

お客様各位

カタログ等資料中の旧社名の扱いについて

2010年4月1日を以ってNECエレクトロニクス株式会社及び株式会社ルネサステクノロジが合併し、両社の全ての事業が当社に承継されております。従いまして、本資料中には旧社名での表記が残っておりますが、当社の資料として有効ですので、ご理解の程宜しくお願ひ申し上げます。

ルネサスエレクトロニクス ホームページ (<http://www.renesas.com>)

2010年4月1日

ルネサスエレクトロニクス株式会社

【発行】ルネサスエレクトロニクス株式会社 (<http://www.renesas.com>)

【問い合わせ先】 <http://japan.renesas.com/inquiry>

ご注意書き

1. 本資料に記載されている内容は本資料発行時点のものであり、予告なく変更することがあります。当社製品のご購入およびご使用にあたりましては、事前に当社営業窓口で最新の情報をご確認いただきますとともに、当社ホームページなどを通じて公開される情報に常にご注意ください。
2. 本資料に記載された当社製品および技術情報の使用に関連し発生した第三者の特許権、著作権その他の知的財産権の侵害等に関し、当社は、一切その責任を負いません。当社は、本資料に基づき当社または第三者の特許権、著作権その他の知的財産権を何ら許諾するものではありません。
3. 当社製品を改造、改変、複製等しないでください。
4. 本資料に記載された回路、ソフトウェアおよびこれらに関連する情報は、半導体製品の動作例、応用例を説明するものです。お客様の機器の設計において、回路、ソフトウェアおよびこれらに関連する情報を使用する場合には、お客様の責任において行ってください。これらの使用に起因しお客様または第三者に生じた損害に関し、当社は、一切その責任を負いません。
5. 輸出に際しては、「外国為替及び外国貿易法」その他輸出関連法令を遵守し、かかる法令の定めるところにより必要な手続を行ってください。本資料に記載されている当社製品および技術を大量破壊兵器の開発等の目的、軍事利用の目的その他軍事用途の目的で使用しないでください。また、当社製品および技術を国内外の法令および規則により製造・使用・販売を禁止されている機器に使用することができません。
6. 本資料に記載されている情報は、正確を期すため慎重に作成したのですが、誤りが無いことを保証するものではありません。万一、本資料に記載されている情報の誤りに起因する損害がお客様に生じた場合においても、当社は、一切その責任を負いません。
7. 当社は、当社製品の品質水準を「標準水準」、「高品質水準」および「特定水準」に分類しております。また、各品質水準は、以下に示す用途に製品が使われることを意図しておりますので、当社製品の品質水準をご確認ください。お客様は、当社の文書による事前の承諾を得ることなく、「特定水準」に分類された用途に当社製品を使用することができません。また、お客様は、当社の文書による事前の承諾を得ることなく、意図されていない用途に当社製品を使用することができません。当社の文書による事前の承諾を得ることなく、「特定水準」に分類された用途または意図されていない用途に当社製品を使用したことによりお客様または第三者に生じた損害等に関し、当社は、一切その責任を負いません。なお、当社製品のデータ・シート、データ・ブック等の資料で特に品質水準の表示がない場合は、標準水準製品であることを表します。
標準水準： コンピュータ、OA 機器、通信機器、計測機器、AV 機器、家電、工作機械、パーソナル機器、産業用ロボット
高品質水準： 輸送機器（自動車、電車、船舶等）、交通用信号機器、防災・防犯装置、各種安全装置、生命維持を目的として設計されていない医療機器（厚生労働省定義の管理医療機器に相当）
特定水準： 航空機器、航空宇宙機器、海底中継機器、原子力制御システム、生命維持のための医療機器（生命維持装置、人体に埋め込み使用するもの、治療行為（患部切り出し等）を行うもの、その他直接人命に影響を与えるもの）（厚生労働省定義の高度管理医療機器に相当）またはシステム等
8. 本資料に記載された当社製品のご使用につき、特に、最大定格、動作電源電圧範囲、放熱特性、実装条件その他諸条件につきましては、当社保証範囲内でご使用ください。当社保証範囲を超えて当社製品をご使用された場合の故障および事故につきましては、当社は、一切その責任を負いません。
9. 当社は、当社製品の品質および信頼性の向上に努めておりますが、半導体製品はある確率で故障が発生したり、使用条件によっては誤動作したりする場合があります。また、当社製品は耐放射線設計については行っておりません。当社製品の故障または誤動作が生じた場合も、人身事故、火災事故、社会的損害などを生じさせないようお客様の責任において冗長設計、延焼対策設計、誤動作防止設計等の安全設計およびエージング処理等、機器またはシステムとしての出荷保証をお願いいたします。特に、マイコンソフトウェアは、単独での検証は困難なため、お客様が製造された最終の機器・システムとしての安全検証をお願いいたします。
10. 当社製品の環境適合性等、詳細につきましては製品個別に必ず当社営業窓口までお問合せください。ご使用に際しては、特定の物質の含有・使用を規制する RoHS 指令等、適用される環境関連法令を十分調査のうえ、かかる法令に適合するようご使用ください。お客様がかかる法令を遵守しないことにより生じた損害に関し、当社は、一切その責任を負いません。
11. 本資料の全部または一部を当社の文書による事前の承諾を得ることなく転載または複製することを固くお断りいたします。
12. 本資料に関する詳細についてのお問い合わせその他お気付きの点等がございましたら当社営業窓口までご照会ください。

注 1. 本資料において使用されている「当社」とは、ルネサスエレクトロニクス株式会社およびルネサスエレクトロニクス株式会社とその総株主の議決権の過半数を直接または間接に保有する会社をいいます。

注 2. 本資料において使用されている「当社製品」とは、注 1 において定義された当社の開発、製造製品をいいます。



お客様各位

資料中の「三菱電機」、「三菱XX」等名称の株式会社ルネサス テクノロジへの変更について

2003年4月1日を以って株式会社日立製作所及び三菱電機株式会社のマイコン、ロジック、アナログ、ディスクリート半導体、及びDRAMを除くメモリ(フラッシュメモリ・SRAM等)を含む半導体事業は株式会社ルネサス テクノロジに承継されました。

従いまして、本資料中には「三菱電機」、「三菱電機株式会社」、「三菱半導体」、「三菱XX」といった表記が残っておりますが、これらの表記は全て「株式会社ルネサス テクノロジ」に変更されておりますのでご理解の程お願い致します。尚、会社商標・ロゴ・コーポレートステートメント以外の内容については一切変更しておりませんので資料としての内容更新ではありません。

注:「高周波・光素子事業、パワーデバイス事業については三菱電機にて引き続き事業運営を行います。」

2003年4月1日
株式会社ルネサス テクノロジ
カスタマサポート部

カスタムウィンドウサンプルプログラムのご紹介

1 カスタムウィンドウサンプルプログラムの使用方法

カスタムウィンドウサンプルプログラムの使用方法は、「PD79 ユーザーズマニュアル」の「カスタマイズ機能」を参照下さい。

2 プログラムの概要

2.1 アニメート実行ウィンドウ(animate_win)

アニメート実行するウィンドウです。アニメート実行とは、ステップ実行および PC 表示更新を繰り返す実行です。PD79 のプログラムウィンドウの PC 表示マークの移動を確認しながら、適当なところで停止させることができます。

- <Step>ボタンを入力することにより、ソース行ステップのアニメート実行を繰り返します。
- <StepI>ボタンを入力することにより、1 命令ステップのアニメート実行を繰り返します。
- <Stop>ボタンを入力することにより、アニメート実行を停止します。
- アニメート実行中は、PD79 ウィンドウのツールバーの<Stop>ボタンおよび<STOP>スクリプトコマンドは機能しません。

(ウィンドウ表示例)

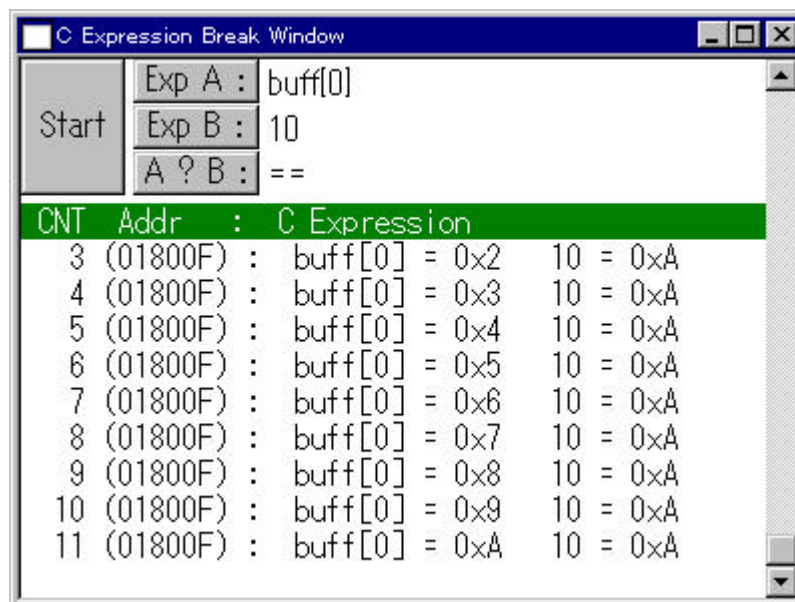


2.2 C 言語式ブレークウィンドウ(bpcheck_win)

ターゲットプログラム停止時に C 言語式で設定した比較条件が成立しているかを判定して、成立していなければ再度実行を繰り返すウィンドウです。ある条件が成立するまで実行を繰り返す場合に有効です。

- 本ウィンドウを使用するには、あらかじめターゲットにプログラムをダウンロードしておき、任意の位置にブレークポイントを設定しておいてください。
- <Exp A:>ボタンを入力し、オープンする入力ダイアログに C 言語式を入力することにより条件式の左辺を設定します。
- <Exp B:>ボタンを入力し、オープンする入力ダイアログに C 言語式を入力することにより条件式の右辺を設定します。
- <A ? B:>ボタンを入力し Exp A と Exp B の比較条件 (==, !=等の比較演算子) を選択します。
- 上記の設定の後、<Start>ボタンを入力することによりターゲットを実行させます。その後、ブレークポイント通過などによるターゲット停止時に、設定した比較条件を評価します。条件が成立していなければ、再度ターゲットを実行させ、比較条件が成立するまで実行を繰り返します。
- <Start>ボタンのラベルは、ターゲットの実行時に<Stop>に変更されます。実行を中断する際には<Stop>ボタンを入力します。
- 実行中は、ウィンドウにターゲットが停止したときの回数、PC 値、Exp A と Exp B の値をリスト形式で表示します。

(ウィンドウ表示例)

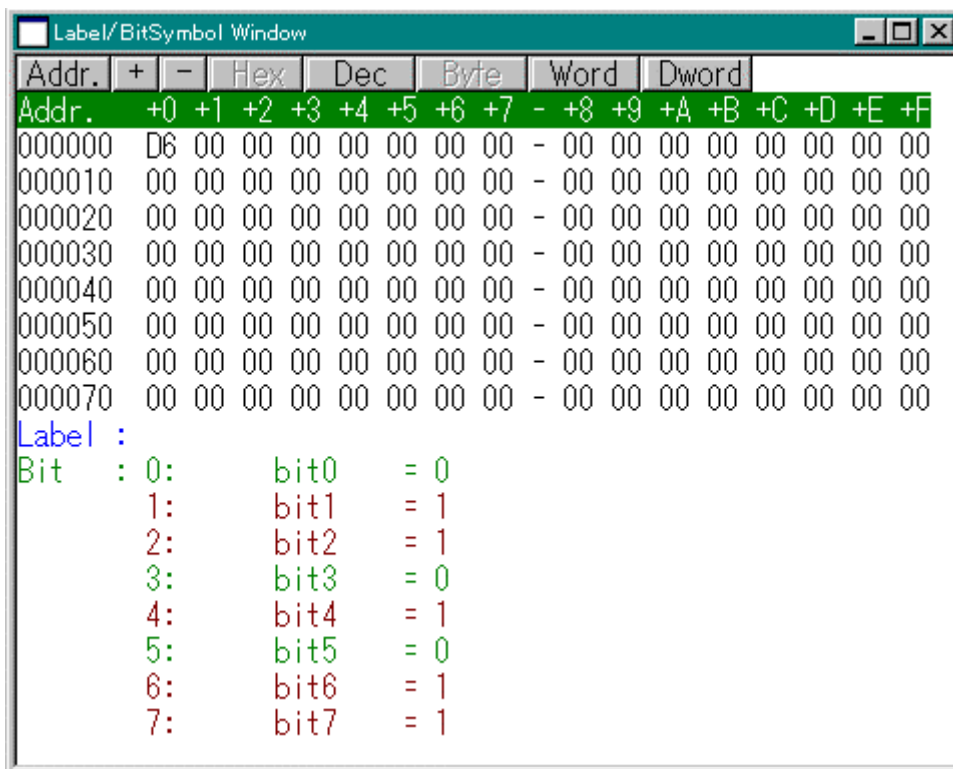


2.3 ラベル/ビットシンボル表示付きダンプウィンドウ(label_win)

ダンプ形式で表示されたデータのラベル/ビットシンボルを表示するウィンドウです。

- 表示されているデータにマウスのカーソルを移動すると、データのアドレスにラベル/ビットシンボルが割り当てられている場合には、ラベル名/ビットシンボル情報(ビットシンボル名、ビット番号、ビット値)が表示されます。
- 表示されているデータをダブルクリックし、オープンする入力ダイアログに値を入力することにより、データを変更することができます。
- <Addr>ボタンを入力し、オープンする入力ダイアログに値を入力することにより、表示開始アドレスを変更することができます。
- <+>/<->ボタンを入力することにより、128 バイト単位で表示開始アドレスを増減できます。
- <Hex>/<Dec>ボタンを入力することにより、表示基数を 16 進数/10 進数に切り替えることができます。
- <Byte>/<Word>/<Dword>ボタンを入力することにより、表示バイト数を 1 バイト、2 バイト、4 バイトに切り替えることができます。

(ウィンドウ表示例)



2.4 18 セグメントLED ウィンドウ (led18r_win、led18g_win、led18b_win)

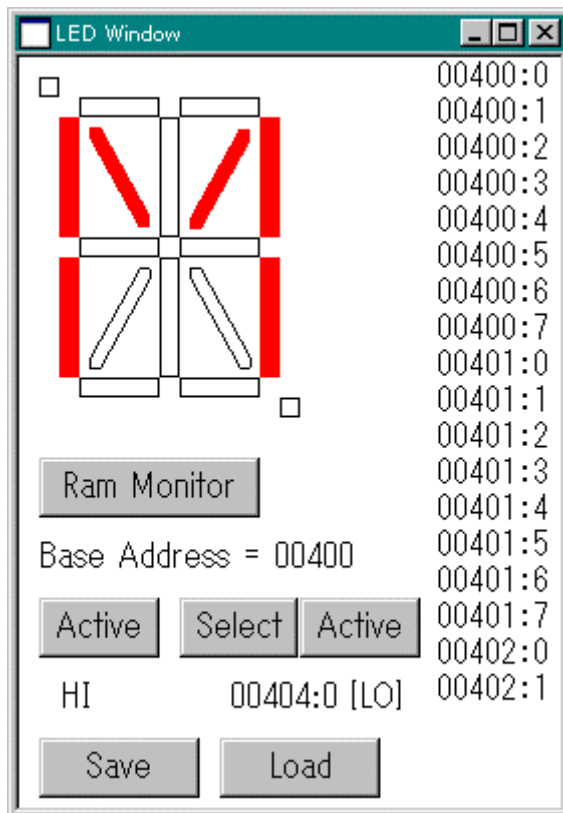
18 個のセグメント LED 表示を行うウィンドウです。メモリ値に対応して LED を点灯するような場合に、ターゲットシステムなしに LED 表示を確認できます。led18r_win は赤色に LED 点灯し、led18g_win は緑色に LED 点灯し、led18b_win は青色に LED 点灯します。

- LED セグメントをダブルクリックし、オープンする入力ダイアログにアドレス値を入力し、次にオープンする入力ダイアログにビット番号を入力することにより、LED セグメントにアドレスとビット番号を設定することができます。設定した LED セグメント情報はウィンドウの右側に表示されます。
- アドレス値としてビットシンボルが指定できます。ビットシンボルを指定した場合には、ビット番号を入力するダイアログはオープンしません。
- LED セグメントをクリックすると、対応する LED セグメント情報が点灯色で表示されます。
- LED セグメントに設定したアドレスが、リアルタイム RAM モニタ領域(1KB)内にある場合には、サンプリングによってターゲットの実行を妨げることなく LED セグメントが最新情報に更新されます。
- LED セグメントに設定したアドレスが、リアルタイム RAM モニタ領域(1KB)内にはない場合には、サンプリングの際にターゲットの実行を妨げます。
- <RAM Monitor>ボタンを入力し、オープンする入力ダイアログにアドレス値を入力することにより、リアルタイム RAM モニタ領域の開始アドレスを変更することができます。
- <Active>ボタンを入力することにより、各ビットが High アクティブか Low アクティブかを選択します。
- <Select>ボタンを入力し、オープンする入力ダイアログにアドレス値を入力し、次にオープンする入力ダイアログにビット番号を入力することにより、全 LED セグメントの点灯/消灯を選択する選択ビットを設定します。この選択ビットは、LED が複数桁存在する場合の、桁選択ビットのことです。選択ビットがアクティブな場合にのみ LED セグメント情報に従って LED セグメントを点灯します。
- <Select>ボタンにつながっている<Active>ボタンを入力することにより、選択ビットが High アクティブか Low アクティブかを選択します。たとえば、High アクティブの場合は、選択ビットが High の場合にのみ設定された LED セグメント情報に従って LED セグメントを点灯します。
- <Save>/<Load>ボタンを入力することにより、オープンするファイルダイアログでファイルを指定することにより、設定した LED セグメント情報をファイルにセーブ/ファイルからロードできます。

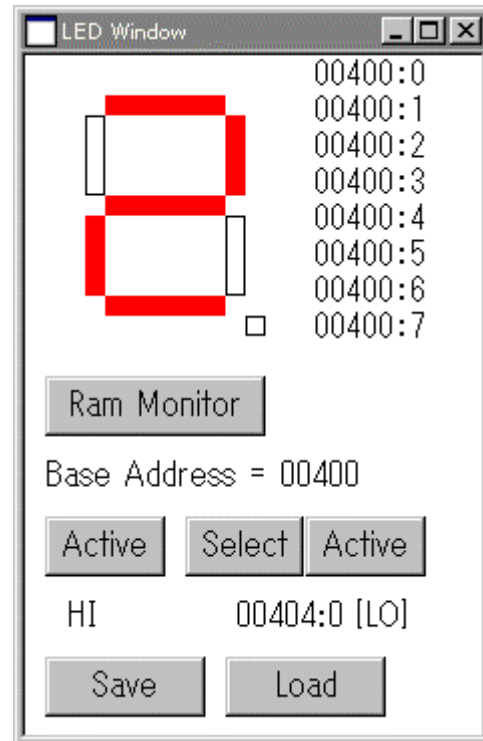
2.5 8 セグメントLED ウィンドウ (led8r_win、led8g_win、led8b_win)

8 個のセグメント LED 表示を行うウィンドウです。メモリ値に対応して LED を点灯するような場合に、ターゲットシステムなしに LED 表示を確認できます。led8r_win は赤色に LED 点灯し、led8g_win は緑色に LED 点灯し、led8b_win は青色に LED 点灯します。使用方法は、18 セグメント LED ウィンドウと同じです。

(18 セグメント LED ウィンドウ表示例)



(8 セグメント LED ウィンドウ表示例)

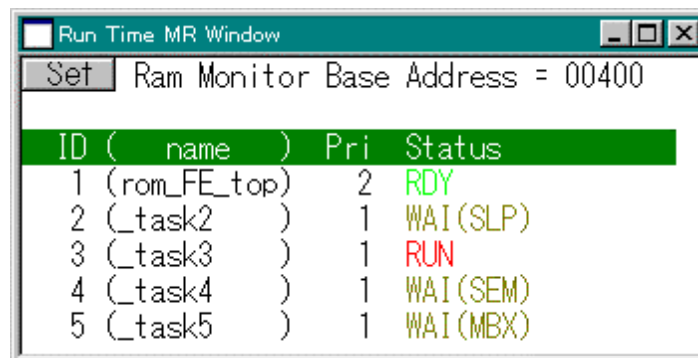


2.6 ランタイムMR ウィンドウ(mr_win)

リアルタイム OS「MR79」のタスク状態をターゲット実行中にサンプリング表示するウィンドウです。エミュレータのリアルタイム RAM モニタ機能を使用しますので、ターゲットの実行を妨げることなく、MR79 のタスク状態を参照することができます。

- <Set>ボタンを入力することにより、ターゲット状態を解析するために必要な設定を行い、リアルタイム RAM モニタ領域(1KB)を MR79 のタスクコントロールブロックが含まれるように設定します。<Set>ボタンは、MR79 を使用したプログラムをダウンロードした後に入力してください。
- MR79 のタスクコントロールブロック全体が、リアルタイム RAM モニタ領域に含まれないほど大きい場合には、本ウィンドウを使用していただくことができません。(エラーメッセージが表示されます。)
- 表示内容は、タスク番号、タスク名、優先度、タスクの状態です。表示内容の詳細については、PD79 ユーザズマニュアルの MR ウィンドウの TSK モードの説明を参照してください。

(ウィンドウ表示例)



The screenshot shows a window titled "Run Time MR Window" with a "Set" button and the text "Ram Monitor Base Address = 00400". Below this is a table with the following data:

ID (name)	Pri	Status
1	(rom_FE_top)	2	RDY
2	(_task2)	1	WAI(SLP)
3	(_task3)	1	RUN
4	(_task4)	1	WAI(SEM)
5	(_task5)	1	WAI(MBX)

2.7 小さなレジスタウィンドウ(regs_win)

全レジスタ値が一度に表示可能な縦長のレジスタウィンドウです。縦長のウィンドウ形式のため、画面を有効に使用していただけます。

- レジスタ内容が変更されるたびに自動的に最新情報に更新されます。
- 更新前と、更新後でデータが異なるものについては、赤色表示します。
- レジスタおよび IPL 値は、値をダブルクリックし、オープンする入力ダイアログに値を入力することにより、変更することができます。
- ビットフラグ値は、値またはフラグ名をダブルクリックすることにより、反転させることができます。

(ウィンドウ表示例)

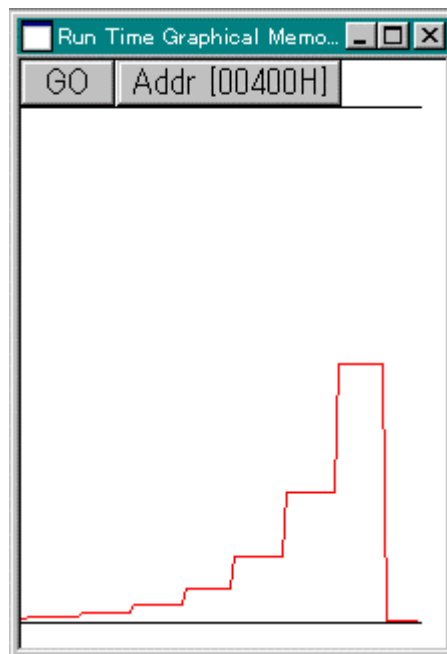
```
Little R...
PG :00
PC :C400
PGPC:00C400
A :54E0
B :FFFF
X :FFFF
Y :FFFF
E :FFFF54E0
S :0FFD
DT :00
DP0 :0000
DP1 :0000
DP2 :0000
DP3 :0000
IPL :0
N V m x
0 0 0 0
D I Z C
0 1 0 0
```

2.8 ランタイムメモリ値グラフ表示ウィンドウ(vmon_win)

指定されたアドレスのメモリ内容を、ターゲット実行中にサンプリングでグラフで表示するウィンドウです。

- <Addr []>ボタンを入力し、オープンする入力ダイアログにアドレス値を入力することにより、グラフに表示する1バイトのメモリのアドレスを設定します。
- <GO>ボタンを入力することにより、ターゲットの実行を開始し、指定されたメモリの内容の変化を折れ線グラフで表示します。
- 指定するアドレスには、リアルタイム RAM モニタ領域に含まれるアドレスを指定してください。リアルタイム RAM モニタ領域外の場合は、サンプリングの度にターゲット実行が妨げられます。

(ウィンドウ表示例)

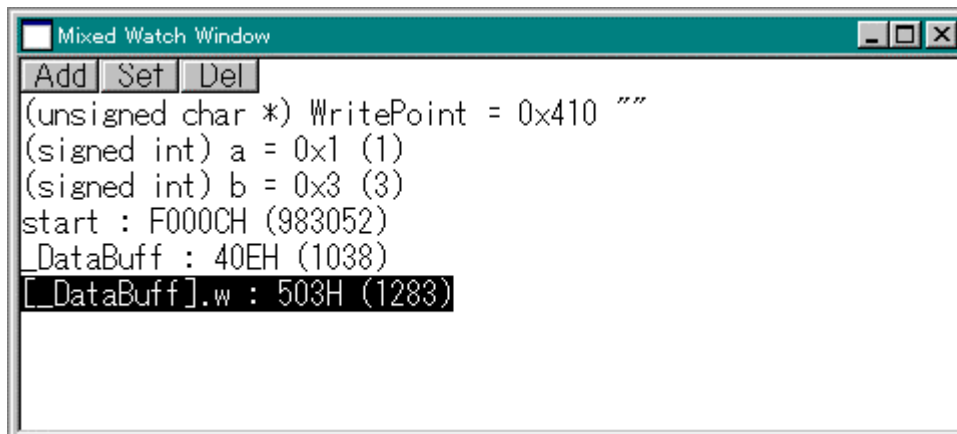


2.9 混合ウォッチウィンドウ(watch_win)

C 言語式、アセンブラ式、ビットシンボルをウォッチ可能なウォッチウィンドウです。値を代入することも可能です。アセンブラ式、C 言語式、ビットシンボルを一つのウィンドウで確認でき、画面を有効に使用していただけます。

- <Add>ボタンを入力すると、オープンする入力ダイアログに、ウォッチしたい式を入力することにより登録されます。入力した式は、C 言語式、アセンブラ式、ビットシンボルの順に検索されます。C 言語式と、アセンブラ式で同じシンボルが存在した場合には、C 言語式であると認識されます。
- 式をクリックし反転表示させて<Set>ボタンを入力し、オープンする入力ダイアログに、C 言語式、アセンブラ式、ビットシンボルを入力すると、値を代入することができます。
- <Set>ボタンの代わりに、式をダブルクリックする操作でも、入力ダイアログがオープンし、値を代入することができます。
- C 言語式に代入する場合には、C 言語式がアドレスを持つ型でなければなりません。また、代入するバイト数は、C 言語式の型のサイズになります。
- ビットシンボルに代入する場合には、値が非ゼロの場合には1がセットされ、ゼロの場合には0がセットされます。
- 式をクリックし反転表示させてボタンを入力することにより、削除することができます。

(ウィンドウ表示例)



```
Mixed Watch Window
Add Set Del
(unsigned char *) WritePoint = 0x410 ""
(signed int) a = 0x1 (1)
(signed int) b = 0x3 (3)
start : F000CH (983052)
_DataBuff : 40EH (1038)
[ _DataBuff ].w : 503H (1283)
```

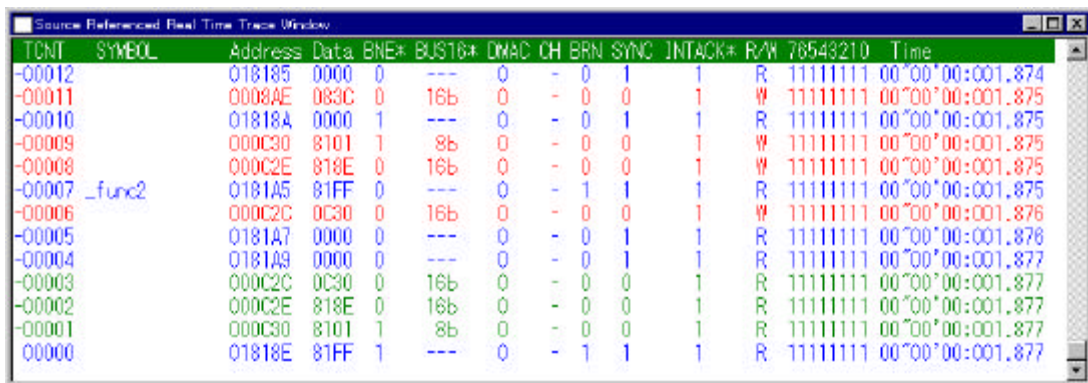
2.10 ソースリファレンス付きRTT ウィンドウ (xrtt_win)

リアルタイムトレース結果の各サイクルを、リードサイクル、ライトサイクル、フェッチサイクルに色分けして表示するウィンドウです。表示されているデータをダブルクリックするとそのサイクルが発生する直前に実行した命令のアドレスに対応するソース行（ソース行情報がないアドレスの場合には、逆アセンブルリスト）をプログラムウィンドウに表示します。

サイクル毎の動作によって色分けして表示されますので、特定動作のサイクルが簡単に確認できます。また、指定したサイクルに対応したソース行をプログラムウィンドウに表示することができますので、注目するサイクルが発生したソースプログラム位置が簡単に確認できます。

- データリードサイクルは緑色で表示されます。
- データライトサイクルは赤色で表示されます。
- 命令フェッチサイクルは青色で表示されます。
- 表示をダブルクリックすると、その行に対応するサイクルが発生する直前に実行した命令のアドレスに対応するソース行をプログラムウィンドウに表示します。ソース行情報がないアドレスの場合には、逆アセンブル表示に切替えます。

(ウィンドウ表示例)



TCNT	SYMBOL	Address	Data	BNE*	BUS16*	DMAC	CH	BRN	SYNC	INTACK*	R/W	76543210	Time
-00012		018185	0000	0	---	0	-	0	1	1	R	11111111	00*00*00:001.874
-00011		0008AE	083C	0	16b	0	-	0	0	1	W	11111111	00*00*00:001.875
-00010		01818A	0000	1	---	0	-	0	1	1	R	11111111	00*00*00:001.875
-00009		000C30	8101	1	8b	0	-	0	0	1	W	11111111	00*00*00:001.875
-00008		000C2E	818E	0	16b	0	-	0	0	1	W	11111111	00*00*00:001.875
-00007	_func2	0181A5	81FF	0	---	0	-	1	1	1	R	11111111	00*00*00:001.875
-00006		000C2C	0C30	0	16b	0	-	0	0	1	W	11111111	00*00*00:001.876
-00005		0181A7	0000	0	---	0	-	0	1	1	R	11111111	00*00*00:001.876
-00004		0181A9	0000	0	---	0	-	0	1	1	R	11111111	00*00*00:001.877
-00003		000C2C	0C30	0	16b	0	-	0	0	1	R	11111111	00*00*00:001.877
-00002		000C2E	818E	0	16b	0	-	0	0	1	R	11111111	00*00*00:001.877
-00001		000C30	8101	1	8b	0	-	0	0	1	R	11111111	00*00*00:001.877
00000		01818E	81FF	1	---	0	-	1	1	1	R	11111111	00*00*00:001.877