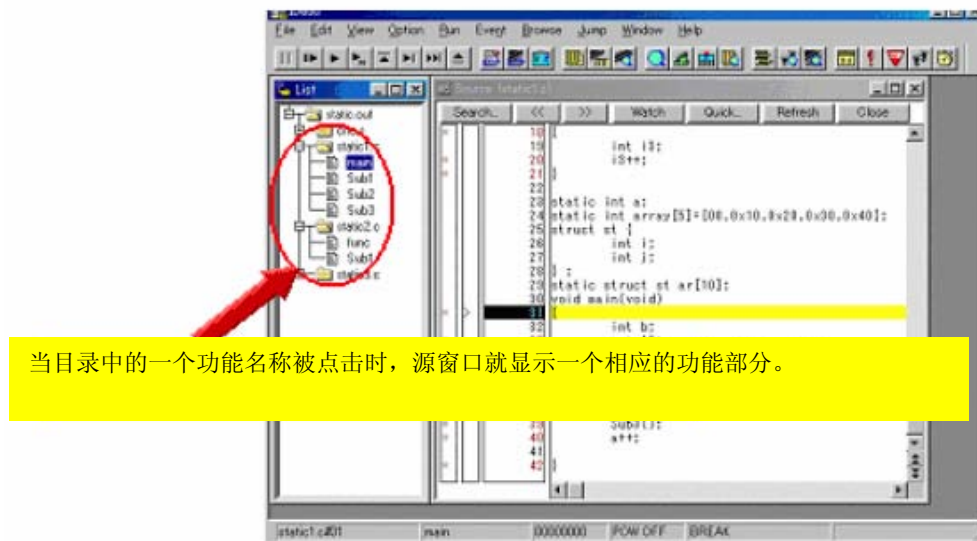
A.4 每个窗口样例的解释

The ID78K0-QB 提供了下列的样例窗口。

目录窗口

资源文件目录和功能在窗口中以树的格式显示。当目录中的功能名称被点击时，显示相应的资源。

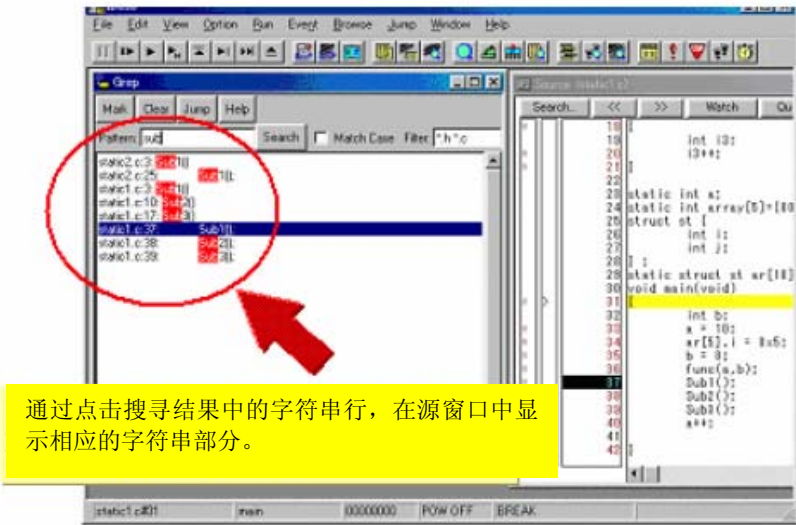
图 A-1 目录窗口



寻找字符串窗口

在资源路径下的文件中寻找字符串。
当点击搜寻结构时，显示相应的资源。

图 A-2 寻找字符串窗口



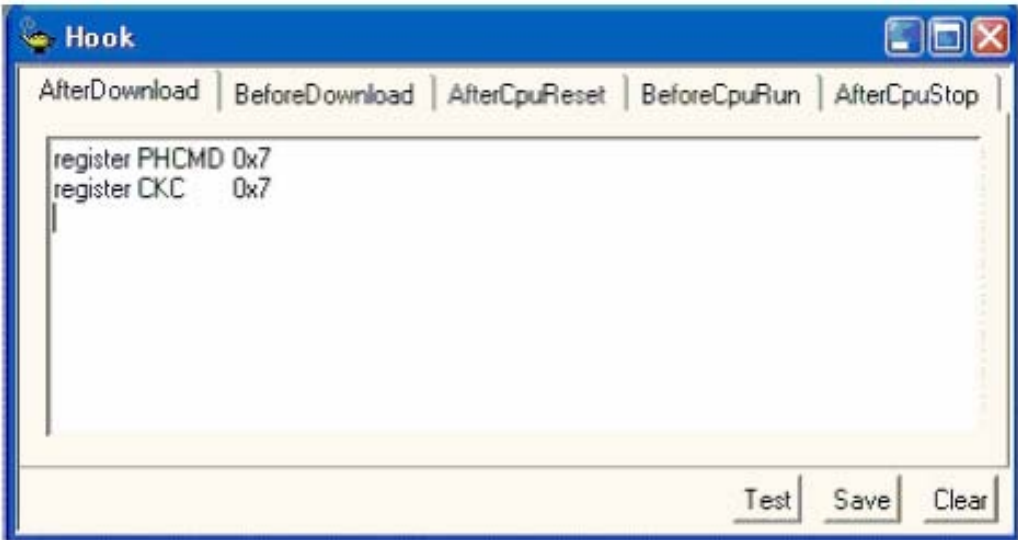
对象	功能
式样	输入被搜寻的字符串
<标志> 按钮	标志搜寻的字符串。
<清除>按钮	清除标志。
<跳转>按钮	将鼠标放在搜寻结果部分并点击这个按钮打开相应的文件。
匹配情况	选择是否区别向上情况和向下情况。
筛选	指定被搜寻的文件类型。

钩窗口

这个窗口利用钩程序给调试器设置一个钩。
下载程序前或 CPU 复位后钩程序启动改变寄存器值。
在这个窗口上，利用下列 5 个标签设置一个钩。
- [下载后] 标签: 下载后钩
- [下载前] 标签: 下载前钩
- [Cpu 复位后] 标签: 在中断时 CPU 复位后钩
- [Cpu 运行前] 标签: 启动执行前钩
- [Cpu 停止后] 标签: 中断后钩

如果设置作为项目存储文件夹中的“项目-文件- 名称.tcl”保存，当项目随后打开时执行设置。为寄存器名指定多种用途寄存器和 SFR。

图 A-3 钩窗口

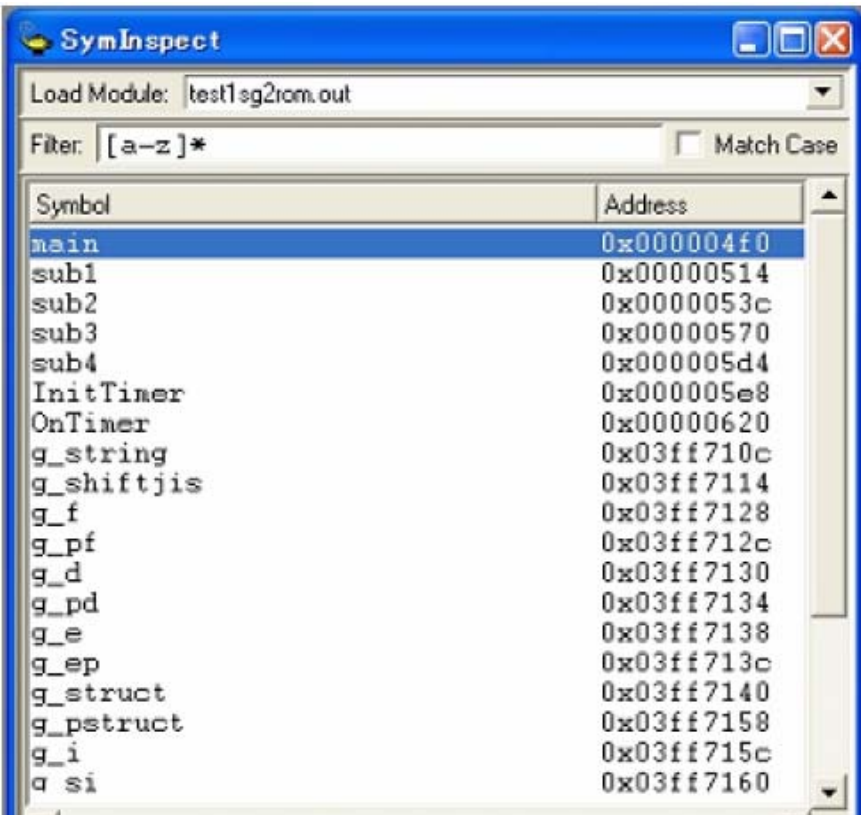


对象	功能
[下载后]标签	钩下载后。下载执行后，寄存器值输入标签被指定值自动覆盖。
[下载前] 标签	钩下载前。下载执行前，寄存器值输入标签被指定值自动覆盖。
[CPU 复位后] 标签	在中断时钩 CPU 复位后。CPU 复位后，寄存器值输入到标签被指定值自动覆盖。
[Cpu 运行前] 标签	钩在启动执行前。启动执行前，寄存器值输入到标签被指定值自动地覆盖。
[Cpu 停止后] 标签	钩中断后。中断后，寄存器值输入到标签被指定值自动地覆盖。
<测试>按钮	测试标签上描述的所有的命令。
<保存>按钮	将标签内容保存到一个文件。如果 ID78K0-QB 从项目文件中被激活，文件保存为"项目-文件-名称.tcl"
<清除>按钮	清除标签上的所有描述。

符号检查 窗口

这个窗口显示符号目录和加载模块文件的地址，并且用于搜索适当描述符号的目录。

图 A-4 符号检查窗口



对象	功能
加载模块	选择加载模块文件。
筛选:	指定合适的描述符号因此符号重新找到。
匹配情况	在筛选中: 指定区分或不区分敏感情况。选中这个框区分敏感情况。
符号	显示符号。点击这个图标有按字母顺序存储的符号。
地址	显示这个地址。点击这个图标有按数字升序存储的符号。

内容菜单	功能
复制	复制选择的地址到剪切板。
跳转到资源	从选择行中的地址跳转到在源窗口中显示同样的地址。
跳转到集合	从选择行中的地址跳转到在汇编窗口中显示同样的地址。
跳转到存储器	从选择行中的地址跳转到在存储器窗口中显示同样的地址。

附录B 输入惯例

- 可用的字符设置
- 符号
- 数字的值
- 公式和操作员
- 文件名

B.1 可用的字符设置

表 B-1字符设置目录

分类	字符
字母字符	大写字母: A B C D E F G H I J K L M N O P Q R S T U V W X Y Z 小写字母: a b c d e f g h i j k l m n o p q r s t u v w x y z
数字	0 1 2 3 4 5 6 7 8 9
字符等同字母字符	@ _ ?

表 B-2 指定字符目录

字符	名称	用途
(	左括号	改变操作顺序。
)	右括号	改变操作顺序。
+	加号	加操作或正号
-	减号	减操作或负号
*	乘号	乘操作或间接参考操作
/	除号	除操作
%	百分号	余操作
~	Tilde	补足操作
	竖线	位求和操作
^	抑扬符号	位异操作
&	&(=and)的记号名称	位产生操作或地址操作
.	句点	直接成员操作或位位置说明
,	逗号	操作数间的分界符

字符	名称	用途
[	左括弧	写在数组下方的操作或间接显示符号
]	右括弧	
!	感叹号	绝对地址启动符号
\$	Dollar	相对地址启动符号
#	Sharp 标志	符号表示一个立即值

B.2 符号

(1)符号由后面的字符组成。

A to Z, a to z, @, _ (underbar), . (period) and 0 to 9

- (2)符号除了以数字 0 to 9 开头还以字符开头。
- (3) 区分大写字母 (A to Z) 和小写字母 (a to z)。
- (4) 符号的长度不超过 2048 字符 (如果符号定义超过 2048 字符,只有前 2048 个字符有效)。
- (5)通过装载加载模块文件定义符号。
- (6)通过有效范围将符号分类为下列类型:

全局符号(汇编语言, 结构汇编语言, C 语言)
静态符号(C 语言)

- 在-文件静态符号中
- 在-功能静态符号中

局部符号(C 语言)

- 在-模块局部符号中 (汇编语言, 结构汇编语言)]
- 在-文件局部符号中
- 在-功能局部符号中
- 在-封锁局部符号中

(7) 每种语言都可使用下列符号:

汇编语言, 结构汇编语言
标签名称, 常数名称, 位符号名称

C 语言
变量名 (包括指针变量名, 列举类型变量名, 数组名,结构体名, 和共同体名)
功能名, 标签名
数组元素, 结构体元素, 共同体元素,位域 (如果符号是一个数组、结构体、或共同体)

- (8)可以描述符号代替地址或数字值。
- (9)当资源文件被汇编或编译时符号的有效范围基于资源调试工具信息决定。
- (10)只描述全符号的符号名称。
- (11)局部符号用文件名表达。

B.3 数字值

可以用下列4种类型数字值。每种类型输入格式显示如下。
16进制数字的后缀（黑体）和字母字符可以是大写字母和小写字母。如果第一个字符是A到F, 0必须作为它的前缀。
在 ID78K0-QB的输入区域内,依赖默认基数，可以交替选择十六进制和十进制。

表 B-3 数字值输入格式

数字值	输入格式
二进制数字	nY n...nY (n=0,1)
八进制数字	nO n...nO (n=0,1,2,3,4,5,6,7) nQ n...nQ (n=0,1,2,3,4,5,6,7)
十进制数字	n n...n nT n...nT (n=0,1,2,3,4,5,6,7,8,9)
十六进制数字	n n...n nH n...nH 0xn 0xn...n (n=0,1,2,3,4,5,6,7,8,9,A,B,C,D,E,F)

B.4 表达式和操作符

(1) 表达式

一个表达式包括常数、寄存器名、SFR名和操作数连接符。
如果SFR名、标签名、功能名或变量名描述为一个符号，地址被计算为符号值。
构成表达式的元素，除了操作数，被称为条件（常数和标签）。条件在第一时期、第二时期等期间被调用，从左边开始。

(2) 操作符

C 语言操作符可以使用:

表 B-4 操作数目录

符号	意义	解释
算术操作		
+	加	返回第一和第二期间的和
-	减	返回第一和第二期间之间的差
*	乘号	返回第一和第二期间的乘积
/	除号	第一期间除以第二期间的值，并返回结果的整数值。
MOD %	求余	第一期间除以第二期间的值，并返回结果的余数值。
- sign	一元操作（负）	返回 2's 期间的补充值。
+ sign	一元操作（正）	返回期间值。
逻辑操作		
NOT ~	否定	每个位期间逻辑否定，返回结果。
AND &	逻辑乘积	获得每个位上第一和第二期间的逻辑乘积值，返回结果。
OR 	逻辑和	获得每个位上第一和第二期间的逻辑和值，返回结果。
XOR ^	高级逻辑和	获得每个位上第一和第二期间的高级逻辑和值，返回结果。
移位操作		
SHR>>	右移	移动第二期间向左移到第一期间的值（数字位），并返回结果。移位 0 的多少插入高位。
SHL <<	左移	移动第二期间向左移到第一期间的值（数字位），并返回结果。移位 0 的多少插入低位。
位分离操作		
高	高字节	这个期间的最高 16 位，返回高 8 位。
低	低字节	这个期间的最低 16 位，返回低 8 位。
词分离操作		
HIGHW	更高词	这个期间的 32 位，返回高 16 位。
LOWW	更低词	这个期间的 32 位，返回低 16 位。

符号	意义	解释
其它		
(	左圆括号	先执行 () 内再执行 () 外面。'(' and ')' 总是成对使用。
)	右圆括号	

(3)操作规则

根据操作符的优先权执行操作。

表 B-5 操作数优先权

优先权	操作数
1 高位	(,)
2	+ sign, - sign, NOT, ~, HIGHT, LOW, HIGHW, LOWW
3	*, /, MOD, %, SHR, >>, SHL, <<
4	+, -
5	AND, &
6 低位	OR, , XOR, ^

- 如果操作数的优先权相同，从左向右执行。
- 先执行 () 内再执行 () 外面。
- 每个操作期间被看作是无符号32-位数字。
- 所有操作被看作是 32-位数据。
- 再操作时如果发生溢值，低32位有效，并且不发觉溢值。

(4) 期间

描述一个期间的常数，数字值描述如下。

表 B-6 基数范围

基数	范围
二进制数	0Y <= 值 <= 11111111111111111111111111111111Y (32 digits)
八进制数	0O <= 值 <= 37777777777O
十进制数	-2147483648 <= 值 <= 4294967295 (A negative decimal number is internally converted into a 2's complement.)
十六进制数	0H <= 值 <= 0FFFFFFFH

B.5 文件名称

下列规则应用到资源文件名和加载模块文件名。

(1) 资源文件名和加载模块文件名

文件名由a 到 z, A 到 Z, 0 到 9, ., _ , +, and -组成。

除了 "."文件名还可以以字符启动。

文件名不可以用(.)或空格做前缀或后缀。

文件名不区分大小写。

一个文件名和路径可以达到 259 个。

(2) 其它文件名

其它文件名遵守 Windows 文件名规则。

下列的字符不可以用作文件名。

\\ : * ? " < > | ;

文件名不可以用(.)或空格做前缀或后缀。

文件名不区分大小写。

一个文件名和路径可以达到 259 个。

附录C 键功能目录

表 C-1 键功能目录

键	功能
BackSpace	删除指针前的一个字符和指针移到位置的字符。 这时，指针后面的字符串前移。
Delete	-删除指针后的一个字符和指针后的字符串前移。 -在事件管理器或每个事件对话框中删除一个不同的事件条件。 -在监视窗口中选择删除数据。
Insert	在源窗口和汇编窗口中和交替地选择插入模式和覆盖模式。然而，这个键在存储器、寄存器和 IOR 窗口中无效，并且覆盖模式仅仅用作一个输入模式。
PrintScreen	加载整个显示屏幕到剪切板作为位图图象（窗口功能）。
Esc	-关闭下拉菜单。 -关闭模式对话框。 -存储输入数据。
Alt	将指针移到菜单栏。
End	将指针移到行末尾。
Home	将指针移到行开头。
PageUp	屏幕上翻。指针也可以移到屏幕顶端。
PageDown	屏幕下翻。指针也可以移到屏幕顶端。
Space	插入空白字符。
Tab	将指针移到下一个事件。
向上箭头键	将指针移到上面。 如果指针在屏幕底部，一次可以向上翻一行。
向下箭头键	将指针移到下面。 如果指针在屏幕顶端，一次可以向下翻一行。
向右箭头键	将指针移到左边。 如果指针在屏幕最左端，向右翻一列。
向左箭头键	将指针移到右边。 如果指针在屏幕最右端，向左翻一列。
Enter	-设置输入数据。 -按压默认按钮。
F1	打开帮助窗口。

键	功能
F2	强制停止程序执行。相同功能像[运行] 菜单 -> [停止]。
F3	复位 CPU。相同功能像[运行] 菜单 -> [CPU 复位]。
F4	复位 CPU 并执行程序。相同功能像[运行] 菜单 -> [重启]。
F5	执行程序。相同功能像[运行] 菜单 -> [Go]。
F6	在资源或汇编窗口中执行程序到指针位置。相同功能像[运行] 菜单 -> [来到这儿]。
F7	用户程序实时执行直到程序返回。 相同功能[运行] 菜单 -> [返回输出]。
F8	步骤执行。相同功能像[运行] 菜单 -> [步骤 In]。
F9	在资源或汇编窗口中设置中断点指针位置。 相同功能像[运行] 菜单 -> [中断点]。
F10	下一步执行。相同功能像[运行] 菜单 -> [Next Over]。
F11	设置或删除一个软件中断点。 相同功能像[运行] 菜单 -> [软件中断点]。
Shift+End	扩展行末尾的选择范围。
Shift+Home	扩展行开头的选择范围。
Shift+左箭头	扩展一个向左字符的选择范围。
Shift+右箭头	扩展一个向右字符的选择范围。
Shift+F6	在资源或汇编窗口中从指针位置处执行程序。 相同功能像[运行] 菜单 -> [从这儿启动]。
Shift+F9	复位 CPU。 相同功能像[运行] 菜单 -> [CPU 复位]。
Ctrl+End	显示最后行。指针也移到最后行。
Ctrl+Home	显示第一行。指针也移到第一行。
Ctrl+左箭头键	将指针向左移动一个单词。 如果指针在屏幕的最左边，向右移动一列。
Ctrl+右箭头键	将指针向右移动一个单词。 如果指针在屏幕的最右边，向左移动一列。
Ctrl+F5	忽视中断点设置，并执行程序。 相同的功能像[运行] 菜单 -> [忽视中断点并 Go]。
Ctrl+F9	在源窗口或汇编窗口中设置指针位置处的地址。 相同功能像[运行] 菜单 -> [改变 PC]。
Ctrl+A	全选所有事件。 相同功能像在事件管理器中的[监视] 菜单 -> [选择所有事件] 。
Ctrl+C	复制一个选择的字符串并保存到剪切板缓冲区内。

键	功能
Ctrl+D	分解和显示来自在当前窗口中由选择的数据值指定的跳转目的地址的结果。打开汇编窗口。相同的功能像[跳转] 菜单 -> [汇编]。
Ctrl+E	当 PM 插件运行时, 打开由 PM 插件指定的带编辑器的显示在激活源窗口中的资源文件。 向同的功能像[编辑] 菜单 -> [编辑资源]。
Ctrl+G	执行一个搜索。打开搜索对话框到相应的当前窗口。 相同的功能像[观看] 菜单 -> [搜索...].
Ctrl+J	移动显示位置。依赖当前窗口打开每个对话框。 [观看] 菜单 -> [移动...].
Ctrl+M	显示来自在当前窗口中由选择的数据值指定的跳转目的地址的存储内容。 相同的功能像[跳转] 菜单 -> [存储...].
Ctrl+O	加载一个视图文件、资源文件、或文本文件。打开视图文件加载的对话框。依赖文件的范围操作将有差异。 视图文件：显示相应窗口中的文件。 其它：显示源窗口中的文件。 相同功能像[文件] 菜单 -> [打开...].
Ctrl+S	将显示在当前窗口的数据保存到视图文件。 相同的功能像[视图] 菜单 -> [保存...].
Ctrl+U	利用当前窗口中选择的数据值作为跳转目的地址，显示相应资源文本和资源行。 打开源窗口。 相同的功能像[跳转] 菜单 -> [资源文本]。
Ctrl+V	粘贴剪切板缓冲区的内容到文本指针位置。
Ctrl+W	临时显示指定数据的内容。打开快捷监视监视对话框。相同的功能像[视图] 菜单 -> [快速监视...].
Ctrl+X	剪切选择的字符串保存到剪切板缓冲区。相同的功能像[编辑] 菜单 -> [剪切]。
Ctrl+Shift+左箭头键	向左扩展一个单词的范围。
Ctrl+Shift+右箭头键	向右扩展一个单词的范围。

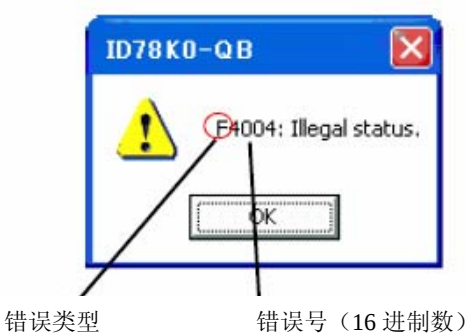
附录D 消息

- 显示格式
- 消息类型
- 消息目录

D.1 显示格式

消息输出到错误/警告对话框。
当错误/警告对话框打开时按 F1 键，在线显示相关的帮助文件。

图 D-1 错误/警告消息



D.2 消息类型

ID78K0-QB 输出下列消息类型。

表 D-1 消息类型

类型	意义
Axxxx	A: 异常中断错误 停止处理并结束调试。 如果发生这个错误，不能继续调试。
Fxxxx	F:失败 停止处理并且打开的窗口和对话框关闭。
Wxxxx	W: 警告。停止处理，但打开的窗口和对话框不关闭。

D.3 消息目录

<来自 X0000 > <来自X1000 > <来自X2000 > <来自X3000 > <来自X4000 > <来自X5000 > <来自X6000 >
 > <来自X7000 > <来自X8000 > <来自X9000 > <来自Xa000 > <来自Xb000 > <来自Xc000 > <来自Xd000 >
 > <来自Xe000 > <来自Xf000 >

(1) 来自 X0000

F0002:	不支持这个特征。
F0100:	不能和 ICE 通话。。请确认 PC 接口板驱动设备是否安装。 1)驱动器没有正确地安装。重装驱动器。
A0101:	没建立安装文件 (expc.ini)。
A0102:	没建立主机名。
F0103:	数据传输到 ICE 被终止。请确认 ICE 电源, 和接口电缆的连接,或 PC 接口面板 I/O 地址。
F0104:	ICE 接受数据被终止。请确认 ICE 电源, 和接口电缆的连接,或 PC 接口面板 I/O 地址。
A0105:	读设备文件失败(d1xxx.78k)。 1) 会损坏必要的设备。重装设备文件。
A0106:	接受非法数据。 1)检查电路竞争电源, 电缆连接,和接口面板设置并重启调试器。
A0107:	不能和 ICE 通话。
A0108:	读初始化文件失败 (expc.ini)。
A0109:	不能和 ICE 通话。请停止调试并检查电源、电缆连接,和接口面板设置并重启调试器。 1)在 USB 通讯时可能已经发生错误(像电源或电路没有连接号) 或 IECUBE 有故障。(IECUBE)
F010a:	不能通信。请确认通信接口是否可用。
A01a0:	来自竞争 CPU 没有响应。请确认时钟或复位信号等等。 1)检查 HOLD 信号, WAIT 信号, clock s 信号,等等。 IOR 值 (或 SFR 值) 可能不正确。
A01a1:	读 EX78K4.OM0 失败。
A01a2:	没有连接中断板。
A01a3:	没有连接竞争板。
A01a4:	ICE 板配置不兼容。
A01a5:	没有连接 POD/EM1 板。
A01a6:	运行执行者。

A01a7:	读 micro 程序文件失败 (m0xxx.78k)。
A01a8:	读初始化文件失败(expc.ini)。
A01ad:	请更新 PC 接口板的驱动器设备。 1)驱动器设备可能老化。安装最新的驱动器设备。
A01ae:	读初始化文件失败(expc.ini)。
A01af:	执行监控命令失败。
A01b0:	不能和监控程序通话。请检查可用通信口, CPU 板设置或电缆类型。
A01b1:	不能和监控程序通话。请终止调试并检查 CPU 板电源或电缆连接然后重启调试器。
F0200:	确认发生的错误。写存储失败。 1) 外存储器不能访问, 就像没有设置。为访问外存储器在下载前利用 SFR 窗口 或 钩程序 改变必要的寄存器值。
F02a0:	Bus 支持错误。 1) CPU 在 bus-hold 状态。复位调试器。
F02a2:	不能强迫中断。
F02a3:	持续复位。
F02d2:	跟踪-缓冲器没有足够的存储空间。
F0300:	运行用户程序。
F0301:	中断用户程序。
F0302:	跟踪用户程序。
F0303:	不跟踪。
F0304:	跟踪存储器没有设置。
F0306:	不存在跟踪阻塞。
F0307:	不存在事件条件。
F0308:	没有使用定时器测量。
F0309:	不存在触发帧。
F030a:	跟踪器被停止。
F030b:	指定的突发-事件没有注册。
F030c:	指定的 stub-event 没有注册。
F030d:	运行定时器。
F030e:	存储器复制区域交迭。
F030f:	跟踪已经设置。
F0310:	事件条件没有设置。
F0311:	太多有效定时器事件条件。
F0312:	指定的定时器事件没有设置。

F0313:	非法 map 范围。 在配置对话框中检查 map 范围。当映射到外部存储器已经执行时，下载前利用 SFR 窗口或钩程序改变必要的寄存器值。
F0314:	只有跟踪延迟模式可以用延迟触发器设置。
F0315:	没有跟踪延迟模式的情况下，不能用跟踪延迟触发器设置。
F0316:	映射的溢出数字。
F03a0:	没有打开目标。 1)检查目标电源供应。检查电路竞争中的和目标板的电缆连接。 检查输出到目标板连接器的 VDD 信号。
F03a1:	步骤执行被操作。
F03a2:	运行定时器和跟踪器。
F03a3:	混合事件连接和 BRS 事件。
F03d0:	执行后-跟踪。
F03d1:	停止后-跟踪。
F03d2:	后-跟踪执行点超出最旧帧。
F03d3:	除了事件连接的相位 1 寄存器状态或存储器状态不可能被设置。
F03d4:	不存在后-跟踪信息。
F03d5:	最新命令不能回退。
F0400:	非法条件。 1)电路中竞争者的使用设置和对话框配置将不匹配。检查芯片选择。
F0401:	定时器测量结果溢出。
F0402:	过多带路径数的事件条件。
F0403:	过多地址范围。
F0404:	过多同时-使用-事件条件。
F0405:	过多突发-事件。
F0406:	过多突发-事件。
F0407:	过多初始化数据。
F0408:	过量搜寻数据(> 16 byte).
F0409:	过量搜寻数据(> search range).
F040a:	过多连接-事件条件。
F04a0:	软件中断条件数字溢出。
F04a1:	没有足够的竞争存储器。
F04a2:	大小的过多划分。
F04a3:	过多执行-事件条件。
F04a4:	过多 bus-事件条件。

A0600:	缓冲区没有足够的存储空间。 1) 没有足够的系统存储空间。关闭正在执行的应用软件并打开文件。
A0601:	操作系统没有足够的资源。
F0b20:	不能使用事件号。
F0b61:	部分跟踪事件条件溢出。
F0b66:	在执行事件前不能使用中断和此时的软件中断。 1) 在执行前因为一个中断用于执行软件中断(当 MINICUBE 被连接时)。
F0b80:	通过硬件错误复位。
F0c00:	监控文件读错误。 1) 必要的文件可能损坏。重装调试器。
A0c01:	在访问寄存器时, CPU 终止。 1) 检查时钟信号, 等等。寄存器值可能没有不正确。
A0c02:	在访问存储器时, CPU 终止。 1) 检查 HOLD 信号, WAIT 信号, clock 信号, 等等。存储器值可能不正确。
A0c03:	在访问 I/O 寄存器时, CPU 终止。 1) 检查 HOLD 信号, WAIT 信号, clock 信号, 等等。The I/O 寄存器值可能不正确。
F0c20:	防护区域不能被访问。
F0c21:	存储器在没有准备状态。
F0c22:	存储器在没有准备状态被取消。
F0c23:	Bus 支持在连续的情况下。 1) 检查目标板或 HOLD pin 掩码的设置。
F0c24:	不能移动调试模式。 1) 检查时钟信号。这可能是由停止时钟或减慢时钟造成的。
F0c25:	Flash macro 服务 ROM 被访问或进入。 1) 请执行 [Go] 操作或 CPU 复位。
F0c26:	FLMD 终端在写-保护状态中。 1) FLMD 不在写-启动状态中。检查 FLMD0 和 FLMD1 pins 状态。
F0c27:	在一个写-保护状态中的安全标志。 1) 闪存的安全标志已经丧失了写、阻塞擦除、或芯片擦除功能。没有东西可以写入闪存。
F0c28:	内部 RAM 是不够的, 不能进行写入闪存。 1) 内部 RAM 大小小于 4 KB 以及 flash 自我-程序不能被执行。
F0c29:	闪存的空白检查失败。
F0c2a:	闪存的擦除失败。
F0c2b:	闪存写操作失败。
F0c2c:	闪存的内部确认失败。

F0c2d:	写闪存中的故障。
F0c2e:	来自 flash macro 服务没有响应。
F0c2f:	来自 flash macro 服务的响应不正确。
F0c30:	闪存 I/O 寄存器操作阻止设置需要取消。
F0c31:	在持续状态下的停止模式。不能强行中断。请释放停止模式或复位 CPU。
F0c32:	请在单芯片模式 0 中写入闪存。
F0c33:	阻止丧失芯片调试功能。
F0c34:	阻止芯片保留区域写操作。
F0c36:	不正常的内部 ROM 大小。大小不同于设备的默认大小。
F0c37:	电压太低而不能操作 flash 程序。
F0c39:	实时 RAM 监控失败。
F0c40:	有效事件条件的监控不能改变。
F0c41:	执行覆盖测试。
F0c42:	在调试模式中移动监控器失败。请复位 CPU。
F0c43:	ICE 不能通话。请确认 ICE 电源, 接口电缆连接。 1)检查电路中竞争者的电源和电缆连接。如果使用桌面电脑并且有两个或更多的 PC 卡被插入, 开关的设置可能错误。检查这个设置。或它可能已经发生故障 (当 IECUBE 已经连接)。
F0c44:	执行覆盖操作。
F0c45:	电源关闭复位竞争的内部不能执行程序。
F0c46:	在 Flash 自己竞争时这个功能是无效的。
F0c60:	在执行前事件不能被设置除了中断条件。
F0c61:	不能注册不能用于硬件中断的事件号。
F0c62:	不能使用硬件中断保留的事件号。
F0c63:	不能设置事件连接条件。
F0c64:	过多 ROM-竞争-RAM 区域。
F0c67:	在阻塞时不能进行闪存的写操作。
F0c70:	DCU 不能被访问。 1) IE 可能有故障(当 IECUBE 被连接时)。
F0c71:	不能执行复位。 1)检查时钟信号。这可能是由停止时钟或减慢时钟造成的。
F0c72:	不能访问监控器存储器。 1)在配置对话框中修正主要 OSC 值。如果这不能解决问题, IE 可能有故障 (当 the IECUBE 被连接时)。
F0c73:	不能执行监控器操作。 1) IE 可能有故障 (当 the IECUBE 被连接时)。

F0c74:	CPU 寄存器不能被访问。 1) 选择的设备文件可能不正确。在配置对话框中的芯片选择中选择支持目标芯片的文件。如果不能解决问题, IE 可能有故障 (当 the IECUBE 被连接时)。
F0c75:	在调试模式中监控器移动失败。请复位 CPU。
F0c76:	DCU 访问时初始状态启动是不寻常的。 1) 设备文件的选择可能是不正确的。在设置对话框中的芯片选择中选择支持目标芯片的设备文件。如果这不能解决问题, IE 可能有故障 (当 the IECUBE 被连接时)。
F0c77:	DCU 访问是不寻常的。 1) IE 可能有故障 (当 the IECUBE 被连接时)。
F0c78:	读跟踪数据失败。
F0ca0:	ICE 不能通话。请确认 ICE 电源, 接口电缆的连接, 或 PC 接口板的 I/O 地址。 在调试中发生错误。(IECUBE) 1) 竞争者不能通话。检查竞争者的电源, 电缆连接, 和接口板的设置等等。
F0ca1:	没建立监控器文件。 1) 必要的文件损坏。重新安装调试器。
F0ca2:	这个设备文件不包括芯片调试信息。 1) 尝试用一个不支持芯片调试的设备文件启动。这个设备文件可能旧了。重新安装最新设备文件(当 the MINICUBE 被连接时)。 2) IE 可能有故障 (当 the IECUBE 被连接时)。
F0ca3:	不支持信息包括在设备文件中的芯片调试信息中。 1) 一个不知名的标志包括在设备文件的芯片调试信息中。其执行模块可能旧了。安装最新执行模块。
F0ca4:	这个设备文件不包括 IECUBE 信息。 1) 用一个不支持 IECUBE 的设备文件尝试启动。设备文件可能旧了。重新安装最新的设备文件。
F0caf:	跟踪阻塞不能越步进行。

(2) 来自 X1000

A1000:	初始化 ICE 中的故障。
A1001:	不存在指定的登陆号。
A1002:	不能重新部署内部的 RAM。
F1003:	非法重新部署地址。
F1004:	非法条件。
A1005:	无效特征。

F1006:	非法地址。
A1007:	ICE 上没有足够的存储空间。
A1008:	工作台上没有足够的存储空间。 1) 没有足够的系统存储空间。关闭正在执行的应用软件并打开文件。
A1009:	已经初始化。
A100a:	没有初始化。
F100b:	运行用户程序。
F100c:	不同的 bus 大小已经指定。
F100d:	过大 bus 大小。
F100e:	过量 bus 部分大小。
W100f:	不转动目标。
F1010:	非法 map 范围。
F1011:	设置内部 ROM 和 RAM 中的故障。
F1012:	不支持这个特征。
F1013:	没有终端名。
W1014:	不存在数据。
A1015:	程序-IOR 不存在。
F1016:	程序-IOR 不移动。 1)必要的文件可能被损坏。 重新安装最新的设备文件。
F1017:	I/O 保护映射可能仅仅只是个目标特征。
F1018:	非法内部 ROM 大小。
A10ff:	ICE 不能通话。
A1dbe:	调试器内部发生的错误。

(3) 来自 X2000

F2000:	非法 SFR 名称。
A2001:	非法地址。
F2002:	运行用户程序。
F2003:	非法SFR号。
F2004:	非法位号。
W2005:	读保护特征的SFR被指定。
F2006:	隐藏 SFR 被指定。
F2007:	禁止读或写的SFR被指定。
F2008:	SFR 不存在被指定。
A2009:	设备文件被损坏或文件错误。
F200a:	指定 SFR 的非法值。

A200b:	不能复制。
A200c:	没有足够的存储器。 1) 没有足够的系统存储空间。关闭正在执行的应用软件并打开文件。
W200d:	SFR 的非初始化数据。
F200e:	SFR 区域不能被访问。
A20ff:	ICE 不能通话。
A2222:	非法条件。

(4) 来自 X3000

F3000:	访问非映射地址。 1)程序的分配地址和调试器的地址不匹配。 在根据编辑连接指示的文件中指定的分配地址设置映射到配置对话框中的外部存储器。当已经执行映射到外部存储器，在下载前为了访问外部存储器利用SFR 窗口或 钩程序改变必要的寄存器值。
F3001:	存储器有不同的值。
F3002:	非法启动地址。
F3003:	非法结束地址。
F3004:	非法启动地址和结束地址。
F3005:	非法条件。
F3006:	运行用户程序。
F3007:	确认错误。
F3008:	没有条件指定。
F3009:	参量大小不用通路大小排列。
F300a:	指定地址大小不用通路大小排列。
F300b:	资源地址大小不用通路大小排列。
F300c:	目的地址大小不用通路大小排列。
F300d:	非法结束地址。
F300e:	指定区域内不同的通道大小。
F300f:	资源和目的区域内不同的通道大小。
F3010:	目的区域内不同的通道大小。
F3011:	不同的通道大小, 资源 & 目的。
A3012:	没有足够的存储空间。 1)没有足够的系统存储器。关闭正在执行的应用软件并打开文件。
F3013:	写入DMM中失败。
F3014:	溢出映射区域。
F3015:	中断程序。

F3016:	不支持这个特征。
A30ff:	ICE 不能通话。

(5) 来自 X4000

F4000:	不能删除指定的事件。 1) 指定的事件不能删除因为它在其它条件下可以用。在删除前使它在其它的使用中无效。
F4001:	非法表格号。
F4002:	非法启动地址。
F4003:	非法结束地址。
F4004:	非法状态。
F4005:	非法数据。
F4006:	指定的数据号已经使用。
F4007:	太多相同的事件被注册。
F4008:	指定的事件没有被注册。
F4009:	非法数据大小。
F400a:	非法模式。
F400b:	设置值不准确。
F400c:	事件连接条件不能用于部分跟踪条件。
F400d:	太多同样的事件被注册。(>= 32767).
F400e:	指定事件条件不存在。
F400f:	非法事件连接条件。
F4010:	没建立功能。
A4011:	没有足够的存储器。 1) 没有足够的系统存储器。关闭正在执行的应用软件并打开文件。
F4012:	定时器丧失功能。
W4013:	通道大小不同于 bus 大小。
F4014:	不能使用软件中断。
F4015:	不能使用事件条件指定地址范围。
F4016:	不能改变事件条件。
F4017:	不能在奇数地址中访问单词。
A4018:	非充足存储。 1) 没有足够的系统存储器。关闭正在执行的应用软件并打开文件。
F4019:	不支持这个特征。
F401a:	没有事件。
F401b:	不能使用日志-事件。

W401c:	软件中断不能在这个区域内设置。
F401d:	定时器的启动事件和结束事件不能使用相同的设置。
F401e:	过多的跟踪-事件。
F401f:	不能设置路径数。
F4020:	执行前不能在事件中设置地址范围。
F4021:	事件条件号溢出。
F4022:	软件 DMM 条件号溢出。
F4023:	实时调用条件号溢出。
F4024:	软件中断调用条件号溢出。
F4025:	非法突发条件。
F4026:	过多事件条件不能被设置为事件连接条件的相位 1 和相位 2。
F4027:	可以设置为内部 ROM 的软件中断条件号溢出。
F4318:	非法存储器存储设置。

(6) 来自 X5000

A5000:	非法设备文件类型。
A5001:	非充足存储。 1) 没有足够的系统存储器。关闭正在执行的应用软件并打开文件。
A5002:	不能打开设备文件。
A5003:	读设备文件错误。
A5004:	不能关闭设备文件。
A5005:	非法设备文件格式。 1) 必要文件损坏。重新安装设备文件。
A5006:	ICE 初始化失败。
A5007:	设备文件被破坏或文件中有错误。
F5008:	不能打开设备文件。 1) 必要文件损坏。重新安装设备文件。
F5009:	不能打开 EX78K4.OM0。
F500a:	指定设备文件是非法译本。 1) 必要文件损坏。重新安装设备文件。
W500b:	指定设备文件不是重新部署的 IRAM。
A500c:	读 expc.ini 失败。
A500d:	非充足存储。 1) 没有足够的系统存储器。关闭正在执行的应用软件并打开文件。
W500e:	将要涉及设备文件的非日志数据。
A5300:	非法设备文件类型。

A5301:	非充足存储。 1) 没有足够的系统存储器。关闭正在执行的应用软件并打开文件。
A5302:	不能打开数据库文件。 1) 必要文件可能被损坏。重新安装调试器和设备文件。
A5303:	读数据库文件错误。
A5304:	不能关闭数据库文件。
A5305:	非法数据库文件格式。 1) 必要文件可能被损坏。重新安装调试器和设备文件。
A5306:	数据库信息已经初始化。
A5307:	数据库信息不存在。
F5308:	不能打开指定的数据库文件。 1) 必要文件可能被损坏。重新安装调试器。
F5309:	指定设备文件是非法译本。 1) 必要文件损坏。重新安装设备文件。

(7) 来自 X6000

F6000:	当前功能不存在。
F6001:	非法符号名称。
F6002:	非法条件。
F6003:	非法功能名称。
F6004:	溢出输出缓冲区大小。
F6005:	非法表达式。

(8) 来自 X7000

F7000:	非法模式
F7001:	运行用户程序。
F7002:	用户程序被停止。
F7003:	跟踪启动。
F7004:	没有设置跟踪存储器。
F7005:	功能返回地址不存在，不能执行步骤。
W7010:	不存在资源信息。
W7011:	步骤执行未知的结果。
A7012:	非充足存储。 1) 没有足够的系统存储器。关闭正在执行的应用软件并打开文件。
A70fe:	Bus hold 错误。 1) CPU 处于 the bus-hold 状态。复位调试器。

A70ff:	ICE 不能通话。
F7801:	步骤执行的结束等待状态被取消。
F7802:	步骤执行的结束等待状态被取消。
F7f00:	失败的步骤执行。
F7f02:	暂停步骤执行。
A7f03:	取消 RUN/STEP 失败。
F7f04:	不能执行非-映射区域。
F7f05:	不支持这个特征。

(9) 来自 X8000

F8000:	没建立指定的文件。
F8001:	非法行号。
F8002:	没有设置当前信息。
F8003:	非地址。
F8004:	不支持这个特征。

(10) 来自 X9000

A9000:	不存在指定的寄存器符号。
A9001:	不存在指定的寄存器符号 ID。
F9002:	非法值。
A9003:	非法条件。
A9004:	过大寄存器大小。
F9005:	不支持这个特征。

(11) 来自 Xa000

Fa001:	非法表达式。
Fa002:	启动地址比结束地址大。
Fa003:	非法资源路径。
Fa004:	太长的表达式。
Aa005:	非充足存储。 1) 没有足够的系统存储器。关闭正在执行的应用软件并打开文件。
Fa006:	非法论点。
Fa007:	非法程序号。
Fa008:	没有设置资源路径。
Fa009:	没建立文件。

Fa00a:	不能打开文件。 1) 文件损坏或不存在。重新创建这个文件。
Aa00b:	不能关闭文件。
Aa00c:	读文件失败。 1) 文件损坏或不存在。重新创建这个文件。
Fa00d:	非加载模块资源文件。
Fa00e:	非法行号。
Fa00f:	变量不存在。
Aa010:	ICE 不能通话。
Fa011:	不能访问寄存器。
Fa012:	不能访问存储器。
Aa013:	读文件错误。
Fa014:	将要打开二进制文件。
Fa015:	不能获得临时路径。 1) 磁盘已满。删除或移动不需要的文件并增加磁盘中可用的存储空间。
Fa016:	不能创建临时文件。 1) 磁盘已满。删除或移动不需要的文件并增加磁盘中可用的存储空间。
Fa017:	不能移动临时文件。
Fa020:	不支持这个特征。
Fa021:	符号分配到寄存器不能被指定。
Fa022:	不能用于被包含的文件夹或文件夹不存在的字符。

(12) 来自 Xb000

Fb000:	非法命令行。
Fb001:	不存在指定模块文件中的程序信息。
Fb002:	没建立文件。
Fb003:	没建立功能。
Fb004:	选择加载模块不同加载的种类。(芯片)
Fb005:	没建立符号 1) 没建立地址。指定一个位置支持地址信息。
Fb008:	非法表达式。
Ab009:	非充足存储。 1) 没有足够的系统存储器。关闭正在执行的应用软件并打开文件。
Fb00a:	加载模块文件中的非法符号。
Fb00b:	当前程序不存在。
Fb00c:	当前文件不存在。

Ab00d:	当前功能不存在。
Ab00e:	当前行不存在。
Ab00f:	没建立日志。
Ab010:	加载符号表失败。
Ab011:	非法行号。
Ab015:	读文件出错。 1) 文件损坏或不存在。重新创建文件。
Ab016:	不能打开文件。 1) The 文件损坏或不存在。重新创建文件。
Ab017:	写文件失败。 1) 文件损坏或不存在。重新创建文件。
Ab019:	读文件出错。
Ab01a:	不能关闭文件。
Fb01b:	太长的加载文件名。
Ab01c:	太多任务种类。
Fb01d:	地址没建立
Wb01e:	没有调试信息 (调试 Build 模式中不编译)。
Fb01f:	找不到结构成员。
Fb020:	找不到值。
Fb021:	在加载模块文件中不存在调试信息。 1) 用附加调试信息创建一个加载模块, 在调试 Build 中的构造模式中执行构造。
Fb022:	非法行号。
Ab023:	当前堆栈结构不起作用。
Ab024:	不同部分。
Fb026:	太多数组元 (> 4)。
Fb027:	建立末尾文件。 1) 指定的文件可能损坏。重新创建文件。
Fb028:	不支持这个特征。
Fb029:	非法地址。
Ab02a:	ICE 不能通话。
Fb02b:	不能用当前 PC 值跟踪堆栈。
Fb02c:	一个功能的太多阻塞。
Fb02d:	非法论点。

Fb02e:	在 SOURCE PATH 中文件不存在。 1) 停止程序执行时,不建立显示调试资源。检查路径是否连接 调试选项对话框 中的资源,或检查资源和外面的目录是否在同一个目录中。涉及显示错误消息的 汇编窗口 ,检查相应的路径是否连接。
Fb02f:	由于最优化信息已经删除。
Ab030:	监控器中断。 1) 检查电路竞争中的电源,电缆连接,和接口板设置并重启调试器。
Ab031:	在存储器中已经设置。
Ab032:	超出范围。
Ab033:	LP 不被存储。
Fb034:	从现在的 PC 位置返回执行不能被实现。
Fb037:	太多行-号信息。
Fb038:	编译器译本失谐。 1) 用最新编译器重新创建加载模块。
Ab039:	加载调试器信息失败。
Ab03a:	没有更多的部分信息。
Fb040:	指定得到文件不是加载模块。 1) 这不是连接器输出文件。在从连接器输出前资源调试不能用加载模块执行。指定加载模块从连接器输出。
Ab041:	下载太多的加载模块中的文件。
Wb042:	符号模块没有初始化。
Fb32e:	非法端口号。
Fb32f:	非法端口名。
Fb330:	非法端口位置。
Fb331:	非法增量号。
Fb332:	没有设置存储器存储端口。
Fb333:	非法存储号。
Fb334:	没有设置存储器存储区域。
Wb335:	太长的符号名。

(13) 来自 Xc000

Fc001:	不能打开文件。 1) 这个文件损坏或不存在。重新创建这个文件。
Ac002:	不能关闭文件。

Ac003:	读文件出错。 1) 这个文件损坏或不存在。重新创建这个文件。
Ac004:	读文件出错。
Fc005:	非法文件类型。
Fc006:	加载模块的种类(芯片)是非法的。
Fc007:	指定的文件不是加载模块。 1) 这不是连接器输出文件。在从连接器输出前资源调试不能用加载模块执行。指定加载模块从连接器输出。
Fc008:	指定加载模块文件 (COFF) 是旧译本。
Ac009:	非充足存储。 1) 没有足够的系统存储空间。关闭正在执行的应用软件并打开文件。
Fc00a:	不访问映射地址。
Fc00b:	没有加载加载模块。
Fc00c:	非法论点。
Fc00d:	运行用户程序。
Fc00e:	跟踪用户程序。
Fc00f:	中断。
Ac010:	ICE 不能通话。
Fc011:	非法加载文件格式。
Fc012:	检查求和错误。
Fc013:	上传的地址范围太宽。 (> 1M byte).
Fc014:	写文件失败。 1) 文件被损坏或不存在。重新创建文件。
Fc015:	非法程序号。
Fc016:	加载信息已满。
Wc017:	复制符号信息, 请复位符号。
Fc018:	指定的文件不是加载模块。 1) 这不是连接器输出文件。在从连接器输出前资源调试不能用加载模块执行。指定加载模块从连接器输出。
Fc019:	写存储器失败。
Wc01a:	BSS 区域被分配到非-映射区域。 1) 当程序执行时, 一个 非-映射中断可能发生。利用一个连接指示分配BSS区域到内部RAM, 或利用调试器的配置对话框映射竞争存储器或目标存储器到BSS 区域。
Fc01b:	可编程的-IOR 地址没有指定。 1) 需要的文件可能损坏。重新安装调试器。

Wc01c:	可编程的 IOR 地址不匹配。 1) 必要文件可能损坏。重新创建文件。
Wc01d:	选择加载不同种类(芯片)的加载模块。
Fc01e:	不支持 Flash 擦除。
Fc100:	不支持这个特征。

(14) 来自 Xd000

Ad000:	调试器内部发生错误。
Ad001:	非充足存储。 1) 没有足够的系统存储空间。关闭正在执行的应用软件并打开文件。
Ad002:	读初始化文件失败 (expc.ini).
Ad003:	ICE 没有被连接。
Fd004:	找不到动态连接库。

(15) 来自 Xe000

Fe000:	非法论点。
Fe001:	非法启动地址。
Fe002:	非法结束地址。
Fe003:	大小太大。
Fe004:	不能打开文件。 1) 文件损坏或不存在。重新创建文件。
Fe005:	读文件失败。 1) 文件损坏或不存在。重新创建文件。
Fe006:	读文件出错。
Fe007:	写文件出错。 1) 文件损坏或不存在。重新创建文件。
Ae008:	非充足存储。 1) 没有足够的系统存储空间。关闭正在执行的应用软件并打开文件。
Fe009:	非法文件格式。
Fe00a:	确认错误。
Fe010:	不支持这个特征。

(16) 来自 Xf000

Af000:	非充足存储。 1) 没有足够的系统存储空间。关闭正在执行的应用软件并打开文件。
Ff000:	非充足存储。

Ff001:	[XXX] 没有建立。
Wf002:	没有建立 [XXX].从启动处搜索?
Wf003:	已经超过搜索范围。
Ff004:	缺少的参数。
Ff005:	非法功能名称。
Ff006:	非法数字。
Ff007:	启动地址比末尾地址大。
Ff008:	非法符号或表达式。
Ff009:	[XXX] 这个文件是非法类型。
Ff00a:	运行用户程序。
Ff100:	磁盘不能写入或已满。
Ff101:	文件没有建立。
Ff102:	文件没有创建。
Ff103:	旧项目文件译本。
Ff104:	非法项目文件格式。
Ff105:	这个文件是[XXX]的一个项目文件。请选择一个正确的文件。
Wf106:	改变项目文件中的 CPU 。你必须推出新 CPU 的调试器。你退出调试器了吗?
Wf107:	改变项目文件中的 CPU 。你用这个 CPU 启动调试器吗?
Wf108:	从打开的芯片[XXX]中选择不同的[YYY]项目文件。虽然芯片不可能改变, 打开它了吗?
Wf109:	项目管理器不可能和调试器的译本一样使用。请使用 PM 插件。
Wf200:	没有遇到差异。
Ff201:	存储器映射错误。
Ff202:	校验错误。 不能访问外存储器, 因为它没有设置。为了访问外存储器, 下载前利用 SFR 窗口 和 钩程序 改变存储器值。
Wf203:	当运行一个程序时, 当重写一个存储器时, 程序执行停止一会儿。你想重新写入吗?
Wf300:	你愿意用[XXX]保存改变吗?
Ff301:	在事件条件上使用的符号不能被求值。
Wf302:	删除: [XXX]
Wf303:	编辑[XXX] 。删除: [YYY]?
Wf304:	编辑[XXX] 保存: [YYY]?
Wf305:	[XXX] 已经存在。你要代替它吗?
Ff306:	名字太长。
Ff307:	在其它种类中有相同的名字。

Ff308:	地址不能省略。
Ff309:	非法地址掩码。
Ff30a:	非法数据掩码。
Ff30b:	非法 ext 探针掩码。
Ff30c:	非法 ext 探针数据。
Ff30d:	非法通道数。
Ff30e:	非法寄存器名。
Ff30f:	非法寄存器存储。
Ff310:	非法延迟数。
Wf311:	只有一个[XXX] 能被启动。你可以启动[YYY]吗?
Ff312:	[XXX] 已经在那儿。
Ff313:	事件数目已经存在。
Ff314:	事件名称没有设置。
Ff315:	[XXX] 已经在那儿。
Ff316:	最大数量地启动 [XXX]事件结束。 请不要启动其它 [YYY] 事件。
Ff317:	最大数量的设置事件结束。
Ff31e:	非法启动地址。
Ff31f:	非法末尾地址。
Ff322:	非法计数率。
Ff323:	非法中断计数。
Ff324:	部分和限制可以同时指定。
Wf325:	运行用户程序。你想停止用户程序一会儿吗并设置它?
Wf326:	运行用户程序。你想停止用户程序一会儿吗并删除它?
Ff350:	事件不在中间有一个相位。
Ff351:	相同的事件包含在连接和丧失功能中。
Ff352:	不指定事件。
Ff357:	在相位中的 AND 事件。
Ff400:	覆盖映射错误。
Wf401:	清除覆盖?
Ff500:	非法符号。
Ff501:	非法值。
Ff502:	非法参量。
Ff503:	最大数量的符号结束。

Ff504:	这个变量不能设置为一个中断。 1)不能给下列的变量设置中断。 - 局部变量, 静态变量 - 数组变量, 结构体/共同体的成员变量 - 寄存器/SFR - 变量表达式
Wf600:	保存项目文件?
Wf601:	当连接目标系统时, 请转到目标系统。 1) 当没有连接目标时, 简单地点击 <OK> 按钮。
Wf602:	请改变一个 MODE 掩码条件或连接目标系统。
Ff603:	不正确的 ID 代码。 1)这可能由下列因素造成(当 MINICUBE 被连接时)。 - The ID 代码不正确。 ->输入正确的 ID 代码。 - 内部闪存处于写状态因为 FLMD0 pin 处于高位。 -> 使 FLMD0 pin 处于地位。 -设置竞争者连接阻止模式因为 ID 代码(0x84 地址 0,1 位)是 0。 ->用 flash 复写器立刻擦除内部闪存。
Af604:	不正确的 ID 代码。异常中断调试器。 1)可能由下列因素造成 (当 the MINICUBE 已被连接)。 - ID 代码不正确。 -> 输入正确 ID 代码。 - 内部闪存处于写状态因为 FLMD0 pin 处于高位。 -> 使 FLMD0 pin 处于地位。 -设置竞争者连接阻止模式因为 ID 代码(0x84 地址 0,1 位)是 0。 ->用 flash 复写器立刻擦除内部闪存。
Ff605:	请检查目标板的连接。 1)检查目标连接器的连接 (TC)。如果一个目标没有被连接,在配置对话框中回顾目标设置。
Ff606:	请检查目标板的连接, 并将它接通电源。 1) 检查目标电源供应。如果目标没有被连接, 在配置对话框中回顾目标设置。
Wf607:	请检查交换适配器的连接。 1) 检查交换适配器的连接 (EA)。 如果目标没有连接, 推荐交换适配器的安装。

Ff608:	请断开目标板。 电流可能从内部电源供应流到目标。从交换适配器(EA)断开目标连接器(TC)。如果目标没有连接在 设置对话框 中回顾设置。
Ff609:	请关闭目标板电源，并断开它。
Af60a:	不正确的 ID 代码。闪存被擦除。异常中断调试器。 1)当模式擦除闪存内容已经设置时如果 ID 鉴定地址 0x84 导致失败，显示这个消息。并且闪存内容被删除(当 MINICUBE 被连接时)。
Af60b:	丧失 ID 代码。擦除闪存。异常中断调试器。 1)当利用竞争模式关闭电源中的调试器时，当目标电源关闭后调试器激活时或当闪存内容被擦除时(当 MINICUBE 连接时)，如果目标连接电缆断开则显示消息。
Af60c:	在中断时目标没有接通。
Wf700:	你想下载加载模块文件吗？
Wf701:	你想仅仅加载符号信息吗？
Wf800:	存储器存储配置没有设置。
Wf801:	BANK 地址必须在目标存储器中。
Ff802:	所有的事件被删除。因为外部探针用法已经改变。
Ff803:	在当前配置上事件地址是有效的。
Ff804:	有效 PC 值。
Ff805:	在这个地址上不能设置临时中断。
Ff806:	外部数据通过调试器使用。
Ff900:	非法 I/O 端口名称。
Ff901:	存储器映射错误。 1) 地址的规格是非法的。在 添加 I/O 端口对话框 中检查可能被指定的地址。
Ff902:	非法通道大小。
Ff903:	非法通道类型。
Ff904:	有相同的名字。
Wf905:	[XXX] 已经存在。你想代替它吗？
Wf906:	你愿意记录在[XXX]中所作的改变吗？
Ffa00:	PC 位置上当前程序的[XXX]功能没有建立。 1) 在 main() 标签中指定的符号:在 调试选项对话框 中可以找到。设置程序的主程序符号。默认值是 _main。

Ffa01:	PC 位置上的行信息没有建立。 1)当程序停止时资源文件相应的程序计数器(PC)值找不到。可能是下列原因造成的。 -资源路径没有连接的位置存在资源文件。 -程序停止在资源文件处,像库或 RX,或不存在的位置。 -程序循环,跳转到程序不能使用的地址,以及停在那儿。
Wfb00:	运行用户程序。你想要停止用户程序吗? 1) <是> 按钮被选择,停止执行用户程序并且接着显示退出调试对话框。如果它在调试选项对话框中被详细说明则不显示退出调试对话框,然而, ID78K0-QB 被终止。 <否> 按钮被选择,不停止执行用户程序并且不显示退出调试对话框。ID78K0-QB 不被终止。
Wfb01:	由于在 ID 代码中地址 0x79 的 7 位是 0, N-电线竞争者今后禁止使用。你像它一样退出调试器了吗?
Ffc00:	在线帮助窗口不能被启动。请安装带用户参考手册的 HTML 帮助环境。
Ffd00:	指定[XXX]失败。
Ffe00:	超过 RRM 的最大大小。
Wfe01:	有一个副本 RRM 地址。
Wfe0b:	移到 flash 模式。它完全清除了除了当前事件。注意到了吗?
Fffff:	中断。

附录E 索引

A

关于对话框 ... 266
访问监控器 ... 182
访问监控器功能 ... 60
活跃状况和静止状况 ... 87
添加 I/O 端口对话框 ... 205
添加监视监视对话框 ... 171
地址移动对话框 ... 161
汇编搜寻对话框 ... 159
汇编窗口 ... 154

B

中断

 中断点设置 ... 50
 设置中断变量 ... 51
中断对话框 ... 247
中断功能... 49
浏览对话框 ... 269

C

C0 覆盖 ... 69
调回程序 ... 277
CC78K0 ... 24
改变监视监视对话框... 174
字符设置... 321
时钟 ... 115
代码覆盖 ... 69
代码覆盖窗口 ... 228
来到这儿 ... 54
命令... 268
命令参考 ... 271
条件跟踪 ... 66
配置对话框 ... 110
控制台窗口... 268
内容保存到项目文件 ... 83
...上下文菜单 96

D

调试功能目录 ... 38
延迟计数对话框 ... 227
延迟触发 ... 226
延迟触发跟踪 ... 68
DMM 对话框 ... 193
下载 ... 41
下载对话框 ... 141
下载功能 / 上传功能 ... 41
拖动 & 下降功能 ... 90

E

环境设置文件加载对话框 ... 263
环境设置文件保存对话框 ... 262

启动处的错误消息 ... 32
错误 ... 330
事件 DMM 条件 ... 82
事件 DMM 对话框 ... 255
事件功能... 72
事件图标... 76
事件管理... 76
事件设置状态(事件标志) ... 146
退出调试器窗口 ... 265
表达式 ... 323
扩展选项对话框 ... 119

F

自动防故障装置的中断 ... 116
自动防故障装置的中断对话框 ... 122

G

-g 选项... 25

H

钩程序 ... 278

I

I/O 保护 ... 40
ID 代码 ... 25, 114
IECUBE ... 22
输入习惯 ... 321
安装 ... 26
内部扩展 RAM ... 40
内部高速 RAM ... 40
内部 ROM ... 40
IOR 窗口 ... 199

J

跳转功能 ... 88

L

加载/保存功能 ... 83
局部变量窗口 ... 176
执行覆盖度量的位置... 70

M

主窗口 ... 97
映射 ... 117
映射属性... 40
映射设置... 40
屏蔽 ... 117
存储器比较对话框 ... 190
存储器比较结构对话框 ... 192
存储器复制对话框... 189
存储器填充对话框... 187

存储器处理功能... 59
 存储器搜索对话框... 185
 存储器对话框窗口... 181
 消息 ... 330
 MINICUBE ... 23, 120
 MINICUBE2 ... 23
 混合显示模式源窗口 ... 47
 跟踪视图窗口 ... 66

O OCD Checker ... 27
 操作环境 ... 24
 s 操作数 ... 324

P
 挑选 ... 224
 PM+ ... 24, 33
 点标志区域 ... 146, 155
 程序代码 ... 147
 程序执行功能 ... 53
 项目文件 ... 83, 265
 项目文件加载对话框 ... 139
 项目文件保存对话框 ... 138
 伪仿真器对话框 ... 137

Q
 限制跟踪 ... 68, 226
 快速监视监视对话框 ... 169

R
 RA78K0 ... 24
 基数范围 ... 325
 寄存器窗口 ... 196
 复位... 264
 Reset 复位调试对话框 ... 264
 RRM 对话框 ... 124
 运行-中断事件 ... 64

S
 部分跟踪... 68, 226
 安全 ID ... 25
 设置调试环境 ... 39
 设置文件 ... 86
 设置映射 ... 40
 SFR 选择对话框 ... 203
 SFR 窗口 ... 199
 截图对话框 ... 249
 软件中断管理器 ... 231
 资源搜索对话框 ... 150
 资源文本移动对话框 ... 152
 源窗口 ... 145
 堆栈 ... 40
 堆栈跟踪延迟功能 ... 58
 堆栈窗口 ... 178
 从这里启动 ... 54
 启动选项... 27

启动日常事务... 135
 状态条 ... 107
 存根对话框 ... 253
 Sym 检查窗口 ... 320
 符号转地址对话框 ... 162

T
 目标... 40

Tcl
 汇编... 282
 批 ... 283
 断点... 284
 dbgexit ... 286
 下载 ... 287
 擦除 ... 288
 extwin ... 289
 完成 ... 290
 go ... 291
 帮助 ... 292
 钩 ... 293
 检查 ... 294
 跳转 ... 295
 映射 ... 296
 多文档界面... 297
 存储器 ... 298
 模块 ... 299
 下一步 ... 300
 刷新 ... 301
 寄存器 ... 302
 复位 ... 303
 运行 ... 304
 步骤 ... 305
 停止 ... 306
 tkcon ... 315
 上传 ... 307
 译本 ... 308
 监视 ... 309
 位置 ... 310
 wish ... 311
 x 覆盖... 312
 x 计时... 313
 x 跟踪.. 314

Tcl 命令目录 ... 272
 暂停中断 ... 209
 计时器功能 ... 63
 计时器结果对话框 ... 210
 跟踪数据选择对话框 ... 223
 跟踪对话框 ... 225
 跟踪功能 ... 65
 跟踪存储器 ... 65
 跟踪移动对话框 ... 221
 跟踪搜索对话框 ... 216
 跟踪器控制模式 ... 67
 消息类型... 330

U

无条件跟踪 ... 66
卸载... 26
上传 ... 44
上传对话框 ... 143

V

校验检查 ... 121
译本-向上服务 ... 24
视图文件 ... 85
视图文件加载对话框... 260
视图文件保存对话框 ... 258

W

监视功能 ... 55
监视窗口 ... 164
窗口目录 ... 94
窗口参考 ...93

详细信息请联系:

(中国区)

网址:

<http://www.cn.necel.com/>

<http://www.necel.com/>

[北京]

日电电子(中国)有限公司
中国北京市海淀区知春路 27 号
量子芯座 7, 8, 9, 15 层
电话: (+86)10-8235-1155
传真: (+86)10-8235-7679

[深圳]

日电电子(中国)有限公司深圳分公司
深圳市福田区益田路卓越时代广场大厦 39
楼 3901, 3902, 3909 室
电话: (+86)755-8282-9800
传真: (+86)755-8282-9899

[上海]

日电电子(中国)有限公司上海分公司
中国上海市浦东新区银城中路 200 号
中银大厦 2409-2412 和 2509-2510 室
电话: (+86)21-5888-5400
传真: (+86)21-5888-5230

[香港]

香港日电电子有限公司
香港九龙旺角太子道西 193 号新世纪广场
第 2 座 16 楼 1601-1613 室
电话: (+852)2886-9318
传真: (+852)2886-9022
2886-9044

上海恩益禧电子国际贸易有限公司
中国上海市浦东新区银城中路 200 号
中银大厦 2511-2512 室
电话: (+86)21-5888-5400
传真: (+86)21-5888-5230