Application Note for Trace.Find

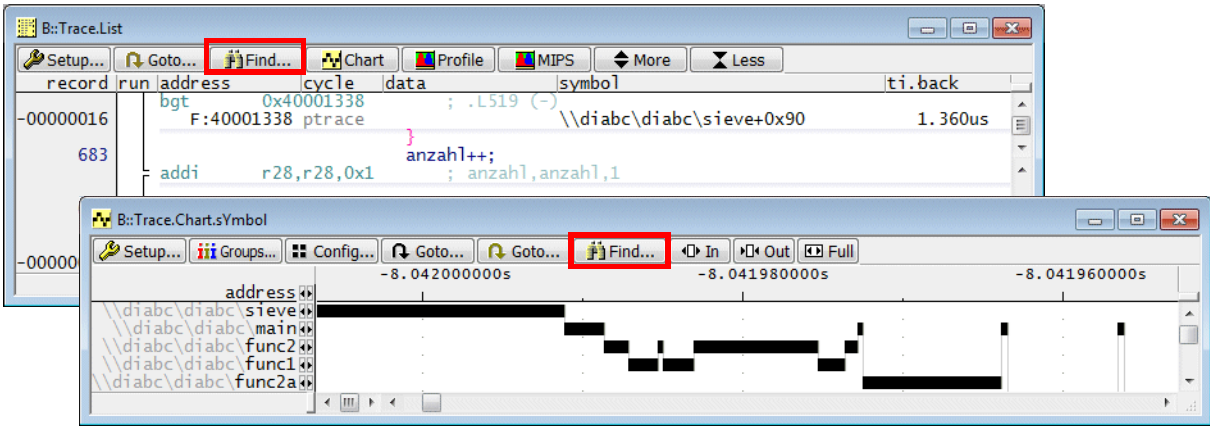
Application Note for Trace.Find

Version 06-Jun-2024

The Trace Find Dialog

NOTE:	The functions of the **Trace Find** dialog are presented by using a trace which contains the executed instructions and the performed read / write cycles. The presented functionality can of course be applied to other trace contents.

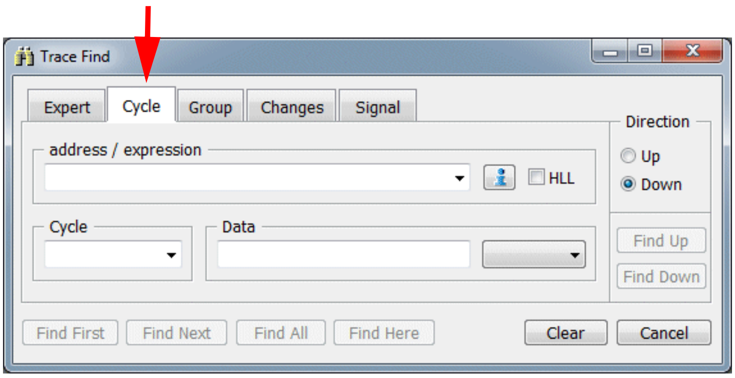
The **Trace.List** window and windows that display a graphical representation of the recorded trace information provide a “**Find…**” button to open the **Trace Find** dialog. The **Trace Find** dialog allows to search for events of interest in the trace.


B::Trace.List
Setup... Goto... Find... Chart Profile MIPS More Less
B::Trace.Chart.sYmbol
Setup... Groups... Config... Goto... Goto... Find... In Out Full

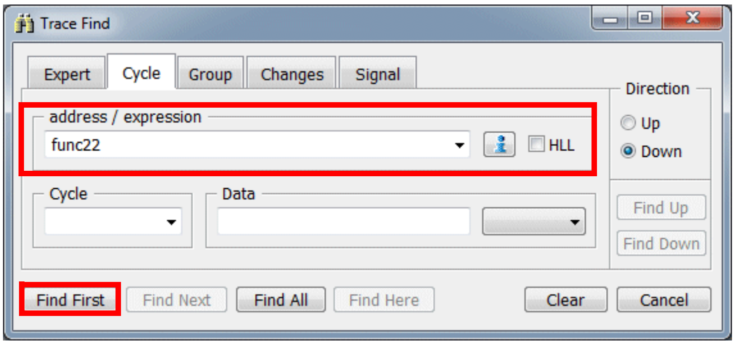
```
 record run address          cycle  data                 symbol                        ti.back
           bgt          0x40001338       ; .L519 (-)
-00000016    F:40001338 ptrace                          \\diabc\diabc\sieve+0x90       1.360us
                                     }
      683                            anzahl++;
           addi         r28,r28,0x1      ; anzahl,anzahl,1
```

```
                address   -8.042000000s          -8.041980000s          -8.041960000s
\\diabc\diabc\sieve
\\diabc\diabc\main
\\diabc\diabc\func2
\\diabc\diabc\func1
\\diabc\diabc\func2a
```

The **Trace Find** dialog opens in the **Cycle** tab by default.



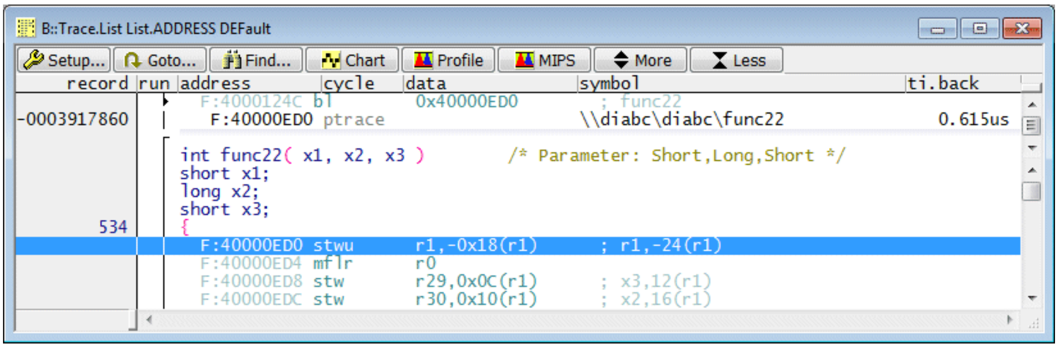
First example: Find the entry to the function func22 in the trace recording.



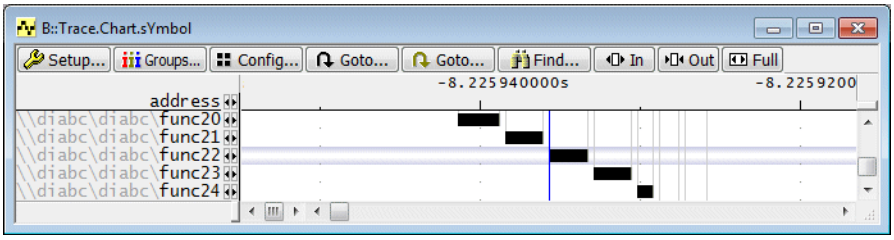
address / expression: func22

Direction: Up [✓ Down]

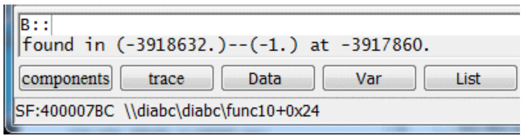
If the specified event is found the cursor is positioned on the trace line or on the point of time in the trace recording.



```
     record |run |address        |cycle    |data              |symbol                        |ti.back
                  F:4000124C  bl           0x40000ED0            ; func22
-0003917860  |      F:40000ED0 ptrace                          \\diabc\diabc\func22              0.615us

              int func22( x1, x2, x3 )         /* Parameter: Short,Long,Short */
              short x1;
              long x2;
              short x3;
         534  {
                  F:40000ED0 stwu          r1,-0x18(r1)          ; r1,-24(r1)
                  F:40000ED4 mflr          r0
                  F:40000ED8 stw           r29,0x0C(r1)          ; x3,12(r1)
                  F:40000EDC stw           r30,0x10(r1)          ; x2,16(r1)
```



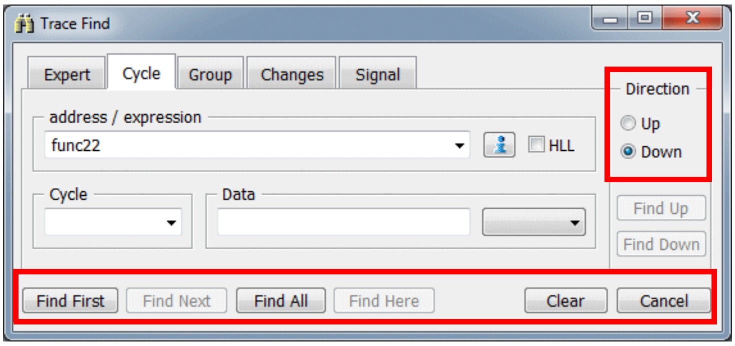
The status line indicates which record range was searched and under which record number the specified event was found.



```
B::
found in (-3918632.)--(-1.) at -3917860.
```

SF:400007BC \\diabc\diabc\func10+0x24

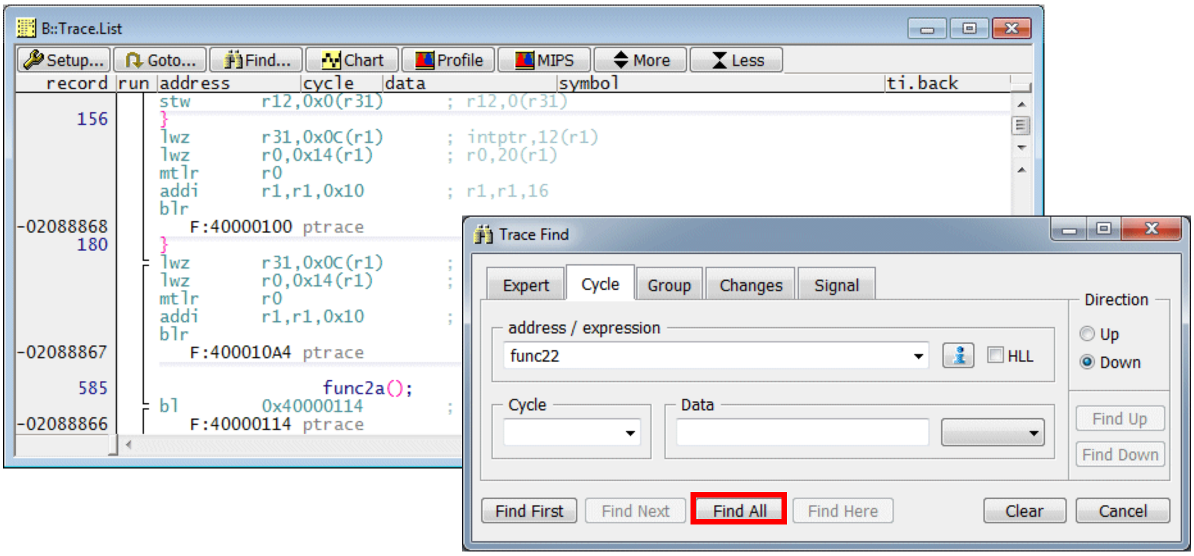
Push Buttons



The push buttons provided by the **Trace Find** dialog perform as follows:

Push button	Direction Radio button	Search procedure
Find First	**Down**	Start searching at first record down to the last record.
Find First	**Up**	Start searching at last record up to the first record.
Find Here	**Down**	Start searching at currently selected record down to the last record.
Find Here	**Up**	Start searching at currently selected record up to the first record.
Find Next	**Down**	Same as “Find Here”, has only a special function in the **Changes** tab.
Find Next	**Up**	Same as “Find Here”, has only a special function in the **Changes** tab.
Find All	—	Search for all occurrences of the specified event.
Clear	—	Clear the search specification.
Cancel	—	Close the Trace Find dialog.

Example for Find All: Find all entries to the function func22 in the trace.



B::Trace.List

```
   record run address           cycle    data                  symbol                    ti.back
                stw     r12,0x0(r31)       ; r12,0(r31)
      156     }
                lwz     r31,0x0C(r1)       ; intptr,12(r1)
                lwz     r0,0x14(r1)        ; r0,20(r1)
                mtlr    r0
                addi    r1,r1,0x10         ; r1,r1,16
                blr
-02088868          F:40000100 ptrace
      180     }
                lwz     r31,0x0C(r1)       ;
                lwz     r0,0x14(r1)        ;
                mtlr    r0
                addi    r1,r1,0x10         ;
                blr
-02088867          F:400010A4 ptrace
      585                         func2a();
                bl      0x40000114         ;
-02088866          F:40000114 ptrace
```

Trace Find

address / expression: func22

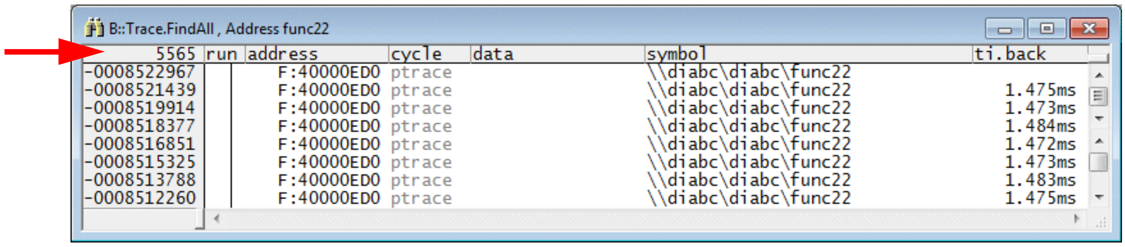
HLL: ☐

Direction: Up [✓ Down]

Cycle:

Data:

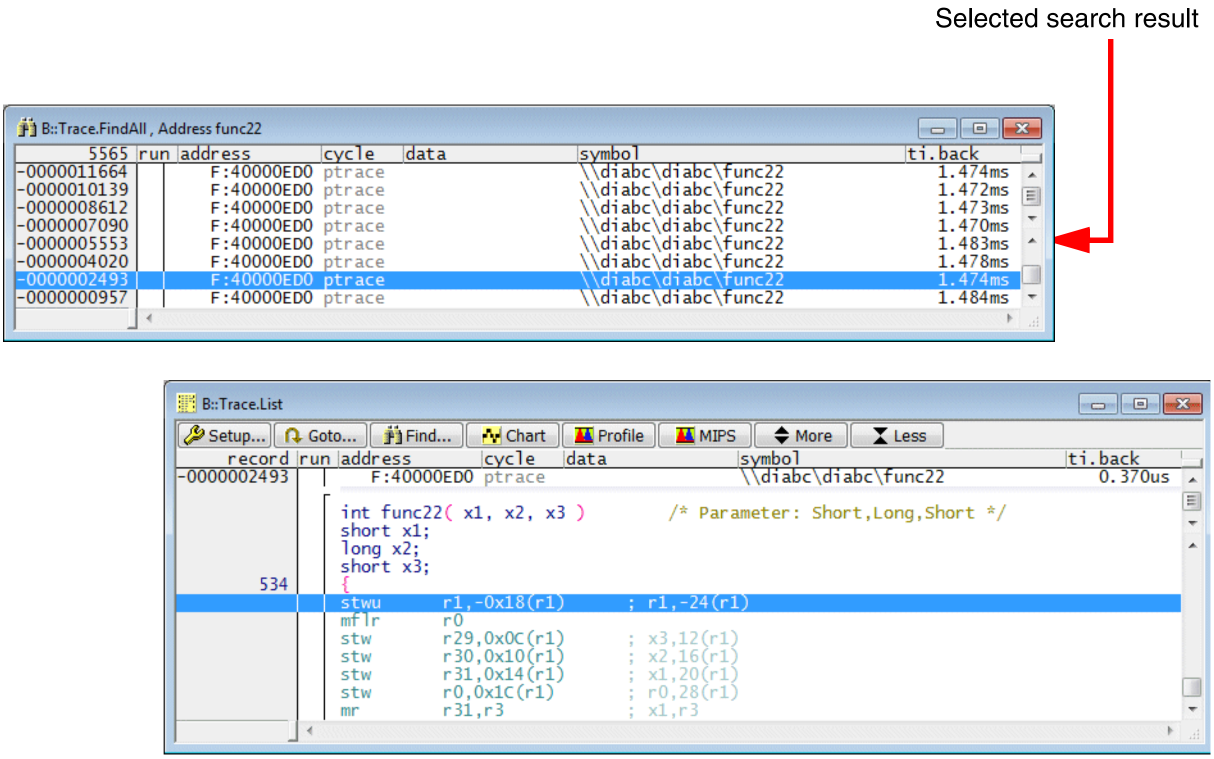
The **Trace.FindAll** window lists all occurrences of the specified event and the total number of events found.



B::Trace.FindAll , Address func22

```
        5565 run address        cycle  data                  symbol                    ti.back
-0008522967          F:40000ED0 ptrace                       \\diabc\diabc\func22
-0008521439          F:40000ED0 ptrace                       \\diabc\diabc\func22        1.475ms
-0008519914          F:40000ED0 ptrace                       \\diabc\diabc\func22        1.473ms
-0008518377          F:40000ED0 ptrace                       \\diabc\diabc\func22        1.484ms
-0008516851          F:40000ED0 ptrace                       \\diabc\diabc\func22        1.472ms
-0008515325          F:40000ED0 ptrace                       \\diabc\diabc\func22        1.473ms
-0008513788          F:40000ED0 ptrace                       \\diabc\diabc\func22        1.483ms
-0008512260          F:40000ED0 ptrace                       \\diabc\diabc\func22        1.475ms
```

If a search result is selected in the **Trace.FindAll** window, the matching record in the trace display window is selected (blue cursor).



```
     5565 run address         cycle   data            symbol                        ti.back
-0000011664      F:40000ED0  ptrace                  \\diabc\diabc\func22          1.474ms
-0000010139      F:40000ED0  ptrace                  \\diabc\diabc\func22          1.472ms
-0000008612      F:40000ED0  ptrace                  \\diabc\diabc\func22          1.473ms
-0000007090      F:40000ED0  ptrace                  \\diabc\diabc\func22          1.470ms
-0000005553      F:40000ED0  ptrace                  \\diabc\diabc\func22          1.483ms
-0000004020      F:40000ED0  ptrace                  \\diabc\diabc\func22          1.478ms
-0000002493      F:40000ED0  ptrace                  \\diabc\diabc\func22          1.474ms
-0000000957      F:40000ED0  ptrace                  \\diabc\diabc\func22          1.484ms
```

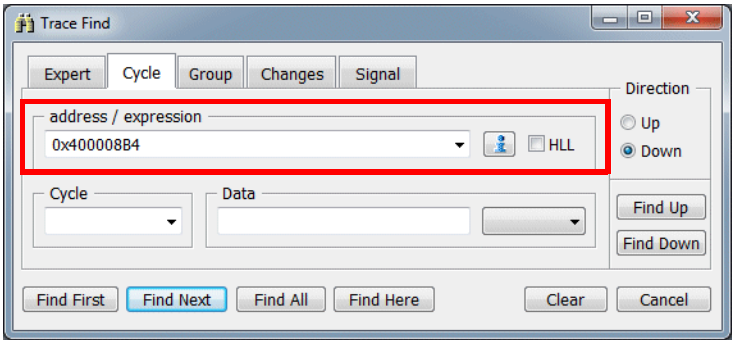
```
    record run address        cycle   data              symbol                      ti.back
-0000002493        F:40000ED0  ptrace                   \\diabc\diabc\func22        0.370us

          int func22( x1, x2, x3 )        /* Parameter: Short,Long,Short */
          short x1;
          long x2;
          short x3;
      534 {
          stwu       r1,-0x18(r1)         ; r1,-24(r1)
          mflr       r0
          stw        r29,0x0C(r1)         ; x3,12(r1)
          stw        r30,0x10(r1)         ; x2,16(r1)
          stw        r31,0x14(r1)         ; x1,20(r1)
          stw        r0,0x1C(r1)          ; r0,28(r1)
          mr         r31,r3               ; x1,r3
```

Cycle Tab (default)

address/expression Field

The **address/expression** field of the **Trace Find** dialog allows to search for the event of interest by specifying an address or an expression.

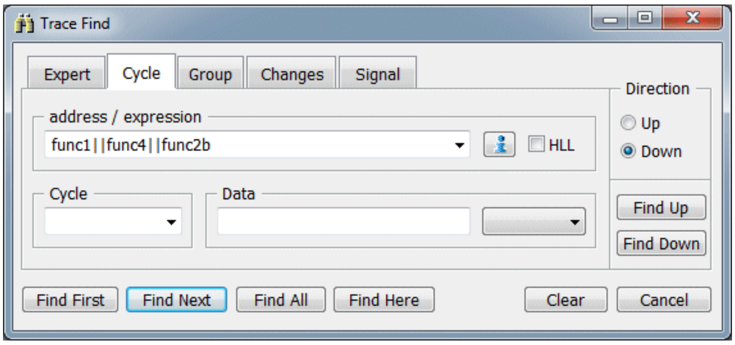
- If the **HLL** check box is unchecked the entered string is interpreted as an address or a TRACE32 expression.
- The **HLL** check box is checked, the entered string is interpreted as an expression by applying the rules of the programming language in use.



address/expression	**HLL check box**	**Search event**
0x400008B4--0x400008BF	Off	Search for an address within the specified address range.
func22	Off	Search for the entry to func22 (first address of function func22).
func22	On	Search for an address within the function func22 (all addresses of the function func22).
flags	On	Search for the variable flags (all addresses of variable flags).
flags[3]	On	Search for the 4th element of the array flags (address of 4th element of variable flags).

If you want to specify several addresses you can specify them by using the OR operator.

- Please be aware that no spaces are allowed.
- Please be aware that this is only supported when the **HLL** check box is unchecked.

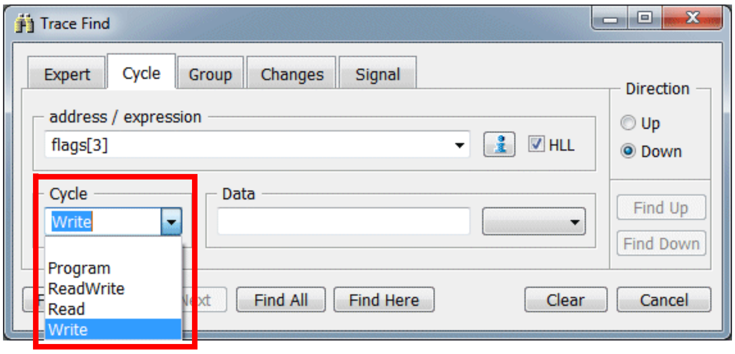


address / expression: func1||func4||func2b

HLL: unchecked

Cycle Field

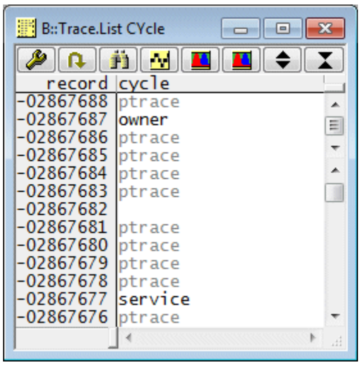
The **Cycle** field of the **Trace Find** dialog allows you to refine your search by adding a cycle type for the specified address.



Default Cycle types	
Program	Program address.
ReadWrite	Address of read or write cycle.
Read	Address of read cycle.
Write	Address of write cycle.

Beside the pre-defined cycle types all cycle types listed in the **cycle** column of the **Trace.List** window can be entered into the **Cycle** field.

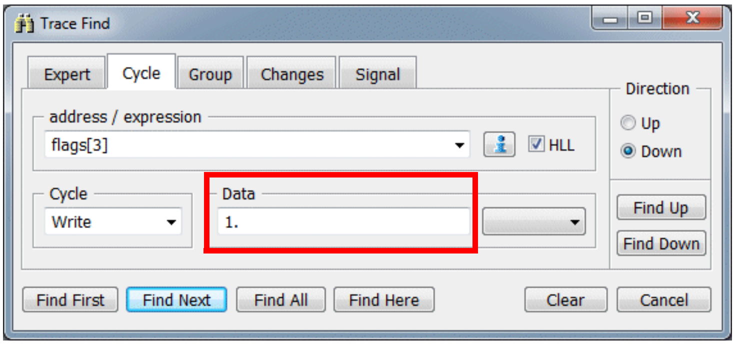
```
Trace.List CYcle                      ; display Trace.List window
                                      ; showing the cycle column
```



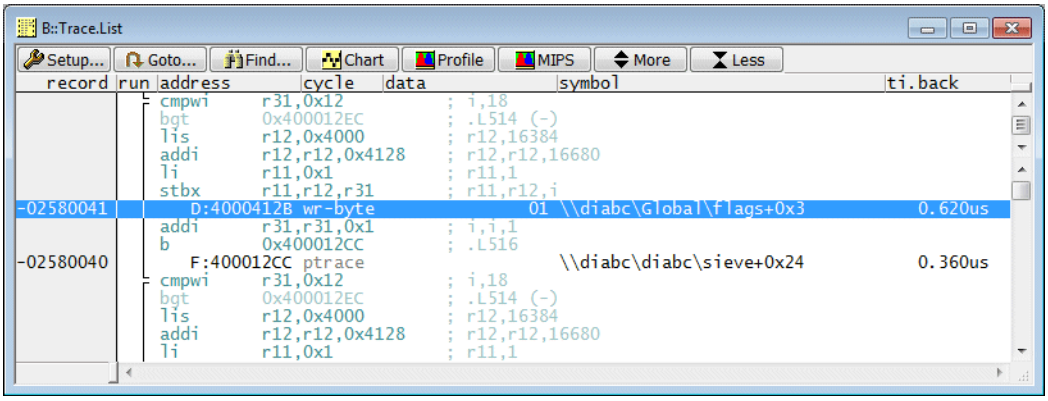
```
  record  cycle
-02867688 ptrace
-02867687 owner
-02867686 ptrace
-02867685 ptrace
-02867684 ptrace
-02867683 ptrace
-02867682
-02867681 ptrace
-02867680 ptrace
-02867679 ptrace
-02867678 ptrace
-02867677 service
-02867676 ptrace
```

Data Field

The **Data** field of the **Trace Find** dialog allows you to further refine your search by adding a **Data** value for the specified address/cycle.



address / expression: flags[3]
[✓ HLL]
Cycle: Write
Data: 1.
Direction: Up [✓ Down]



```
     record run address          cycle    data              symbol                         ti.back
                 ┌ cmpwi      r31,0x12              ; i,18
                 │ bgt        0x400012EC            ; .L514 (-)
                 │ lis        r12,0x4000            ; r12,16384
                 │ addi       r12,r12,0x4128        ; r12,r12,16680
                 │ li         r11,0x1               ; r11,1
                 │ stbx       r11,r12,r31           ; r11,r12,i
-02580041        │    D:4000412B wr-byte                  01 \\diabc\Global\flags+0x3      0.620us
                 │ addi       r31,r31,0x1           ; i,i,1
                 │ b          0x400012CC            ; .L516
-02580040        │    F:400012CC ptrace                      \\diabc\diabc\sieve+0x24     0.360us
                 ┌ cmpwi      r31,0x12              ; i,18
                 │ bgt        0x400012EC            ; .L514 (-)
                 │ lis        r12,0x4000            ; r12,16384
                 │ addi       r12,r12,0x4128        ; r12,r12,16680
                 │ li         r11,0x1               ; r11,1
```

More examples for the **Data** field.

Data	Search event
0x10	Hex. value 0x10
10.	Decimal value 10.
!0.	Not 0.
149.||191.||210.	Decimal value 149. or 191. or 210.
0x10--0x1F	Data value between 0x10 and 0x1F.
0100.--0x199.	Data value between 100 and 199 (decimal).
!(0100.--0x199.)	Data values not between 100 and 199.

Data	Search event
120.--150.||200.--220.	Data value between 120. and 150. or between 200. and 220.
0yxxxxxx11	Data value as byte bit mask.

Expert Tab

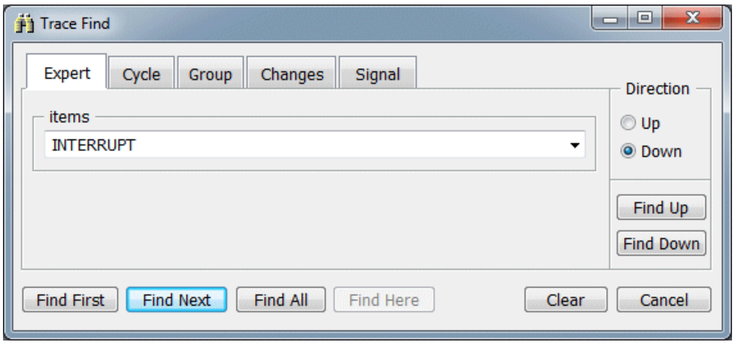
The **Expert** tab can be used for one of the following purposes:

- to convert a search specified in the **Trace Find** dialog into a TRACE32 command. For details refer to **"Convert Setting in Trace Find Dialog to a TRACE32 Command"**, page 45.
- to search for various special events.
- to extent the features provided by the **Cycle** tab.

Special Event Interrupts, Traps etc.

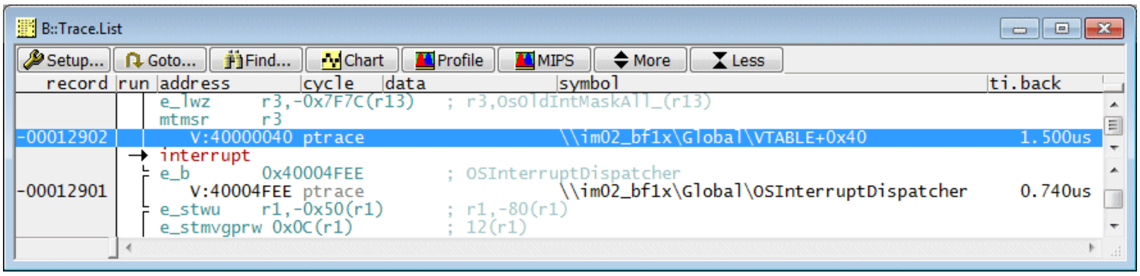
TRACE32 marks interrupts, traps and exceptions in the trace display. How interrupts, traps and exceptions are detected depends on the trace protocol and the processor architecture in use.

Example: Search for interrupts.



items: INTERRUPT

Direction: Up [✓ Down]



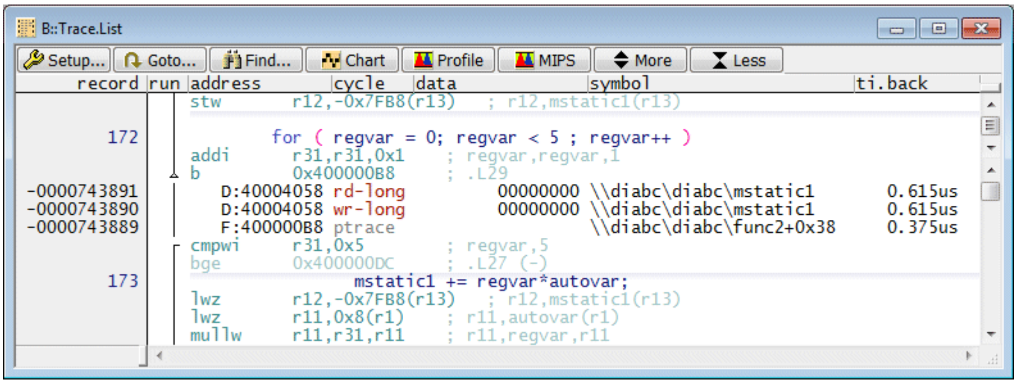
```
  record |run |address         |cycle  |data            |symbol                               |ti.back
              e_lwz    r3,-0x7F7C(r13)   ; r3,OsOldIntMaskAll_(r13)
              mtmsr    r3
-00012902        V:40000040 ptrace                       \\im02_bf1x\Global\VTABLE+0x40           1.500us
           -> interrupt
              e_b      0x40004FEE        ; OSInterruptDispatcher
-00012901        V:40004FEE ptrace                       \\im02_bf1x\Global\OSInterruptDispatcher 0.740us
              e_stwu   r1,-0x50(r1)      ; r1,-80(r1)
              e_stmvgprw 0x0C(r1)        ; 12(r1)
```

Keywords	
INTERRUPT	Search for interrupt.
TRAP	Search for trap.
EXCEPTION	Search for interrupt or trap.

Special Event UNALIGNED

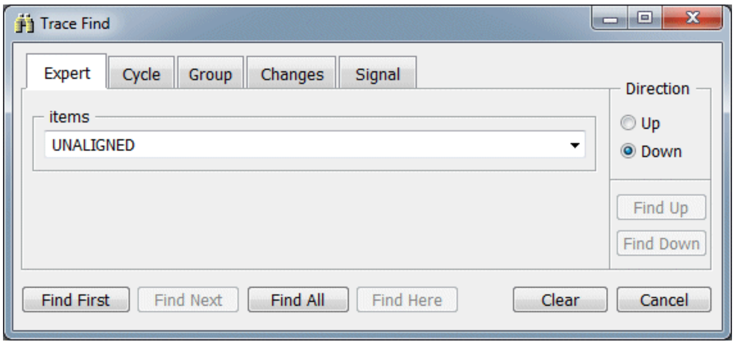
If the trace generation logic generates trace information for the executed instructions and the read/write accesses, TRACE32 assigns the read/write accesses to the corresponding load/store instructions if possible. The technique used is called Data Cycle Assignment.

Read/write accesses that can not be assigned to the corresponding load/store instructions are printed in red in the **Trace.List** window. They are listed preceding the next program trace message (ptrace).



```
     record |run |address        |cycle   |data         |symbol                       |ti.back
                 stw       r12,-0x7FB8(r13)    ; r12,mstatic1(r13)

        172              for ( regvar = 0; regvar < 5 ; regvar++ )
                 addi      r31,r31,0x1     ; regvar,regvar,1
                 b         0x400000B8      ; .L29
-0000743891         D:40004058 rd-long          00000000 \\diabc\diabc\mstatic1        0.615us
-0000743890         D:40004058 wr-long          00000000 \\diabc\diabc\mstatic1        0.615us
-0000743889         F:400000B8 ptrace                    \\diabc\diabc\func2+0x38      0.375us
                 cmpwi     r31,0x5         ; regvar,5
                 bge       0x400000DC      ; .L27 (-)
        173                     mstatic1 += regvar*autovar;
                 lwz       r12,-0x7FB8(r13)    ; r12,mstatic1(r13)
                 lwz       r11,0x8(r1)     ; r11,autovar(r1)
                 mullw     r11,r31,r11     ; r11,regvar,r11
```

The keyword UNALIGNED allows to search for such data accesses.



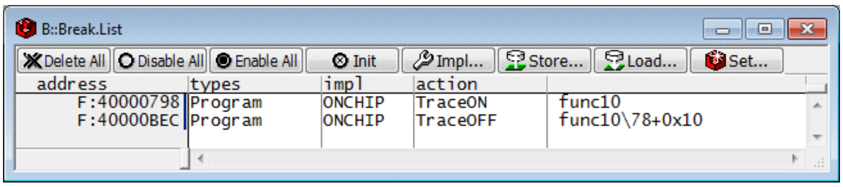
items: UNALIGNED

Direction: Up [✓ Down]

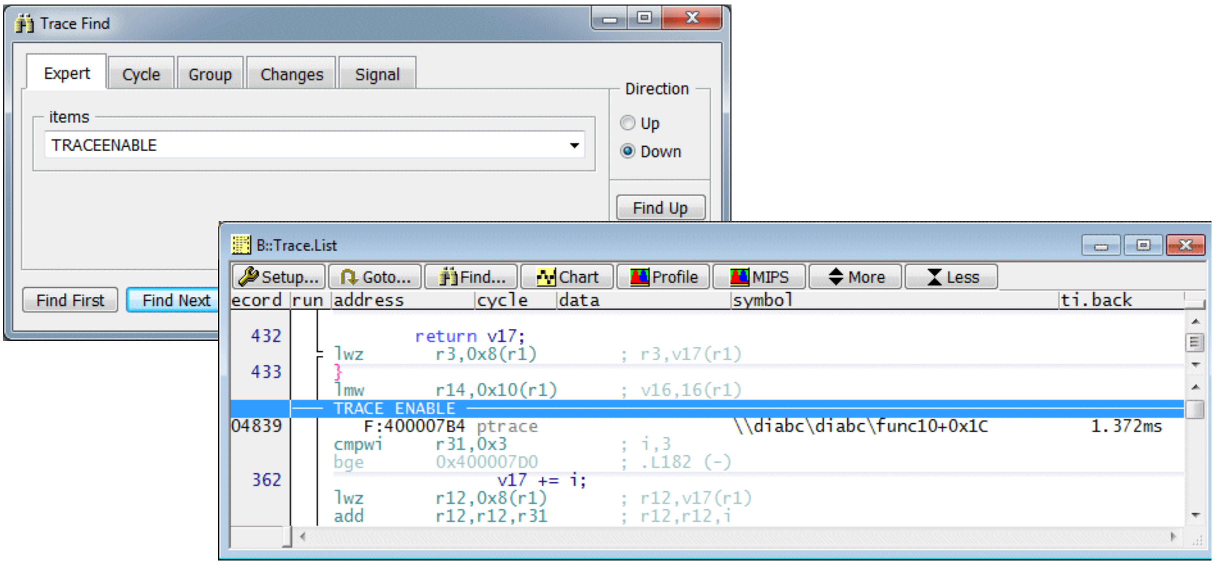
Special Event TRACEENABLE

The on-chip trace generation logic can be programmed so that it only generates trace information for specified program sections. TRACE32 PowerView marks the start points of the trace generation with TRACE ENABLE in the **Trace.List** window.

Example: Advise the trace generation logic to only generate trace information for the function func10. Search for the trace generation starting points in the trace recording afterwards.



```
address      types    impl    action
  F:40000798 Program  ONCHIP  TraceON   func10
  F:40000BEC Program  ONCHIP  TraceOFF  func10\78+0x10
```



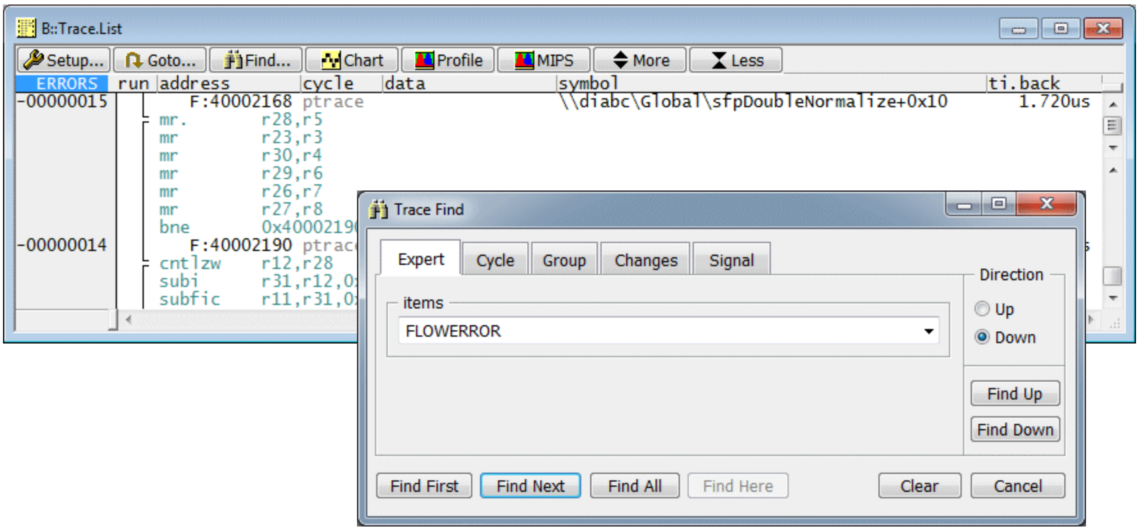
items: TRACEENABLE

Direction: Up [✓ Down]

```
ecord |run |address          |cycle  |data              |symbol                        |ti.back
  432          return v17;
        lwz      r3,0x8(r1)          ; r3,v17(r1)
  433   }
        lmw      r14,0x10(r1)        ; v16,16(r1)
      ── TRACE ENABLE ──────────────────────────────────────────────────────
04839      F:400007B4 ptrace                     \\diabc\diabc\func10+0x1C          1.372ms
        cmpwi    r31,0x3             ; i,3
        bge      0x400007D0          ; .L182 (-)
  362              v17 += i;
        lwz      r12,0x8(r1)         ; r12,v17(r1)
        add      r12,r12,r31         ; r12,r12,i
```

Special Event FLOWERRORs

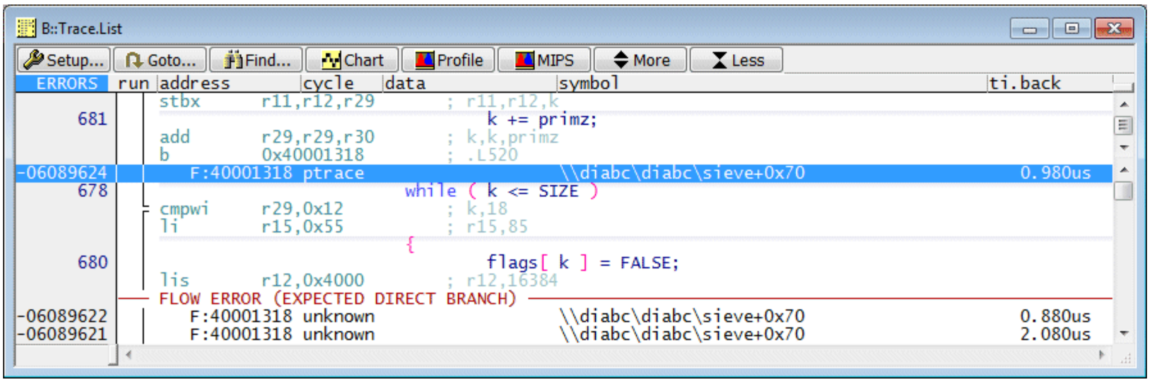
If your trace display indicates **ERRORS** you can search for their actual occurrence by using the keyword **FLOWERROR** in the **Expert** tab of the **Trace Find** dialog.


B::Trace.List
Setup... Goto... Find... Chart Profile MIPS More Less
Trace Find
Expert Cycle Group Changes Signal
items
Direction
Up
Down
Find Up
Find Down
Find First Find Next Find All Find Here Clear Cancel

```
ERRORS     run  address              cycle  data      symbol                              ti.back
-00000015       F:40002168 ptrace                     \\diabc\Global\sfpDoubleNormalize+0x10   1.720us
                mr.      r28,r5
                mr       r23,r3
                mr       r30,r4
                mr       r29,r6
                mr       r26,r7
                mr       r27,r8
                bne      0x4000219
-00000014       F:40002190 ptrac
                cntlzw   r12,r28
                subi     r31,r12,0
                subfic   r11,r31,0
```

Direction: Up [✓ Down]

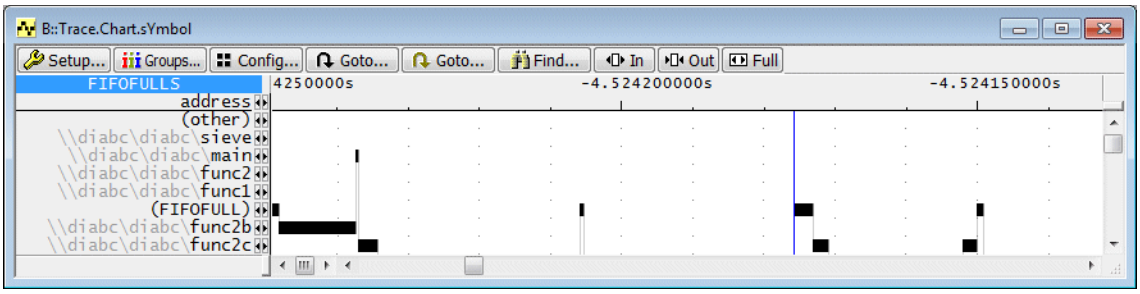
items: FLOWERROR



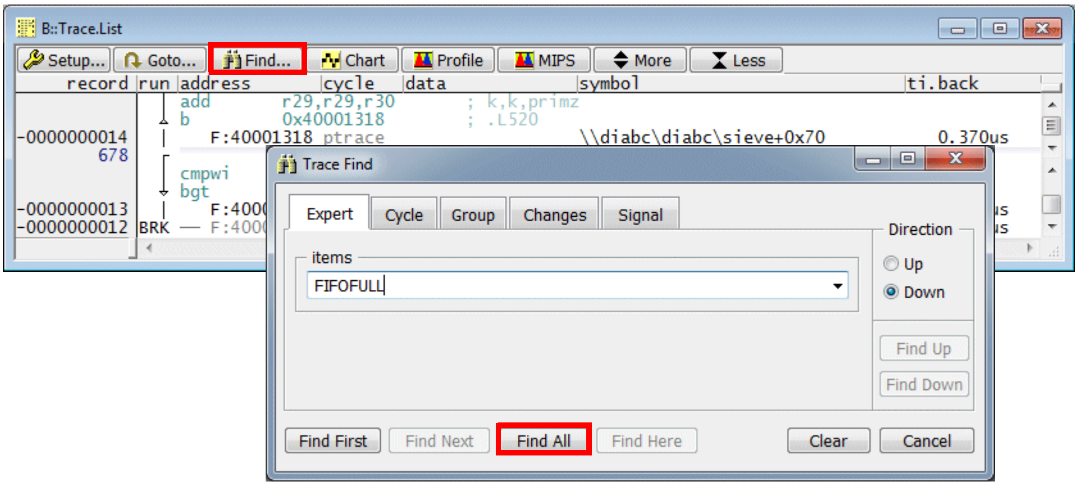
```
ERRORS     run  address              cycle  data      symbol                              ti.back
                stbx     r11,r12,r29          ; r11,r12,k
681                                       k += primz;
                add      r29,r29,r30          ; k,k,primz
                b        0x40001318           ; .L520
-06089624       F:40001318 ptrace             \\diabc\diabc\sieve+0x70                 0.980us
678                               while ( k <= SIZE )
                cmpwi    r29,0x12             ; k,18
                li       r15,0x55             ; r15,85
                                  {
680                                       flags[ k ] = FALSE;
                lis      r12,0x4000           ; r12,16384
         ---- FLOW ERROR (EXPECTED DIRECT BRANCH) ----
-06089622       F:40001318 unknown            \\diabc\diabc\sieve+0x70                 0.880us
-06089621       F:40001318 unknown            \\diabc\diabc\sieve+0x70                 2.080us
```

Special Event FIFOFULL

If your trace display indicates **FIFOFULLS** you can search for their actual occurrence by using the keyword **FIFOFULL** in the **Expert** tab of the **Trace Find** dialog.



```
FIFOFULLS              4250000s           -4.524200000s              -4.524150000s
              address
              (other)
 \\diabc\diabc\sieve
  \\diabc\diabc\main
 \\diabc\diabc\func2
 \\diabc\diabc\func1
           (FIFOFULL)
\\diabc\diabc\func2b
\\diabc\diabc\func2c
```

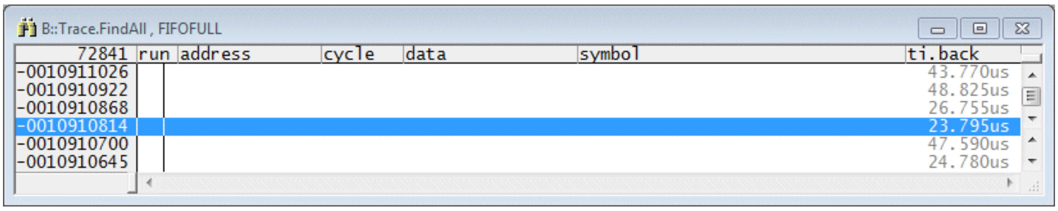


```
      record run address         cycle   data        symbol                       ti.back
              add        r29,r29,r30       ; k,k,primz
              b          0x40001318        ; .L520
-0000000014       F:40001318 ptrace               \\diabc\diabc\sieve+0x70        0.370us
         678
              cmpwi
              bgt
-0000000013       F:4000                                                              us
-0000000012 BRK   F:4000                                                              us
```

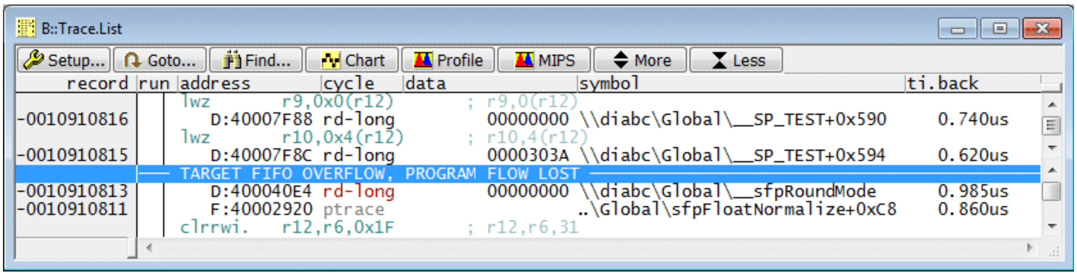
Trace Find: tabs Expert | Cycle | Group | Changes | Signal

items: FIFOFULL

Direction: ( ) Up [✓ Down]



```
      72841 run address         cycle   data        symbol                 ti.back
-0010911026                                                                43.770us
-0010910922                                                                48.825us
-0010910868                                                                26.755us
-0010910814                                                                23.795us
-0010910700                                                                47.590us
-0010910645                                                                24.780us
```

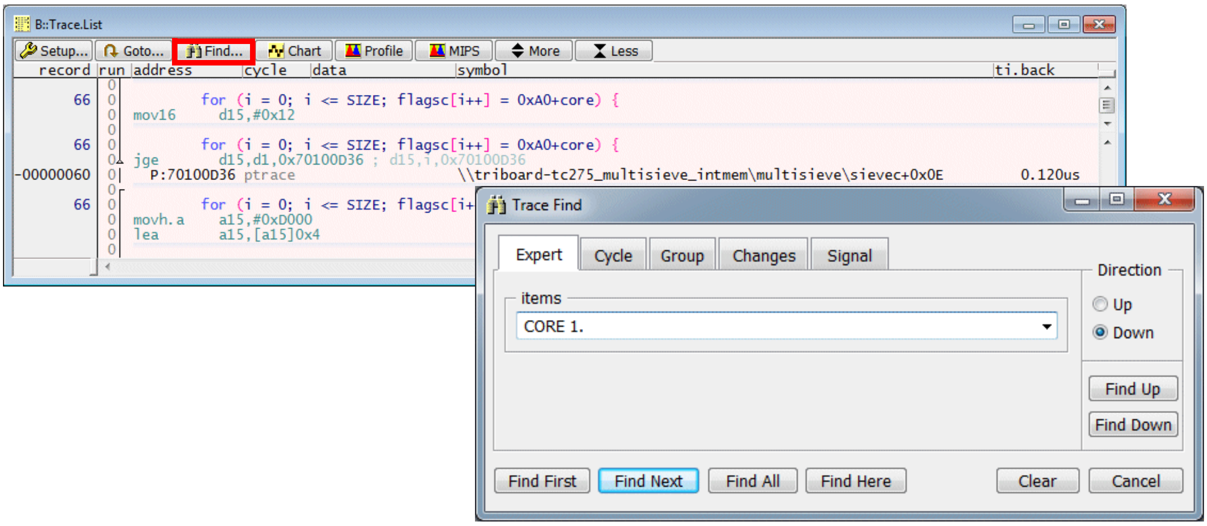


```
      record run address         cycle   data        symbol                                 ti.back
                lwz       r9,0x0(r12)        ; r9,0(r12)
-0010910816        D:40007F88 rd-long     00000000 \\diabc\Global\__SP_TEST+0x590          0.740us
                lwz       r10,0x4(r12)       ; r10,4(r12)
-0010910815        D:40007F8C rd-long     0000303A \\diabc\Global\__SP_TEST+0x594          0.620us
            ---- TARGET FIFO OVERFLOW, PROGRAM FLOW LOST -----------------------------------------
-0010910813        D:400040E4 rd-long     00000000 \\diabc\Global\__sfpRoundMode           0.985us
-0010910811        F:40002920 ptrace               ..\Global\sfpFloatNormalize+0xC8        0.860us
                clrrwi.   r12,r6,0x1F        ; r12,r6,31
```

Trace Information for a Specified Core (SMP only)

TRACE32 analyzes and displays the trace information of all cores together, if trace information is recorded for an SMP system. **CORE** *<number>* allows to search in the trace for information generated by the specified core.

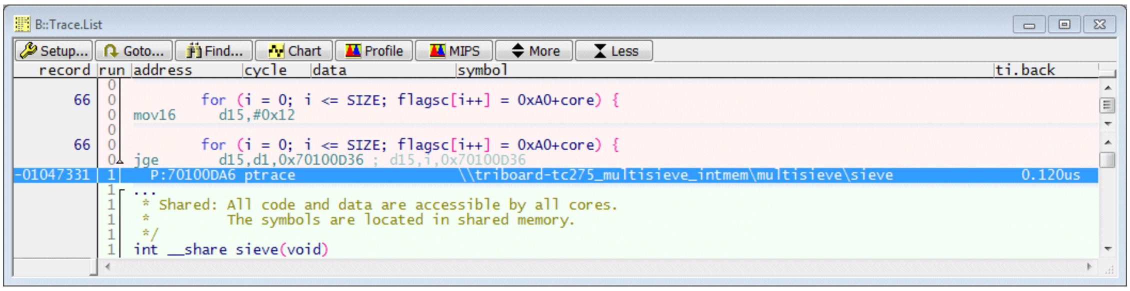
Example: Search for trace information generated by core 1.



```
  record run address          cycle   data              symbol                                          ti.back
          0
      66  0       for (i = 0; i <= SIZE; flagsc[i++] = 0xA0+core) {
          0 mov16      d15,#0x12
          0
      66  0       for (i = 0; i <= SIZE; flagsc[i++] = 0xA0+core) {
          0 jge        d15,d1,0x70100D36 ; d15,i,0x70100D36
-00000060 0   P:70100D36 ptrace                \\triboard-tc275_multisieve_intmem\multisieve\sievec+0x0E      0.120us
          0
      66  0       for (i = 0; i <= SIZE; flagsc[i+
          0 movh.a     a15,#0xD000
          0 lea        a15,[a15]0x4
          0
```

items: CORE 1.

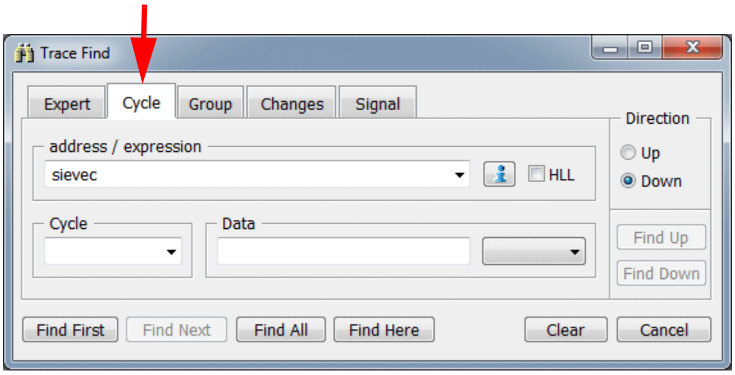
Direction: Up [✓ Down]



```
  record run address          cycle   data              symbol                                          ti.back
          0
      66  0       for (i = 0; i <= SIZE; flagsc[i++] = 0xA0+core) {
          0 mov16      d15,#0x12
          0
      66  0       for (i = 0; i <= SIZE; flagsc[i++] = 0xA0+core) {
          0 jge        d15,d1,0x70100D36 ; d15,i,0x70100D36
-01047331 1   P:70100DA6 ptrace                \\triboard-tc275_multisieve_intmem\multisieve\sieve            0.120us
          1 ...
          1  * Shared: All code and data are accessible by all cores.
          1  *         The symbols are located in shared memory.
          1  */
          1 int __share sieve(void)
```

Example for an SMP system: Find all entries to the function sievec executed on core 0.

Enter the function address in the Cycle tab

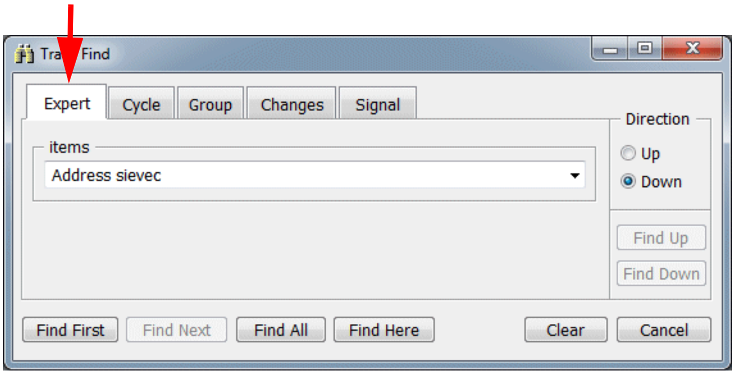


address / expression: sievec

Cycle:

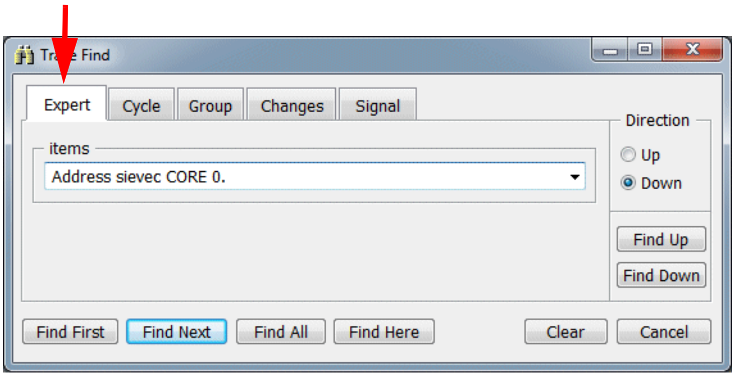
Data:

Switch to the Expert tab



items: Address sievec

Add the CORE *<number>* event

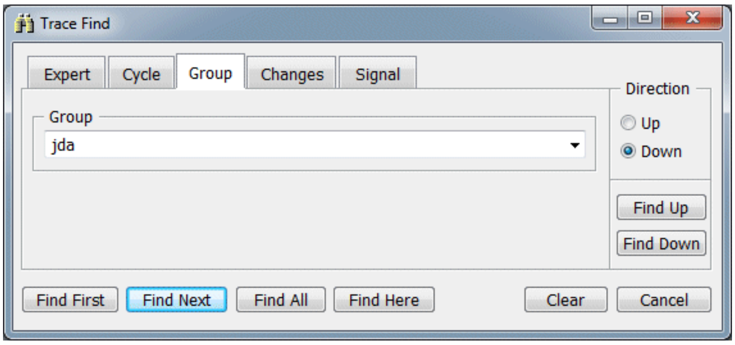


items: Address sievec CORE 0.

Group Tab

The **GROUP** command allows to structure programs consisting of a huge number of functions/modules to ease the debugging process and the evaluation of the trace contents.

The **Group** tab allows you to search for trace information generated for a specified group.

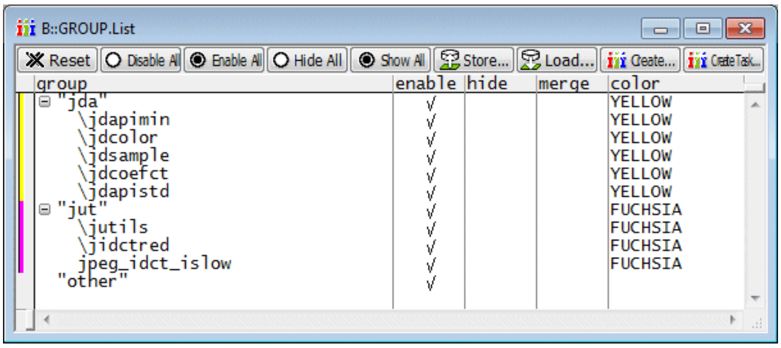


Group: jda

Direction: Up [✓ Down]

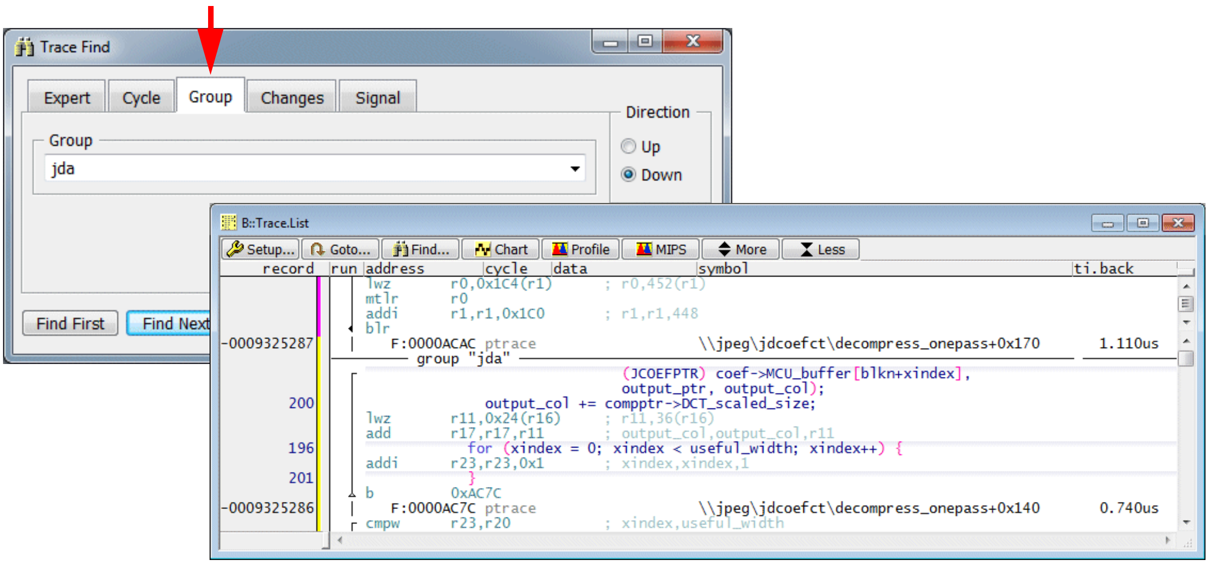
Example:

The **GROUP.List** command displays the known groups.



```
group                 enable hide   merge  color
⊟ "jda"                 √                  YELLOW
    \jdapimin           √                  YELLOW
    \jdcolor            √                  YELLOW
    \jdsample           √                  YELLOW
    \jdcoefct           √                  YELLOW
    \jdapistd           √                  YELLOW
⊟ "jut"                 √                  FUCHSIA
    \jutils             √                  FUCHSIA
    \jidctred           √                  FUCHSIA
    jpeg_idct_islow     √                  FUCHSIA
  "other"               √
```

Find trace information generated for the group “jda”.



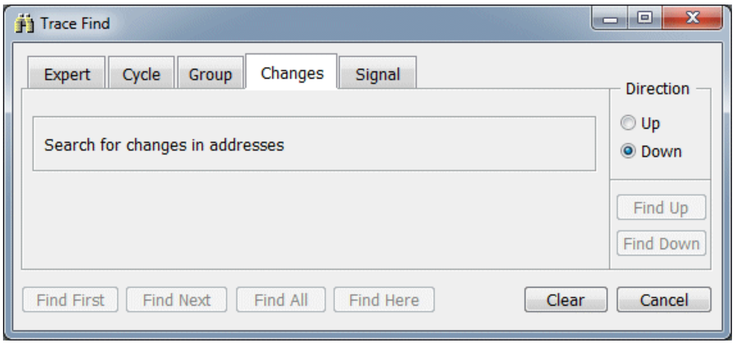
Group: jda

Direction: Up [✓ Down]

```
       record |run |address       |cycle  |data                |symbol                                      |ti.back
                    lwz          r0,0x1C4(r1)         ; r0,452(r1)
                    mtlr         r0
                    addi         r1,r1,0x1C0          ; r1,r1,448
                    blr
-0009325287            F:0000ACAC ptrace                 \\jpeg\jdcoefct\decompress_onepass+0x170     1.110us
                 ────── group "jda" ──────────────────────────────────────────────────────────
                                                        (JCOEFPTR) coef->MCU_buffer[blkn+xindex],
                                                        output_ptr, output_col);
          200                    output_col += compptr->DCT_scaled_size;
                    lwz          r11,0x24(r16)        ; r11,36(r16)
                    add          r17,r17,r11          ; output_col,output_col,r11
          196                 for (xindex = 0; xindex < useful_width; xindex++) {
                    addi         r23,r23,0x1          ; xindex,xindex,1
          201                 }
                    b            0xAC7C
-0009325286            F:0000AC7C ptrace                 \\jpeg\jdcoefct\decompress_onepass+0x140     0.740us
                    cmpw         r23,r20              ; xindex,useful_width
```

Changes Tab

Instruction execution traces may contain multiple runs of a loop. It might be boring to inspect all the loop runs. The **Changes** tab enables you to continue your trace inspection after the last loop run.

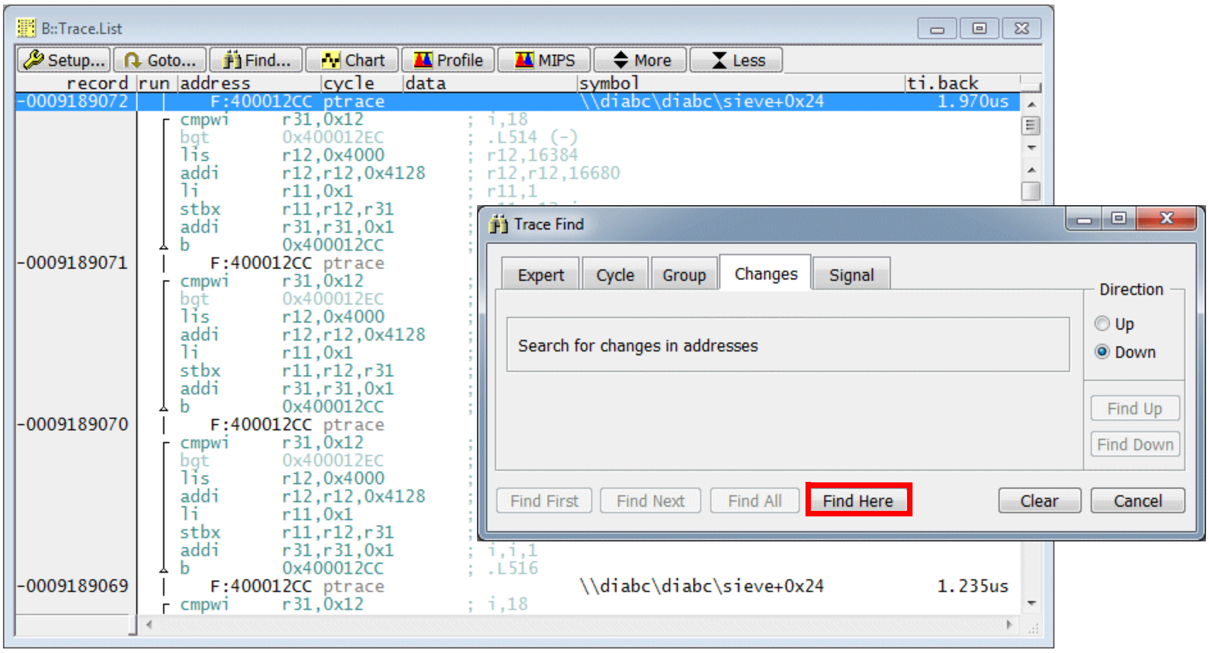


Search for changes in addresses

Direction: Up [✓ Down]

NOTE:	Please be aware that searching via the Changes tab only analyzes **ptrace** cycles.

Example: A trace recording contains multiple runs of a loop.

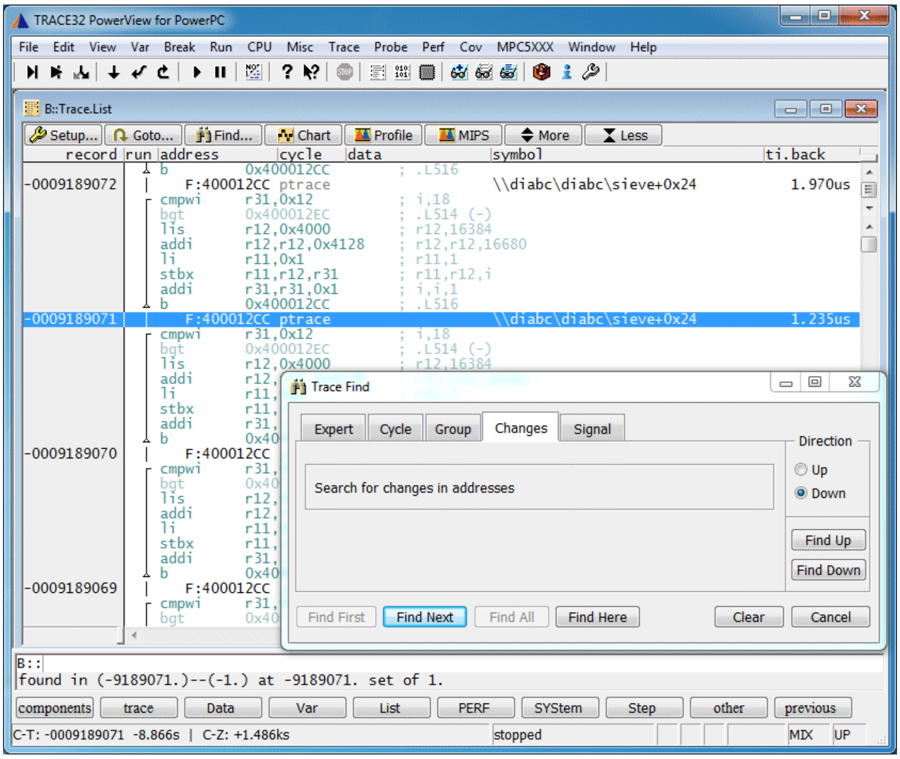

B::Trace.List
Setup... Goto... Find... Chart Profile MIPS More Less
Trace Find
Expert Cycle Group Changes Signal
Find Up
Find Down
Find First Find Next Find All Find Here Clear Cancel

```
   record |run |address        |cycle  |data            |symbol                        |ti.back
-0009189072 |      F:400012CC ptrace                  \\diabc\diabc\sieve+0x24      1.970us
            cmpwi     r31,0x12          ; i,18
            bgt       0x400012EC        ; .L514 (-)
            lis       r12,0x4000        ; r12,16384
            addi      r12,r12,0x4128    ; r12,r12,16680
            li        r11,0x1           ; r11,1
            stbx      r11,r12,r31
            addi      r31,r31,0x1
            b         0x400012CC
-0009189071 |      F:400012CC ptrace
            cmpwi     r31,0x12
            bgt       0x400012EC
            lis       r12,0x4000
            addi      r12,r12,0x4128
            li        r11,0x1
            stbx      r11,r12,r31
            addi      r31,r31,0x1
            b         0x400012CC
-0009189070 |      F:400012CC ptrace
            cmpwi     r31,0x12
            bgt       0x400012EC
            lis       r12,0x4000
            addi      r12,r12,0x4128
            li        r11,0x1
            stbx      r11,r12,r31
            addi      r31,r31,0x1       ; i,i,1
            b         0x400012CC        ; .L516
-0009189069 |      F:400012CC ptrace                  \\diabc\diabc\sieve+0x24      1.235us
            cmpwi     r31,0x12          ; i,18
```

Search for changes in addresses

Direction: Up [✓ Down]

Select the first occurrence of the loop and push the **Find Here** button to prepare the **Changes** procedure:


TRACE32 PowerView for PowerPC
File Edit View Var Break Run CPU Misc Trace Probe Perf Cov MPC5XXX Window Help
B::Trace.List
Setup... Goto... Find... Chart Profile MIPS More Less
Trace Find
Expert Cycle Group Changes Signal
Find Up
Find Down
Find First Find Next Find All Find Here Clear Cancel
components trace Data Var List PERF SYStem Step other previous

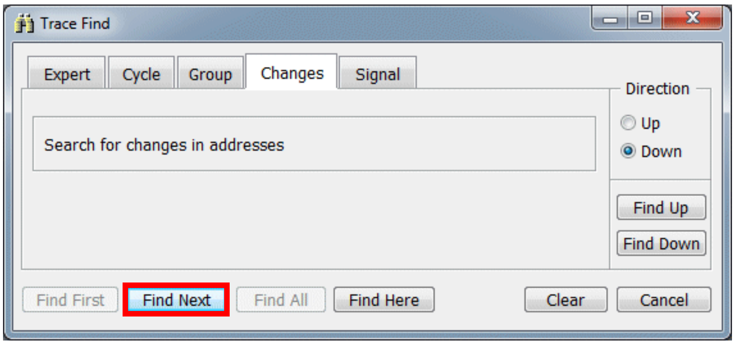
```
   record |run |address        |cycle  |data            |symbol                    |ti.back
            b         0x400012CC        ; .L516
-0009189072 |      F:400012CC ptrace              \\diabc\diabc\sieve+0x24      1.970us
            cmpwi     r31,0x12          ; i,18
            bgt       0x400012EC        ; .L514 (-)
            lis       r12,0x4000        ; r12,16384
            addi      r12,r12,0x4128    ; r12,r12,16680
            li        r11,0x1           ; r11,1
            stbx      r11,r12,r31       ; r11,r12,i
            addi      r31,r31,0x1       ; i,i,1
            b         0x400012CC        ; .L516
-0009189071 |      F:400012CC ptrace              \\diabc\diabc\sieve+0x24      1.235us
            cmpwi     r31,0x12          ; i,18
            bgt       0x400012EC        ; .L514 (-)
            lis       r12,0x4000        ; r12,16384
            addi      r12,
            li        r11,
            stbx      r11,
            addi      r31,
            b         0x40
-0009189070 |      F:400012CC
            cmpwi     r31,
            bgt       0x40
            lis       r12,
            addi      r12,
            li        r11,
            stbx      r11,
            addi      r31,
            b         0x40
-0009189069 |      F:400012CC
            cmpwi     r31,
            bgt       0x40
```

Search for changes in addresses

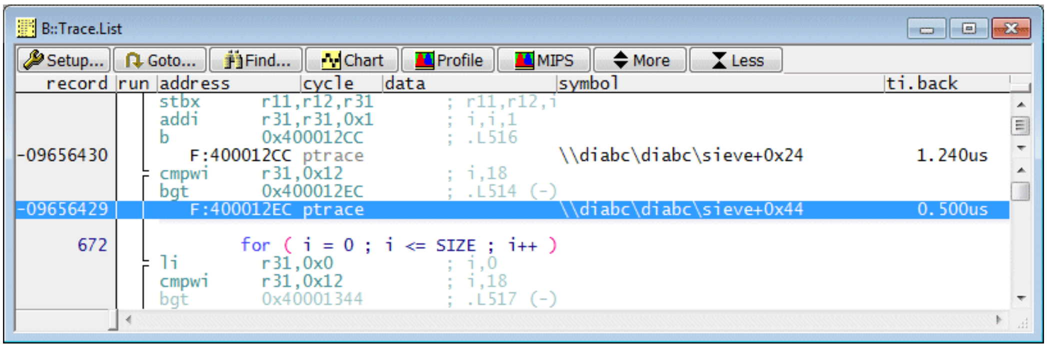
Direction: Up [✓ Down]

```
B::
found in (-9189071.)--(-1.) at -9189071. set of 1.
```

```
C-T: -0009189071  -8.866s | C-Z: +1.486ks        stopped        MIX  UP
```



Search for changes in addresses



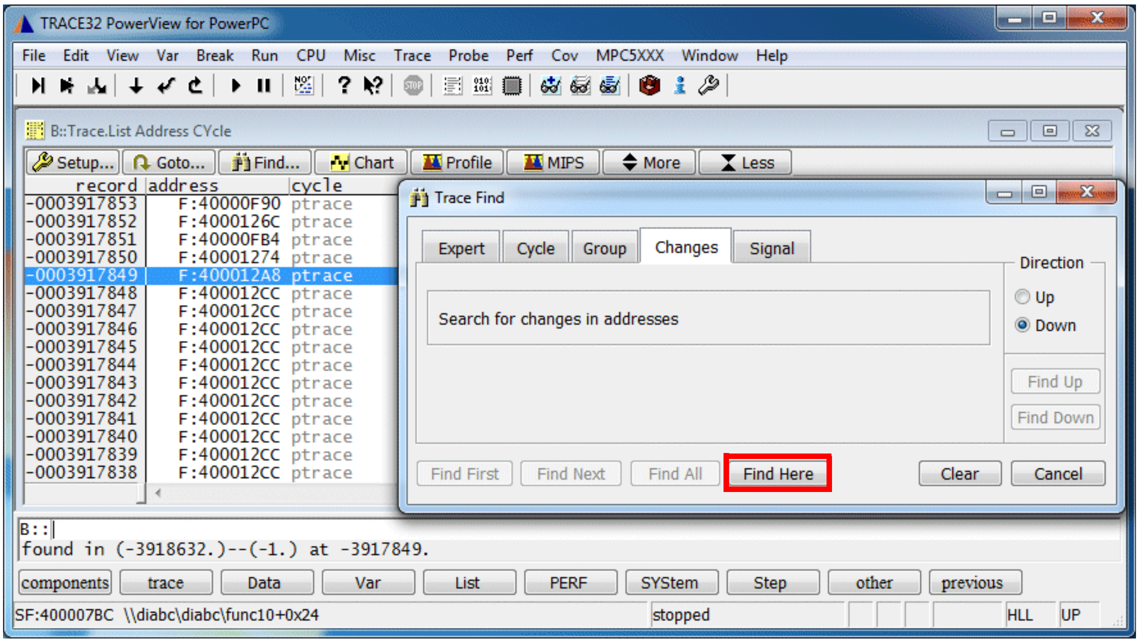
```
   record run address         cycle   data             symbol                       ti.back
             stbx      r11,r12,r31         ; r11,r12,i
             addi      r31,r31,0x1         ; i,i,1
             b         0x400012CC          ; .L516
-09656430       F:400012CC ptrace                       \\diabc\diabc\sieve+0x24          1.240us
             cmpwi     r31,0x12            ; i,18
             bgt       0x400012EC          ; .L514 (-)
-09656429       F:400012EC ptrace                       \\diabc\diabc\sieve+0x44          0.500us

      672           for ( i = 0 ; i <= SIZE ; i++ )
             li        r31,0x0             ; i,0
             cmpwi     r31,0x12            ; i,18
             bgt       0x40001344          ; .L517 (-)
```

Pushing the **Find Next** button (several times) positions the cursor now after the last loop run.

The following example illustrates the **Changes** procedure in detail.

NOTE:	Changes considers only the ptrace cycles.

1. **Select a trace record and start with *Find Here*:**



```
record      address    cycle
-0003917853  F:40000F90 ptrace
-0003917852  F:4000126C ptrace
-0003917851  F:40000FB4 ptrace
-0003917850  F:40001274 ptrace
-0003917849  F:400012A8 ptrace
-0003917848  F:400012CC ptrace
-0003917847  F:400012CC ptrace
-0003917846  F:400012CC ptrace
-0003917845  F:400012CC ptrace
-0003917844  F:400012CC ptrace
-0003917843  F:400012CC ptrace
-0003917842  F:400012CC ptrace
-0003917841  F:400012CC ptrace
-0003917840  F:400012CC ptrace
-0003917839  F:400012CC ptrace
-0003917838  F:400012CC ptrace
```

Search for changes in addresses

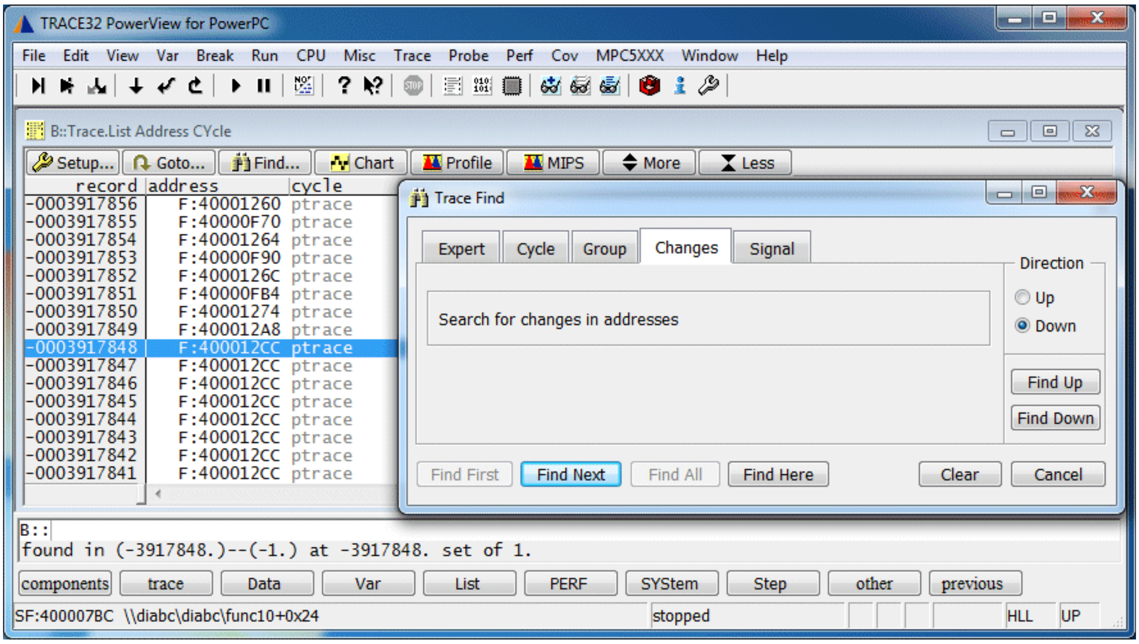
Direction: Up [✓ Down]

```
B::
found in (-3918632.)--(-1.) at -3917849.
```

SF:400007BC \\diabc\diabc\func10+0x24 stopped HLL UP

TRACE32 PowerView for PowerPC
File Edit View Var Break Run CPU Misc Trace Probe Perf Cov MPC5XXX Window Help
B::Trace.List Address CYcle
Setup... Goto... Find... Chart Profile MIPS More Less
Trace Find
Expert Cycle Group Changes Signal
Find Up Find Down
Find First Find Next Find All Find Here Clear Cancel
components trace Data Var List PERF SYStem Step other previous

The cursor is positioned to the next record and the so-called **Changes Address Set** is initialized. The address of the current ptrace cycle is added to the **Changes Address Set**.



```
record      address    cycle
-0003917856  F:40001260 ptrace
-0003917855  F:40000F70 ptrace
-0003917854  F:40001264 ptrace
-0003917853  F:40000F90 ptrace
-0003917852  F:4000126C ptrace
-0003917851  F:40000FB4 ptrace
-0003917850  F:40001274 ptrace
-0003917849  F:400012A8 ptrace
-0003917848  F:400012CC ptrace
-0003917847  F:400012CC ptrace
-0003917846  F:400012CC ptrace
-0003917845  F:400012CC ptrace
-0003917844  F:400012CC ptrace
-0003917843  F:400012CC ptrace
-0003917842  F:400012CC ptrace
-0003917841  F:400012CC ptrace
```

Search for changes in addresses

Direction: Up [✓ Down]

```
B::
found in (-3917848.)--(-1.) at -3917848. set of 1.
```

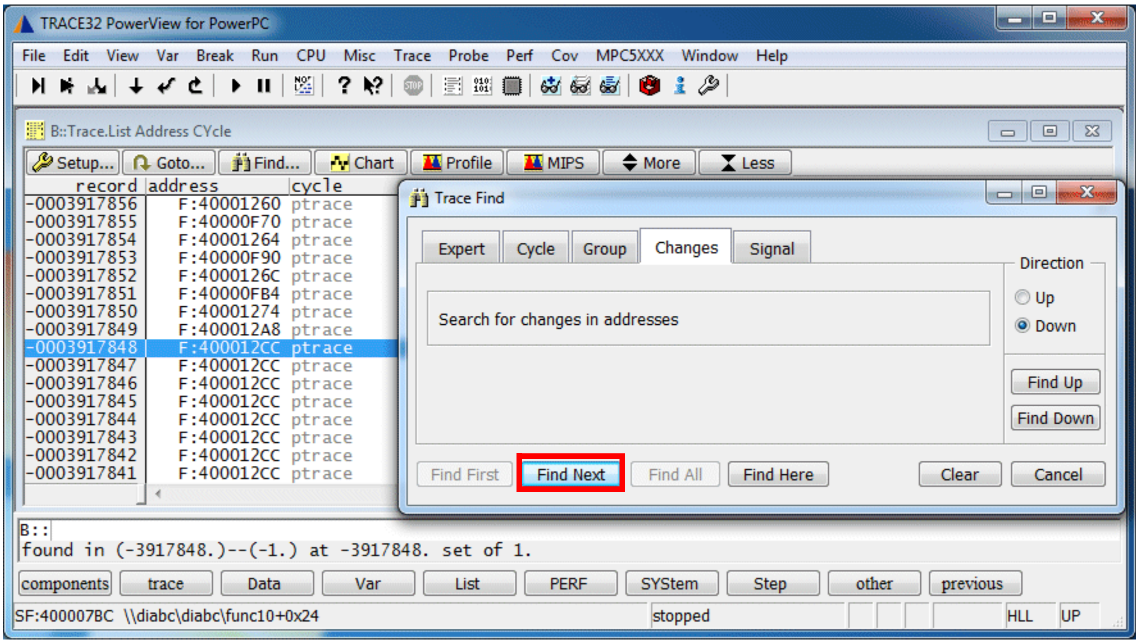
SF:400007BC \\diabc\diabc\func10+0x24 stopped HLL UP

TRACE32 PowerView for PowerPC
File Edit View Var Break Run CPU Misc Trace Probe Perf Cov MPC5XXX Window Help
B::Trace.List Address CYcle
Setup... Goto... Find... Chart Profile MIPS More Less
Trace Find
Expert Cycle Group Changes Signal
Find Up Find Down
Find First Find Next Find All Find Here Clear Cancel
components trace Data Var List PERF SYStem Step other previous

Changes Address Set: 0x400012A8

2. **Continue with *Find Next*.**

Changes Address Set: 0x400012A8


TRACE32 PowerView for PowerPC
File Edit View Var Break Run CPU Misc Trace Probe Perf Cov MPC5XXX Window Help
B::Trace.List Address CYcle
Setup... Goto... Find... Chart Profile MIPS More Less
Trace Find
Expert Cycle Group Changes Signal
Direction
Find Up
Find Down
Find First Find Next Find All Find Here Clear Cancel
components trace Data Var List PERF SYStem Step other previous

```
     record address    cycle
-0003917856   F:40001260 ptrace
-0003917855   F:40000F70 ptrace
-0003917854   F:40001264 ptrace
-0003917853   F:40000F90 ptrace
-0003917852   F:4000126C ptrace
-0003917851   F:40000FB4 ptrace
-0003917850   F:40001274 ptrace
-0003917849   F:400012A8 ptrace
-0003917848   F:400012CC ptrace
-0003917847   F:400012CC ptrace
-0003917846   F:400012CC ptrace
-0003917845   F:400012CC ptrace
-0003917844   F:400012CC ptrace
-0003917843   F:400012CC ptrace
-0003917842   F:400012CC ptrace
-0003917841   F:400012CC ptrace
```

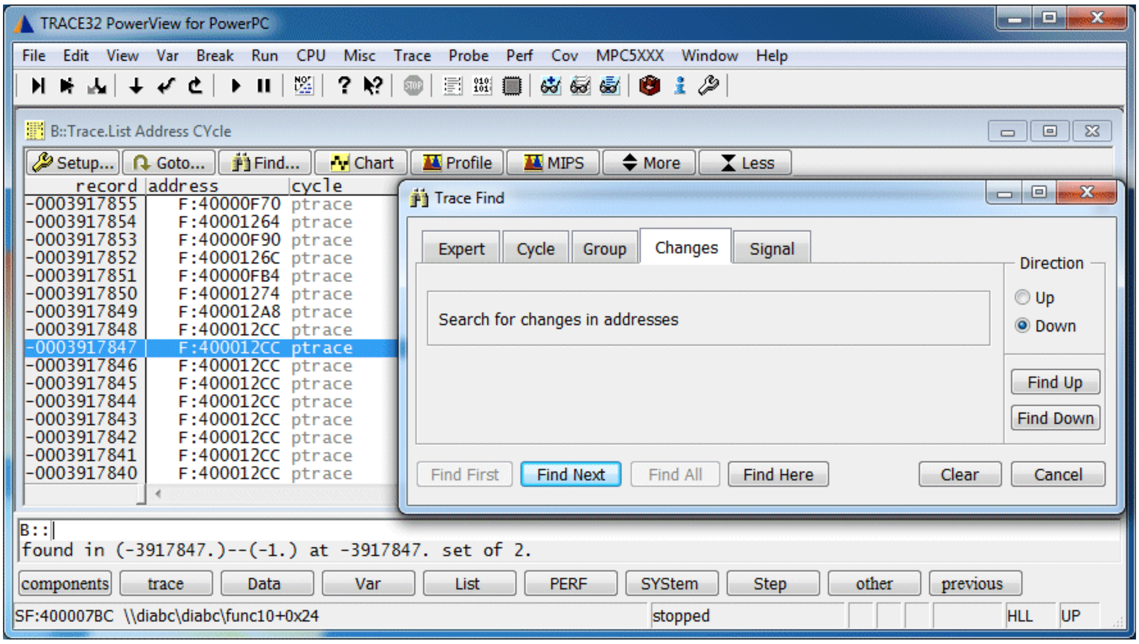
Search for changes in addresses

Direction: Up [✓ Down]

```
B::
found in (-3917848.)--(-1.) at -3917848. set of 1.
```

SF:400007BC \\diabc\diabc\func10+0x24 stopped HLL UP

The cursor is positioned to next ptrace record for which the address is not in **Changes Address Set.** Additionally the address of the current ptrace cycle is added to the **Changes Address Set**.


TRACE32 PowerView for PowerPC
File Edit View Var Break Run CPU Misc Trace Probe Perf Cov MPC5XXX Window Help
B::Trace.List Address CYcle
Setup... Goto... Find... Chart Profile MIPS More Less
Trace Find
Expert Cycle Group Changes Signal
Direction
Find Up
Find Down
Find First Find Next Find All Find Here Clear Cancel
components trace Data Var List PERF SYStem Step other previous

```
     record address    cycle
-0003917855   F:40000F70 ptrace
-0003917854   F:40001264 ptrace
-0003917853   F:40000F90 ptrace
-0003917852   F:4000126C ptrace
-0003917851   F:40000FB4 ptrace
-0003917850   F:40001274 ptrace
-0003917849   F:400012A8 ptrace
-0003917848   F:400012CC ptrace
-0003917847   F:400012CC ptrace
-0003917846   F:400012CC ptrace
-0003917845   F:400012CC ptrace
-0003917844   F:400012CC ptrace
-0003917843   F:400012CC ptrace
-0003917842   F:400012CC ptrace
-0003917841   F:400012CC ptrace
-0003917840   F:400012CC ptrace
```

Search for changes in addresses

Direction: Up [✓ Down]

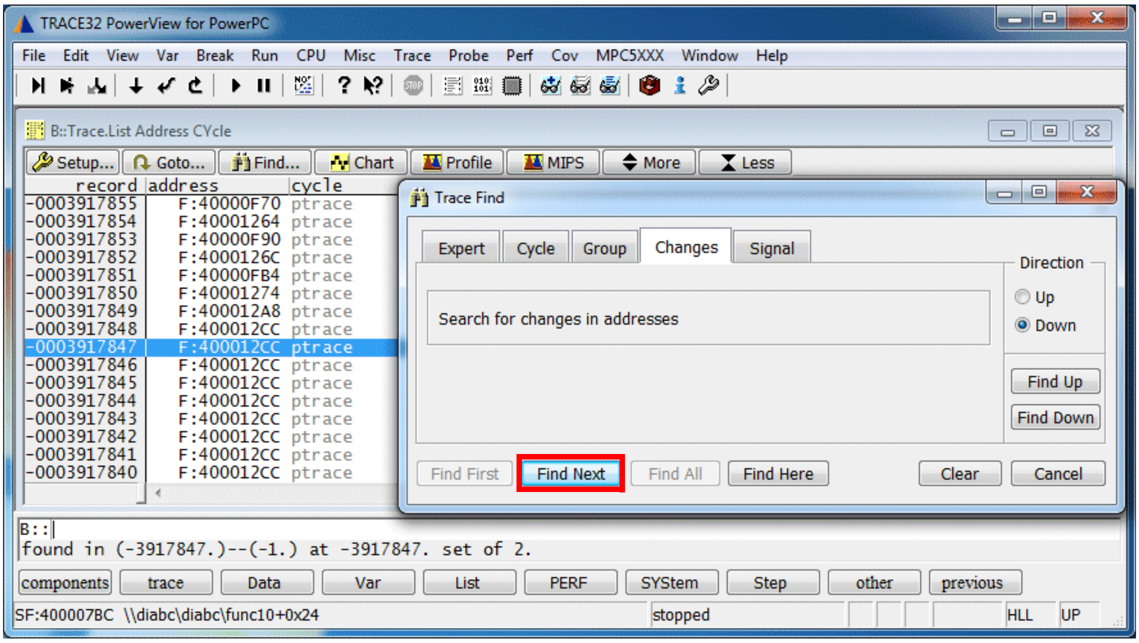
```
B::
found in (-3917847.)--(-1.) at -3917847. set of 2.
```

SF:400007BC \\diabc\diabc\func10+0x24 stopped HLL UP

Changes Address Set: 0x400012A8, 0x400012CC

3. **Continue with *Find Next*.**

Changes Address Set: 0x400012A8, 0x400012CC

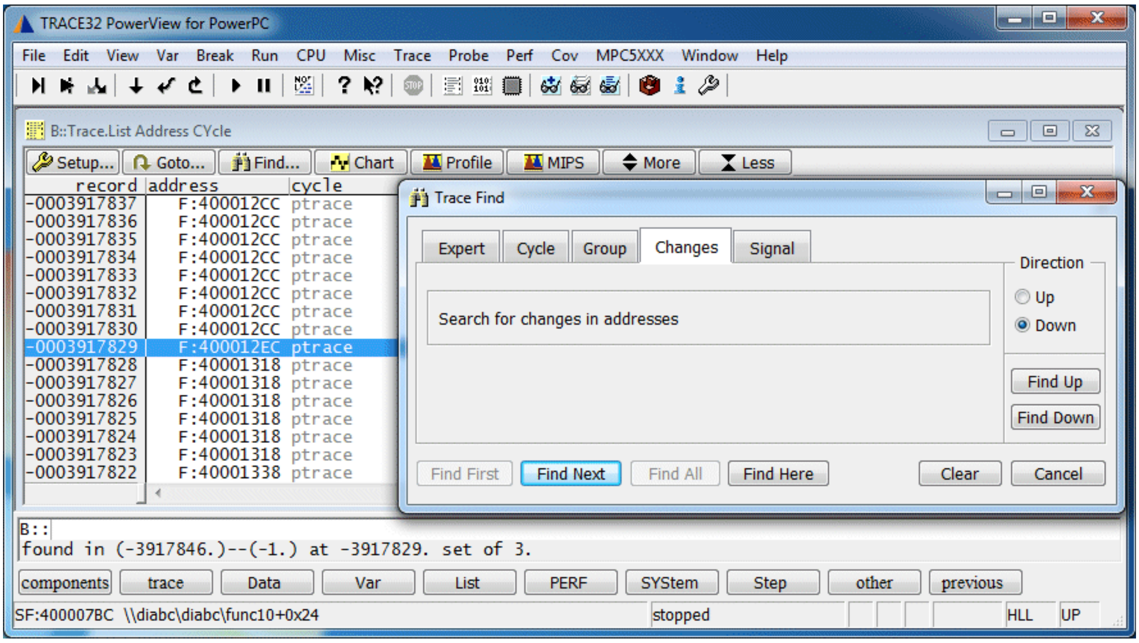

TRACE32 PowerView for PowerPC
File Edit View Var Break Run CPU Misc Trace Probe Perf Cov MPC5XXX Window Help
B::Trace.List Address CYcle
Setup... Goto... Find... Chart Profile MIPS More Less
Trace Find
Expert Cycle Group Changes Signal
Direction
Up
Down
Find Up
Find Down
Find First Find Next Find All Find Here Clear Cancel
components trace Data Var List PERF SYStem Step other previous
stopped HLL UP

```
   record address     cycle
-0003917855   F:40000F70 ptrace
-0003917854   F:40001264 ptrace
-0003917853   F:40000F90 ptrace
-0003917852   F:4000126C ptrace
-0003917851   F:40000FB4 ptrace
-0003917850   F:40001274 ptrace
-0003917849   F:400012A8 ptrace
-0003917848   F:400012CC ptrace
-0003917847   F:400012CC ptrace
-0003917846   F:400012CC ptrace
-0003917845   F:400012CC ptrace
-0003917844   F:400012CC ptrace
-0003917843   F:400012CC ptrace
-0003917842   F:400012CC ptrace
-0003917841   F:400012CC ptrace
-0003917840   F:400012CC ptrace
```

Search for changes in addresses

```
B::
found in (-3917847.)--(-1.) at -3917847. set of 2.
SF:400007BC  \\diabc\diabc\func10+0x24
```

The cursor is positioned to next ptrace record for which the address is not in **Changes Address Set.** Additionally the address of the current ptrace cycle is added to the **Changes Address Set**.


TRACE32 PowerView for PowerPC
File Edit View Var Break Run CPU Misc Trace Probe Perf Cov MPC5XXX Window Help
B::Trace.List Address CYcle
Setup... Goto... Find... Chart Profile MIPS More Less
Trace Find
Expert Cycle Group Changes Signal
Direction
Up
Down
Find Up
Find Down
Find First Find Next Find All Find Here Clear Cancel
components trace Data Var List PERF SYStem Step other previous
stopped HLL UP

```
   record address     cycle
-0003917837   F:400012CC ptrace
-0003917836   F:400012CC ptrace
-0003917835   F:400012CC ptrace
-0003917834   F:400012CC ptrace
-0003917833   F:400012CC ptrace
-0003917832   F:400012CC ptrace
-0003917831   F:400012CC ptrace
-0003917830   F:400012CC ptrace
-0003917829   F:400012EC ptrace
-0003917828   F:40001318 ptrace
-0003917827   F:40001318 ptrace
-0003917826   F:40001318 ptrace
-0003917825   F:40001318 ptrace
-0003917824   F:40001318 ptrace
-0003917823   F:40001318 ptrace
-0003917822   F:40001338 ptrace
```

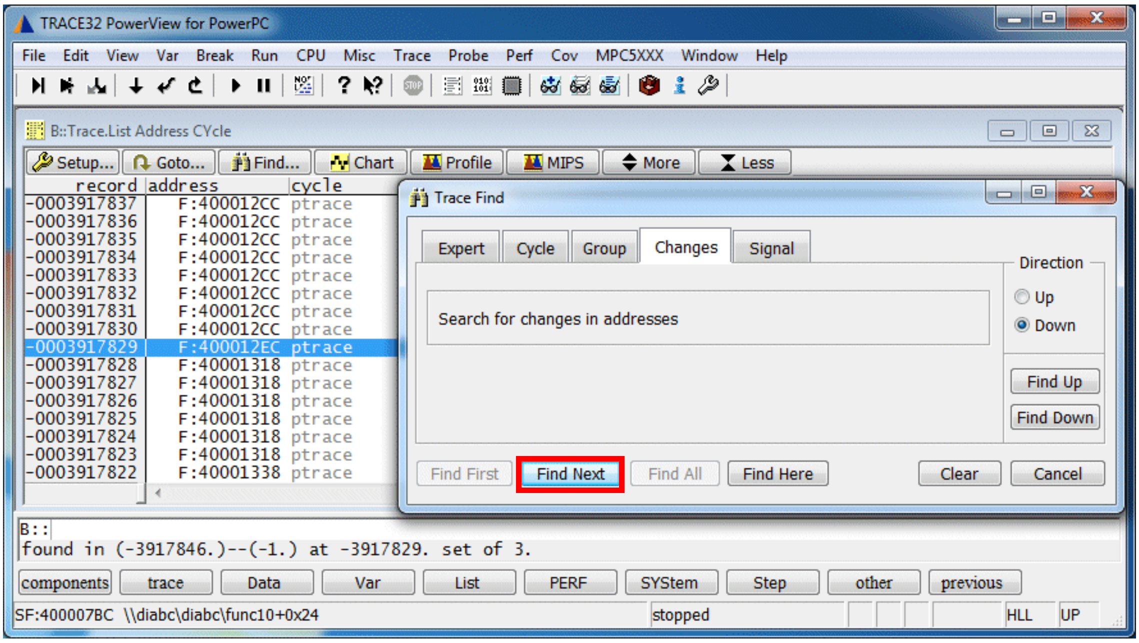
Search for changes in addresses

```
B::
found in (-3917846.)--(-1.) at -3917829. set of 3.
SF:400007BC  \\diabc\diabc\func10+0x24
```

Changes Address Set: 0x400012A8, 0x400012CC

4. **Continue with *Find Next*.**

Changes Address Set: 0x400012A8, 0x400012CC


TRACE32 PowerView for PowerPC
File Edit View Var Break Run CPU Misc Trace Probe Perf Cov MPC5XXX Window Help
B::Trace.List Address CYcle
Setup... Goto... Find... Chart Profile MIPS More Less
Trace Find
Expert Cycle Group Changes Signal
Direction
Up
Down
Find Up
Find Down
Find First Find Next Find All Find Here Clear Cancel
components trace Data Var List PERF SYStem Step other previous

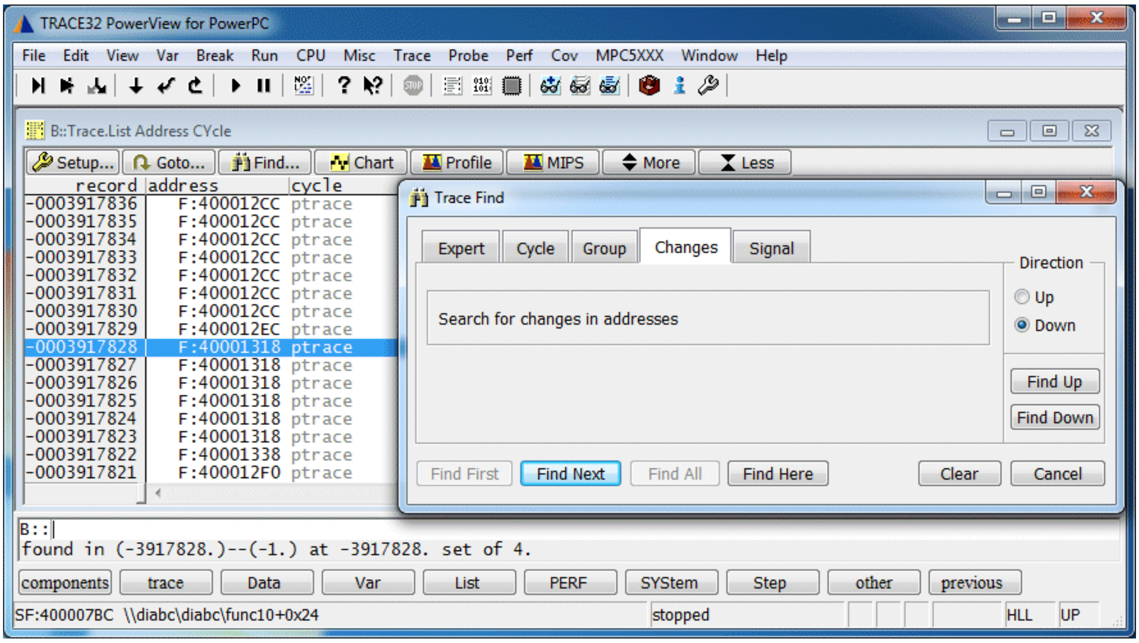
```
    record address      cycle
-0003917837    F:400012CC ptrace
-0003917836    F:400012CC ptrace
-0003917835    F:400012CC ptrace
-0003917834    F:400012CC ptrace
-0003917833    F:400012CC ptrace
-0003917832    F:400012CC ptrace
-0003917831    F:400012CC ptrace
-0003917830    F:400012CC ptrace
-0003917829    F:400012EC ptrace
-0003917828    F:40001318 ptrace
-0003917827    F:40001318 ptrace
-0003917826    F:40001318 ptrace
-0003917825    F:40001318 ptrace
-0003917824    F:40001318 ptrace
-0003917823    F:40001318 ptrace
-0003917822    F:40001338 ptrace
```

Search for changes in addresses

```
B::
found in (-3917846.)--(-1.) at -3917829. set of 3.
```

```
SF:400007BC  \\diabc\diabc\func10+0x24     stopped     HLL   UP
```

The cursor is positioned to next ptrace record for which the address is not in **Changes Address Set.** Additionally the address of the current ptrace cycle is added to the **Changes Address Set**.


TRACE32 PowerView for PowerPC
File Edit View Var Break Run CPU Misc Trace Probe Perf Cov MPC5XXX Window Help
B::Trace.List Address CYcle
Setup... Goto... Find... Chart Profile MIPS More Less
Trace Find
Expert Cycle Group Changes Signal
Direction
Up
Down
Find Up
Find Down
Find First Find Next Find All Find Here Clear Cancel
components trace Data Var List PERF SYStem Step other previous

```
    record address      cycle
-0003917836    F:400012CC ptrace
-0003917835    F:400012CC ptrace
-0003917834    F:400012CC ptrace
-0003917833    F:400012CC ptrace
-0003917832    F:400012CC ptrace
-0003917831    F:400012CC ptrace
-0003917830    F:400012CC ptrace
-0003917829    F:400012EC ptrace
-0003917828    F:40001318 ptrace
-0003917827    F:40001318 ptrace
-0003917826    F:40001318 ptrace
-0003917825    F:40001318 ptrace
-0003917824    F:40001318 ptrace
-0003917823    F:40001318 ptrace
-0003917822    F:40001338 ptrace
-0003917821    F:400012F0 ptrace
```

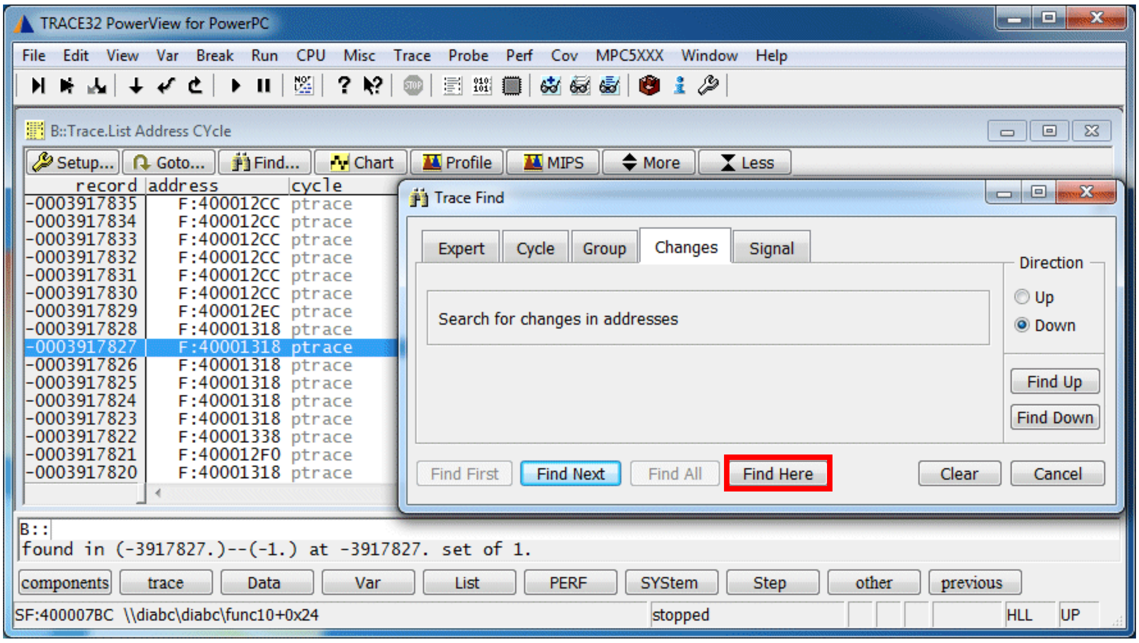
Search for changes in addresses

```
B::
found in (-3917828.)--(-1.) at -3917828. set of 4.
```

```
SF:400007BC  \\diabc\diabc\func10+0x24     stopped     HLL   UP
```

Changes Address Set: 0x400012A8, 0x400012CC, 0x400012EC

Please be aware that ***Find Here*** starts a new **Changes Address Set**.



```
record      address      cycle
-0003917835   F:400012CC ptrace
-0003917834   F:400012CC ptrace
-0003917833   F:400012CC ptrace
-0003917832   F:400012CC ptrace
-0003917831   F:400012CC ptrace
-0003917830   F:400012CC ptrace
-0003917829   F:400012EC ptrace
-0003917828   F:40001318 ptrace
-0003917827   F:40001318 ptrace
-0003917826   F:40001318 ptrace
-0003917825   F:40001318 ptrace
-0003917824   F:40001318 ptrace
-0003917823   F:40001318 ptrace
-0003917822   F:40001338 ptrace
-0003917821   F:400012F0 ptrace
-0003917820   F:40001318 ptrace
```

Trace Find — Changes: Search for changes in addresses

Direction: Up [✓ Down]

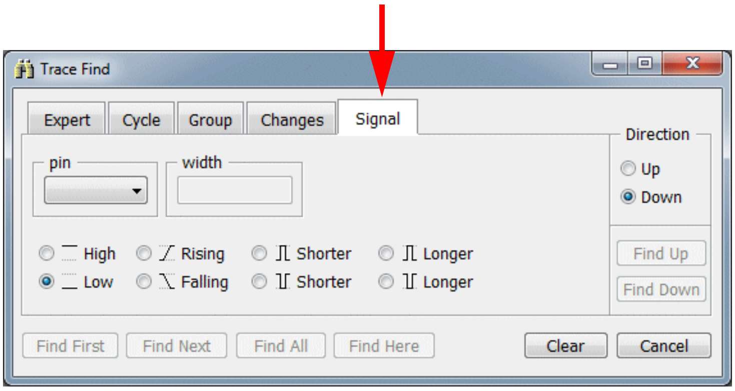
```
B::
found in (-3917827.)--(-1.) at -3917827. set of 1.
```

SF:400007BC \\diabc\diabc\func10+0x24 stopped

TRACE32 PowerView for PowerPC
File Edit View Var Break Run CPU Misc Trace Probe Perf Cov MPC5XXX Window Help
B::Trace.List Address CYcle
Setup... Goto... Find... Chart Profile MIPS More Less
Trace Find
Expert Cycle Group Changes Signal
Find Up
Find Down
Find First Find Next Find All Find Here Clear Cancel
components trace Data Var List PERF SYStem Step other previous
HLL UP

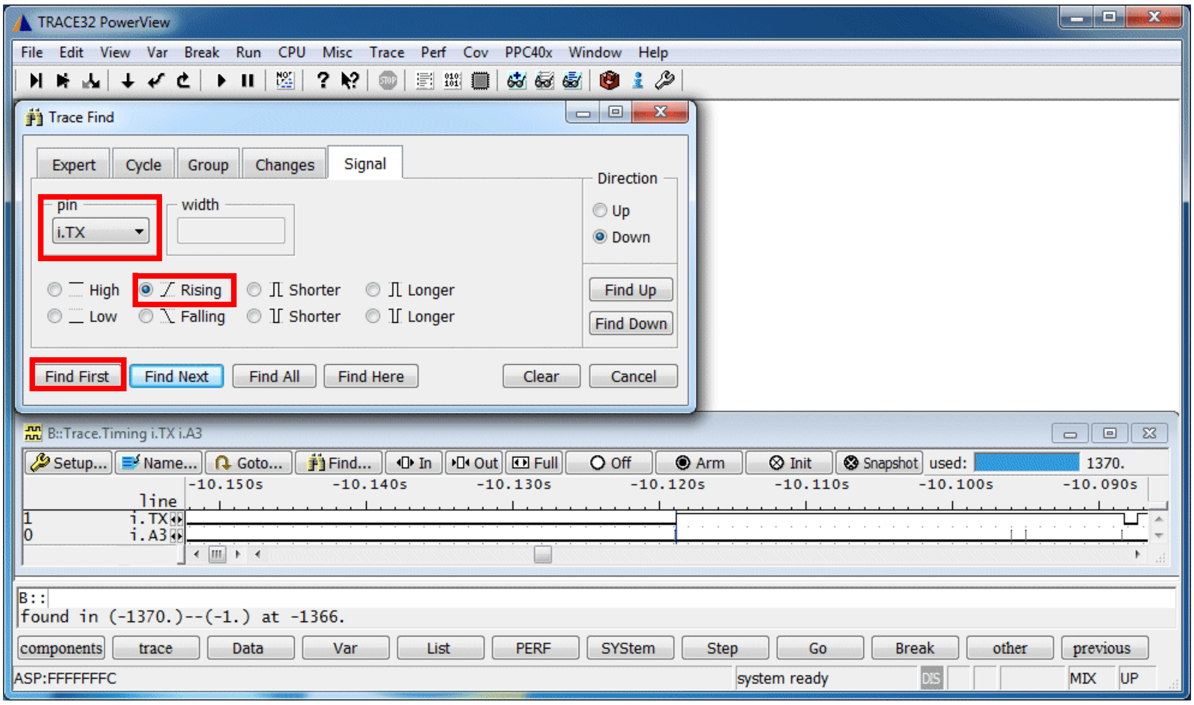
Signal Tab

The **Trace.Timing** command is used to display signal recording as performed e.g. by a TRACE32 logic analyzer. The Signal tab of the Trace Find dialog allows to search for specified signal levels, rising edges etc.



Signal Level/Edges

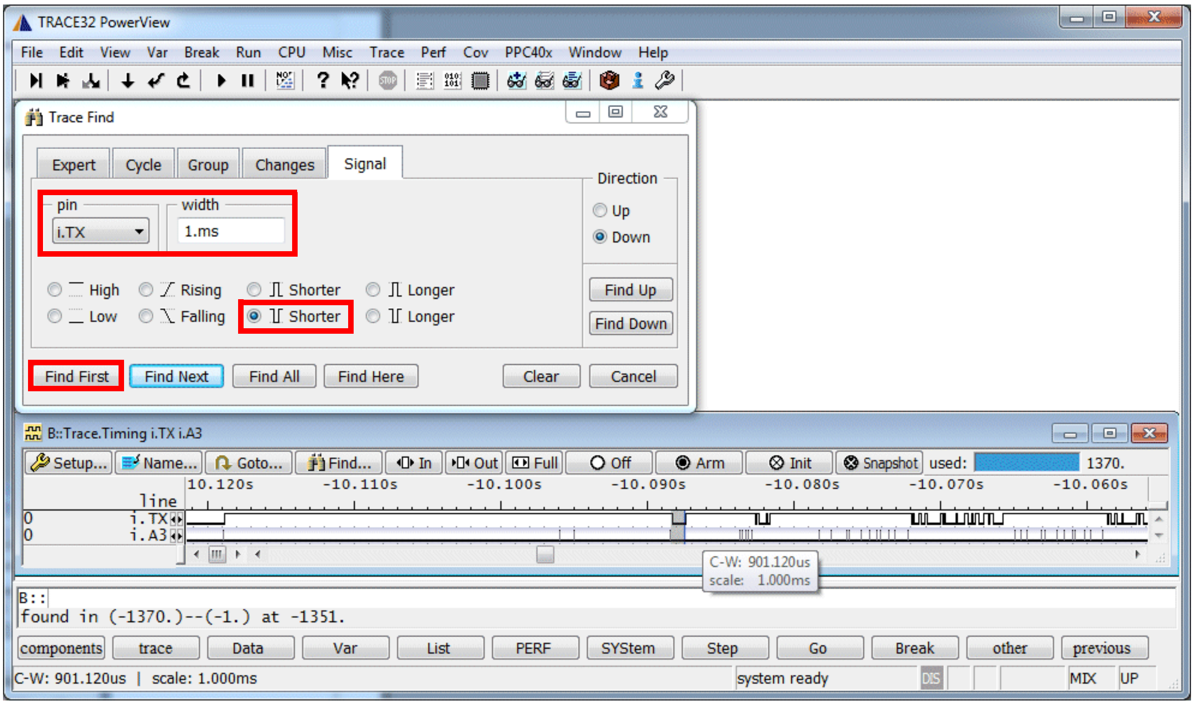
Example: Find the first rising edge of signal i.TX.



```
B::
found in (-1370.)--(-1.) at -1366.
```

Signal Level of Specified Width

Example: Find point in recording where i.TX is low for less than 1.ms.


TRACE32 PowerView
File Edit View Var Break Run CPU Misc Trace Perf Cov PPC40x Window Help
Trace Find
Expert Cycle Group Changes Signal
Direction
Up
Down
Find Up
Find Down
High Rising Shorter Longer
Low Falling Shorter Longer
Find First Find Next Find All Find Here Clear Cancel
B::Trace.Timing i.TX i.A3
Setup... Name... Goto... Find... In Out Full Off Arm Init Snapshot used:
components trace Data Var List PERF SYStem Step Go Break other previous

pin: i.TX
width: 1.ms
[✓ Down]
[✓ Shorter]

```
10.120s   -10.110s   -10.100s   -10.090s   -10.080s   -10.070s   -10.060s
     line
0    i.TX
0    i.A3
```

C-W: 901.120us
scale: 1.000ms

```
B::
found in (-1370.)--(-1.) at -1351.
```

used: 1370.

C-W: 901.120us | scale: 1.000ms   system ready   MIX UP

Trace Find Commands

Overview

TRACE32 PowerView provides the following commands to find a specified event in the trace recording.

; find specified search event in the trace recording
Trace.Find [*<record_number>* | *<record_range>*] *<item>* … [**/***<options>*]

; repeat last search event
Trace.Find

; find all occurrences of the specified search event
Trace.FindAll [*<record_number>* | *<record_range>*] *<items>* … [**/***<options>*]

; find change in program execution sequence
Trace.FindChange [*<record_number>* | *<record_range>*] [**/***<options>*]

Combining Search Items

Several *<item>* can be combined to form a search event. They all have to be true.

```
; Specify search event
; Address equal 0x4000412B AND CYcle equal Write AND Data equal 0x0
Trace.Find Address 0x4000412B CYcle Write Data 0x0
```

The keyword **OR** allows to specify a list of search events.

```
Trace.FindAll Address 0x4000412B CYcle Write Data 0x0 OR \
              Address 0x4000412B CYcle Write Data 0x1
```

```
; Start searching at first trace record down to last trace record
Trace.Find Address sieve
; Repeat searching
; Start searching at (last record found + 1) down to the last record
Trace.Find

; Start searching at last trace record up to first trace record
Trace.Find Address sieve /Back
; Repeat searching
; Start searching at (last record found + 1) up to first trace record
Trace.Find

; Search over the complete trace
Trace.FindAll Address sieve
```

```
; Start searching at the specified trace record down to last trace record
Trace.Find Address sieve -99928.
; Repeat searching
; Start searching at (last record found + 1) down to the last record
Trace.Find

; Start searching at the specified trace record up to first trace record
Trace.Find Address sieve -99928. /Back
; Repeat searching
; Start searching at (last record found - 1) up to first trace record
Trace.Find
```

```
; Start searching at first trace record of the specified record range
Trace.Find Address sieve (-99134.)--(-5666.)
; Repeat searching
; Start searching at (last record found + 1) down to the last record
Trace.Find

; Start searching at last trace record of the specified record range
Trace.Find Address sieve (-99134.)--(-5666.) /Back
; Repeat searching
; Start searching at (last record found - 1) up to first trace record
Trace.Find

; Search only over the specified range
Trace.FindAll Address (-99134.)--(-5666.) sieve
```

Trace Item with Specified Value

Address

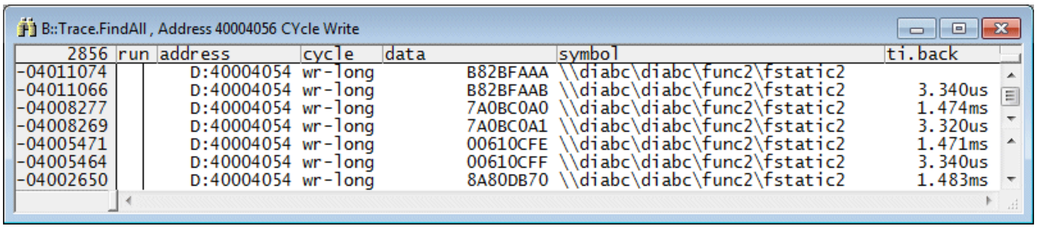
```
; find matching address and data,
; start search at the beginning of the trace recording
Trace.Find Address 0x100--0x200 Data.B 0x55
; find next match
Trace.Find

; find any instruction of the function sieve,
; start search at the end of the trace recording
Trace.Find Address Var.RANGE(sieve) /Back

; find ptrace cycle which contains address 0x40000B7C
Trace.Find FAddress 0x40000B7C
```

Address.Match

Address allows to search for the specified address. While **Address.MATCH** restricts the search exactly to the specified address (address column matches the specified address).

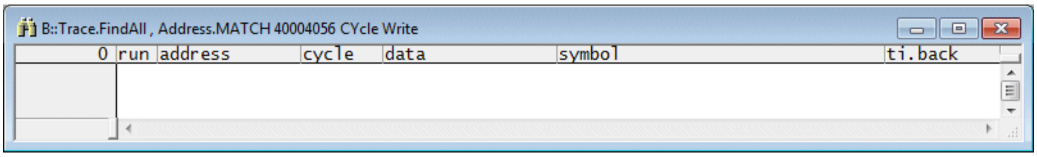
Example for single address

```
Trace.FindAll , Address 0x40004056 CYcle Write
```



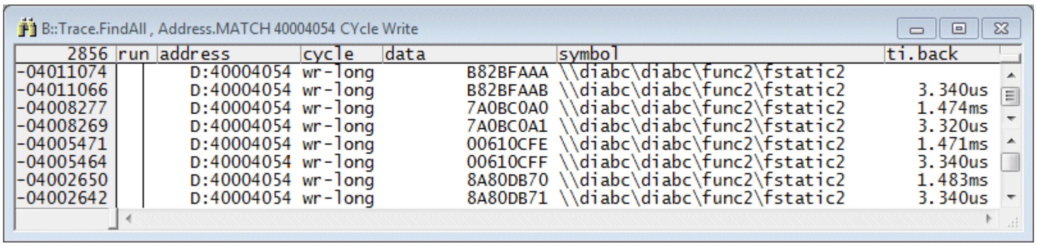
```
B::Trace.FindAll , Address 40004056 CYcle Write
       2856 run address       cycle   data            symbol                        ti.back
-04011074       D:40004054 wr-long         B82BFAAA \\diabc\diabc\func2\fstatic2
-04011066       D:40004054 wr-long         B82BFAAB \\diabc\diabc\func2\fstatic2     3.340us
-04008277       D:40004054 wr-long         7A0BC0A0 \\diabc\diabc\func2\fstatic2     1.474ms
-04008269       D:40004054 wr-long         7A0BC0A1 \\diabc\diabc\func2\fstatic2     3.320us
-04005471       D:40004054 wr-long         00610CFE \\diabc\diabc\func2\fstatic2     1.471ms
-04005464       D:40004054 wr-long         00610CFF \\diabc\diabc\func2\fstatic2     3.340us
-04002650       D:40004054 wr-long         8A80DB70 \\diabc\diabc\func2\fstatic2     1.483ms
```

```
Trace.FindAll , Address.MATCH 0x40004056 CYcle Write
```



```
B::Trace.FindAll , Address.MATCH 40004056 CYcle Write
          0 run address       cycle   data            symbol                        ti.back
```

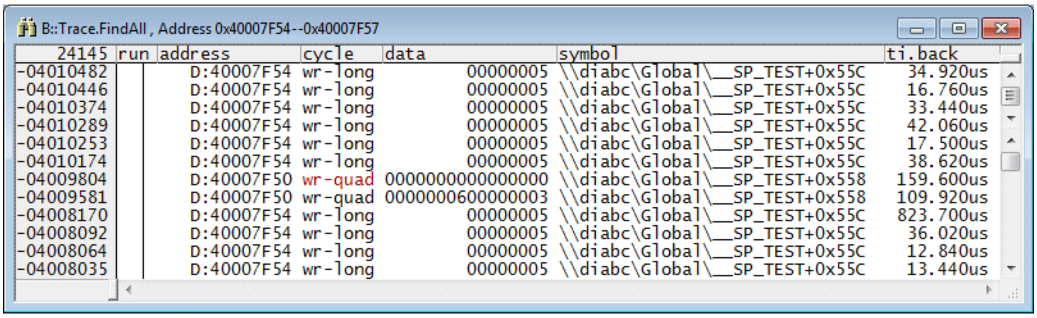
```
Trace.FindAll , Address.MATCH 0x40004054 CYcle Write
```



```
B::Trace.FindAll , Address.MATCH 40004054 CYcle Write
       2856 run address       cycle   data            symbol                        ti.back
-04011074       D:40004054 wr-long         B82BFAAA \\diabc\diabc\func2\fstatic2
-04011066       D:40004054 wr-long         B82BFAAB \\diabc\diabc\func2\fstatic2     3.340us
-04008277       D:40004054 wr-long         7A0BC0A0 \\diabc\diabc\func2\fstatic2     1.474ms
-04008269       D:40004054 wr-long         7A0BC0A1 \\diabc\diabc\func2\fstatic2     3.320us
-04005471       D:40004054 wr-long         00610CFE \\diabc\diabc\func2\fstatic2     1.471ms
-04005464       D:40004054 wr-long         00610CFF \\diabc\diabc\func2\fstatic2     3.340us
-04002650       D:40004054 wr-long         8A80DB70 \\diabc\diabc\func2\fstatic2     1.483ms
-04002642       D:40004054 wr-long         8A80DB71 \\diabc\diabc\func2\fstatic2     3.340us
```

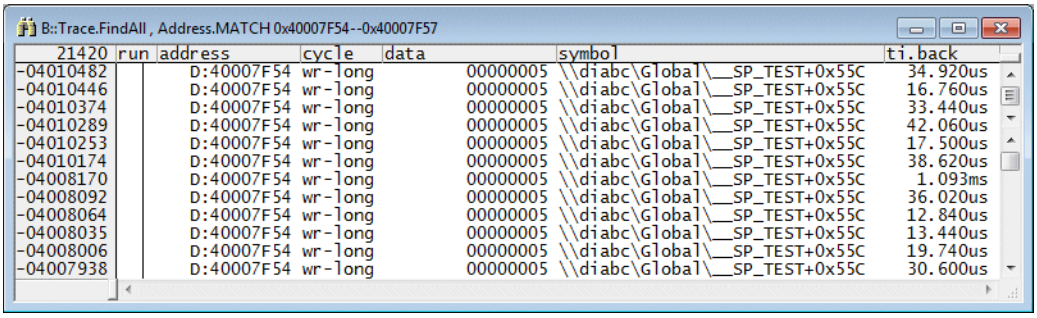
Example for address range

```
Trace.FindAll , Address 0x40007F54--0x40007F57
```



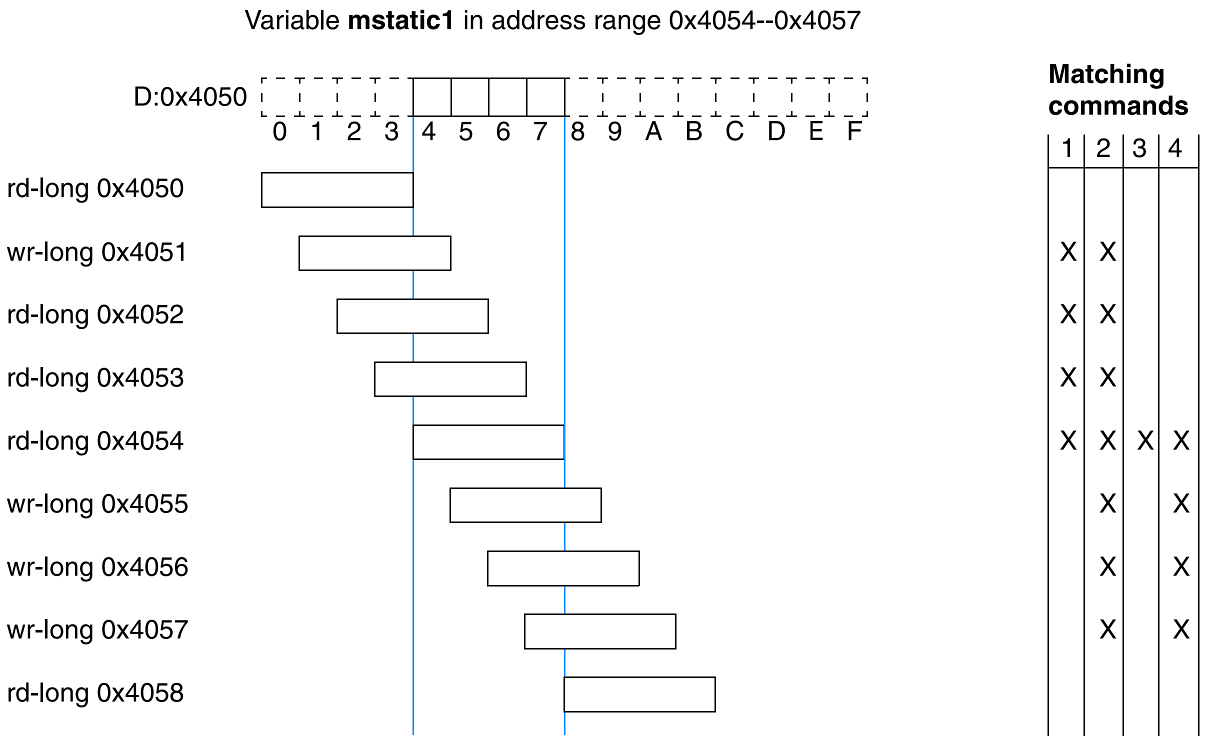
```
B::Trace.FindAll , Address 0x40007F54--0x40007F57
   24145 run address     cycle   data             symbol                        ti.back
-04010482       D:40007F54 wr-long         00000005 \\diabc\Global\__SP_TEST+0x55C   34.920us
-04010446       D:40007F54 wr-long         00000005 \\diabc\Global\__SP_TEST+0x55C   16.760us
-04010374       D:40007F54 wr-long         00000005 \\diabc\Global\__SP_TEST+0x55C   33.440us
-04010289       D:40007F54 wr-long         00000005 \\diabc\Global\__SP_TEST+0x55C   42.060us
-04010253       D:40007F54 wr-long         00000005 \\diabc\Global\__SP_TEST+0x55C   17.500us
-04010174       D:40007F54 wr-long         00000005 \\diabc\Global\__SP_TEST+0x55C   38.620us
-04009804       D:40007F50 wr-quad 0000000000000000 \\diabc\Global\__SP_TEST+0x558  159.600us
-04009581       D:40007F50 wr-quad 0000000600000003 \\diabc\Global\__SP_TEST+0x558  109.920us
-04008170       D:40007F54 wr-long         00000005 \\diabc\Global\__SP_TEST+0x55C  823.700us
-04008092       D:40007F54 wr-long         00000005 \\diabc\Global\__SP_TEST+0x55C   36.020us
-04008064       D:40007F54 wr-long         00000005 \\diabc\Global\__SP_TEST+0x55C   12.840us
-04008035       D:40007F54 wr-long         00000005 \\diabc\Global\__SP_TEST+0x55C   13.440us
```

```
Trace.FindAll , Address.MATCH 0x40007F54--0x40007F57
```



```
B::Trace.FindAll , Address.MATCH 0x40007F54--0x40007F57
   21420 run address     cycle   data             symbol                        ti.back
-04010482       D:40007F54 wr-long         00000005 \\diabc\Global\__SP_TEST+0x55C   34.920us
-04010446       D:40007F54 wr-long         00000005 \\diabc\Global\__SP_TEST+0x55C   16.760us
-04010374       D:40007F54 wr-long         00000005 \\diabc\Global\__SP_TEST+0x55C   33.440us
-04010289       D:40007F54 wr-long         00000005 \\diabc\Global\__SP_TEST+0x55C   42.060us
-04010253       D:40007F54 wr-long         00000005 \\diabc\Global\__SP_TEST+0x55C   17.500us
-04010174       D:40007F54 wr-long         00000005 \\diabc\Global\__SP_TEST+0x55C   38.620us
-04008170       D:40007F54 wr-long         00000005 \\diabc\Global\__SP_TEST+0x55C    1.093ms
-04008092       D:40007F54 wr-long         00000005 \\diabc\Global\__SP_TEST+0x55C   36.020us
-04008064       D:40007F54 wr-long         00000005 \\diabc\Global\__SP_TEST+0x55C   12.840us
-04008035       D:40007F54 wr-long         00000005 \\diabc\Global\__SP_TEST+0x55C   13.440us
-04008006       D:40007F54 wr-long         00000005 \\diabc\Global\__SP_TEST+0x55C   19.740us
-04007938       D:40007F54 wr-long         00000005 \\diabc\Global\__SP_TEST+0x55C   30.600us
```

The graphic below summarizes the behavior of the **Address/Address.MATCH** search item.

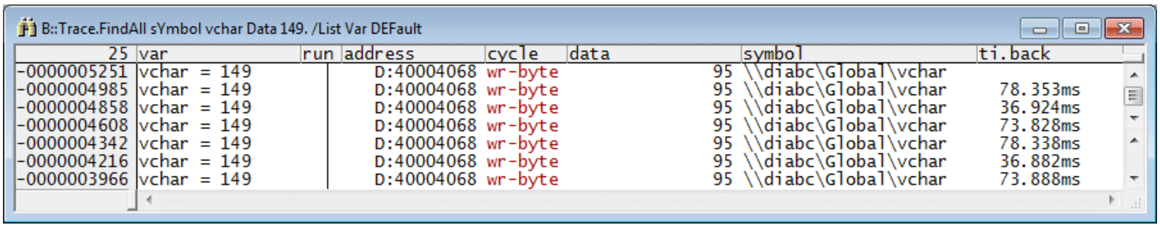


	1	2	3	4
rd-long 0x4050				
wr-long 0x4051	X	X		
rd-long 0x4052	X	X		
rd-long 0x4053	X	X		
rd-long 0x4054	X	X	X	X
wr-long 0x4055		X		X
wr-long 0x4056		X		X
wr-long 0x4057		X		X
rd-long 0x4058				

```
; command 1
Trace.FindAll , Address mstatic1

; command 2
Trace.FindAll , Address Var.RANGE(mstatic1)

; command 3
Trace.FindAll , Address.MATCH mstatic1

; command 4
Trace.FindAll , Address.MATCH Var.RANGE(mstatic1)
```

```
; find specified data value in record range (-1000.)--(-700.)
; start search at the first record of the specified address range
Trace.Find (-1000.)--(-700.) Data 0x100

; find specified data,
; start search at the beginning of the trace recording
Trace.Find Data 0x0--0xAa
```

```
; find all trace records with symbol address vchar and data value 149.
; and display the result is a listing that shows the variable access
; and the default trace display
Trace.FindAll sYmbol vchar Data 149. /List Var DEFault
```


B::Trace.FindAll sYmbol vchar Data 149. /List Var DEFault

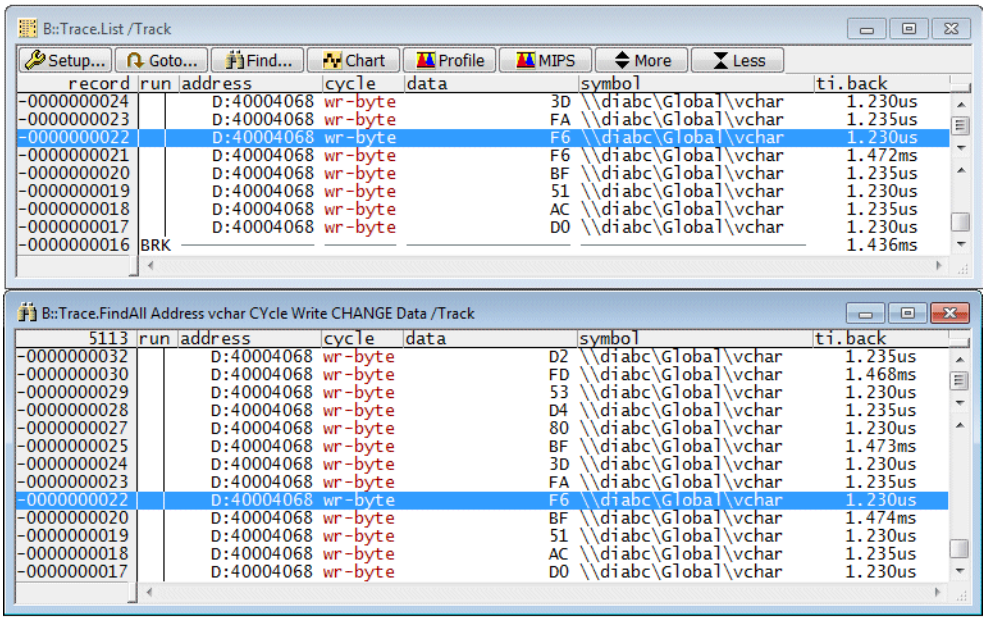
```
           25 var               run address          cycle    data              symbol                   ti.back
-0000005251 vchar = 149             D:40004068 wr-byte                    95 \\diabc\Global\vchar
-0000004985 vchar = 149             D:40004068 wr-byte                    95 \\diabc\Global\vchar      78.353ms
-0000004858 vchar = 149             D:40004068 wr-byte                    95 \\diabc\Global\vchar      36.924ms
-0000004608 vchar = 149             D:40004068 wr-byte                    95 \\diabc\Global\vchar      73.828ms
-0000004342 vchar = 149             D:40004068 wr-byte                    95 \\diabc\Global\vchar      78.338ms
-0000004216 vchar = 149             D:40004068 wr-byte                    95 \\diabc\Global\vchar      36.882ms
-0000003966 vchar = 149             D:40004068 wr-byte                    95 \\diabc\Global\vchar      73.888ms
```

Further examples:

```
; find all trace records with symbol address vchar and with one of the
; specified data values
Trace.FindAll sYmbol vchar Data 149.||191.||210. /List Var DEFault

; find all trace records with symbol address vchar and a data value that
; is not 0.
Trace.FindAll sYmbol vchar Data !0. /List Var DEFault

; find all trace records with symbol address vchar and a data value that
; is not 0., not 100. and not 200.
Trace.FindAll sYmbol vchar Data !(0.||100.||200.) /List Var DEFault

; find all trace record with symbol address vchar and a data value
; between 100. and 200.
Trace.FindAll sYmbol vchar Data (100.--220.) /List Var DEFault

; find all trace record with symbol address vchar and a data value
; smaller than 100. and larger than 200.
Trace.FindAll sYmbol vchar Data !(100.--220.) /List Var DEFault

Trace.FindAll sYmbol vchar Data (120.--150.)||(200.--220.) \
/List Var DEFault
```

```
Trace.FindAll sYmbol vchar Data !((120.--150.)||(200.--220.)) \
/List Var DEFault

; find all trace record with symbol address vchar and a data value
; that is 0x9 in the first 4 bits
Trace.FindAll sYmbol vchar Data 0x9x

Trace.FindAll sYmbol vchar Data !0x9x
```

```
; find all write cycles to symbol address vchar in which the data value
, changed
Trace.FindAll Address vchar CYcle Write CHANGE Data
```

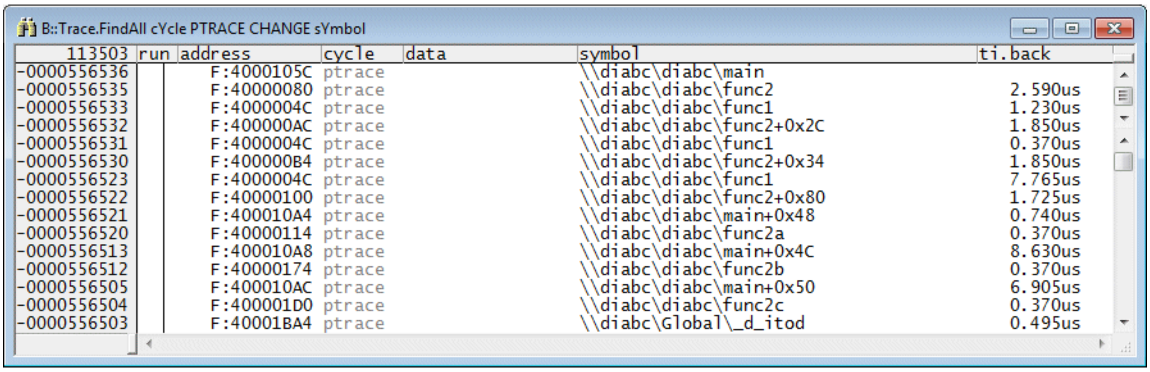


```
      record run address     cycle   data      symbol                 ti.back
-0000000024       D:40004068 wr-byte        3D \\diabc\Global\vchar    1.230us
-0000000023       D:40004068 wr-byte        FA \\diabc\Global\vchar    1.235us
-0000000022       D:40004068 wr-byte        F6 \\diabc\Global\vchar    1.230us
-0000000021       D:40004068 wr-byte        F6 \\diabc\Global\vchar    1.472ms
-0000000020       D:40004068 wr-byte        BF \\diabc\Global\vchar    1.235us
-0000000019       D:40004068 wr-byte        51 \\diabc\Global\vchar    1.230us
-0000000018       D:40004068 wr-byte        AC \\diabc\Global\vchar    1.235us
-0000000017       D:40004068 wr-byte        D0 \\diabc\Global\vchar    1.230us
-0000000016 BRK                                                        1.436ms
```

B::Trace.FindAll Address vchar CYcle Write CHANGE Data /Track

```
        5113 run address     cycle   data      symbol                 ti.back
-0000000032       D:40004068 wr-byte        D2 \\diabc\Global\vchar    1.235us
-0000000030       D:40004068 wr-byte        FD \\diabc\Global\vchar    1.468ms
-0000000029       D:40004068 wr-byte        53 \\diabc\Global\vchar    1.230us
-0000000028       D:40004068 wr-byte        D4 \\diabc\Global\vchar    1.235us
-0000000027       D:40004068 wr-byte        80 \\diabc\Global\vchar    1.230us
-0000000025       D:40004068 wr-byte        BF \\diabc\Global\vchar    1.473ms
-0000000024       D:40004068 wr-byte        3D \\diabc\Global\vchar    1.230us
-0000000023       D:40004068 wr-byte        FA \\diabc\Global\vchar    1.235us
-0000000022       D:40004068 wr-byte        F6 \\diabc\Global\vchar    1.230us
-0000000020       D:40004068 wr-byte        BF \\diabc\Global\vchar    1.474ms
-0000000019       D:40004068 wr-byte        51 \\diabc\Global\vchar    1.230us
-0000000018       D:40004068 wr-byte        AC \\diabc\Global\vchar    1.235us
-0000000017       D:40004068 wr-byte        D0 \\diabc\Global\vchar    1.230us
```

```
; find write accesses to variable flags with specified data values
; start search at specified record number down to the end of the
; trace recording
Trace.Find -3224833. Address V.RANGE(flags) Data 0x0--0xaa CYcle Write
; find next match
Trace.Find

; find read access to variable flags[3]
; start search at the specified record number up to the beginning of the
; trace recording
Trace.Find -3224832. Address Var.RANGE(flags[3]) CYcle Read  /Back

; find all program trace cycles for which the symbol range changed
Trace.FindAll cYcle PTRACE CHANGE sYmbol
```


B::Trace.FindAll cYcle PTRACE CHANGE sYmbol

```
     113503 run address         cycle   data              symbol                                ti.back
-0000556536          F:4000105C ptrace                    \\diabc\diabc\main
-0000556535          F:40000080 ptrace                    \\diabc\diabc\func2                     2.590us
-0000556533          F:4000004C ptrace                    \\diabc\diabc\func1                     1.230us
-0000556532          F:400000AC ptrace                    \\diabc\diabc\func2+0x2C                1.850us
-0000556531          F:4000004C ptrace                    \\diabc\diabc\func1                     0.370us
-0000556530          F:400000B4 ptrace                    \\diabc\diabc\func2+0x34                1.850us
-0000556523          F:4000004C ptrace                    \\diabc\diabc\func1                     7.765us
-0000556522          F:40000100 ptrace                    \\diabc\diabc\func2+0x80                1.725us
-0000556521          F:400010A4 ptrace                    \\diabc\diabc\main+0x48                 0.740us
-0000556520          F:40000114 ptrace                    \\diabc\diabc\func2a                    0.370us
-0000556513          F:400010A8 ptrace                    \\diabc\diabc\main+0x4C                 8.630us
-0000556512          F:40000174 ptrace                    \\diabc\diabc\func2b                    0.370us
-0000556505          F:400010AC ptrace                    \\diabc\diabc\main+0x50                 6.905us
-0000556504          F:400001D0 ptrace                    \\diabc\diabc\func2c                    0.370us
-0000556503          F:40001BA4 ptrace                    \\diabc\Global\_d_itod                  0.495us
```

Time Information

```
; find all trace entries with a TIme.Back time between 500.us--700.us
Trace.Find TIme.Back 0.500us--0.700us
```

```
; find all exits of the function sieve (Address sYmbol.EXIT(sieve))
; for which the time distance to the function entry of sieve is between
; 70 and 71 us (TIme.AddressBack sieve 70.us--71.us)

; display the time distance to the function entry of sieve and the
; default display items (/List TIme.AddressBack sieve DEFault)

Trace.FindAll Address sYmbol.EXIT(sieve) TIme.AddressBack sieve 70.us--71.us \
/List TIme.AddressBack sieve DEFault TIme.Back.OFF
```

The following commands delivery an analogous result.

Trace.FindAll Address *<program_addr_a>* **TIme.AddressBack** *<program_addr_b> <time_range>*

Trace.FindAll Address *<program_addr_b>* **TIme.AddressFore** *<program_addr_a> <time_range>*

Var

```
; use the keyword Var to search for static variable access cycles:
Trace.Find Var     ; find the first one
Trace.Find         ; find the next one by repeating the command
                   ; without Var

Trace.FindAll Var  ;list all occurrences in the Trace.FindAll window
Trace.ListVar      ;same as above
```

GROUP

```
; find any trace information assigned to the group sieve
; start search at the beginning of the trace recording
Trace.Find GROUP "sieve"
```

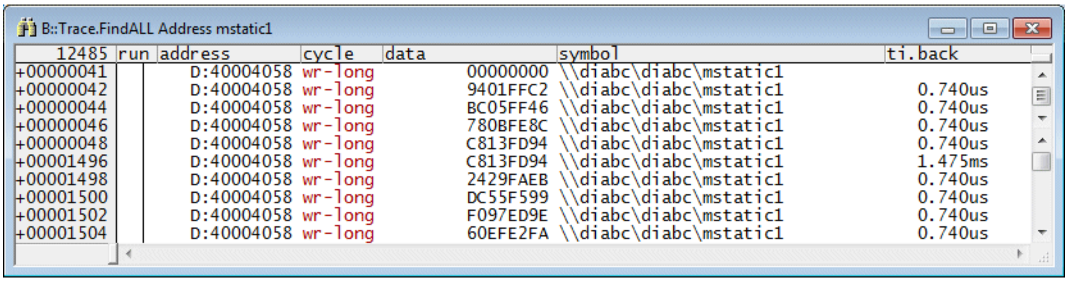
CORE

```
; find entry to function func8 executed by core 0
; start search at the beginning of the trace recording
Trace.Find Address func8 CORE 0.
```

Format the Result

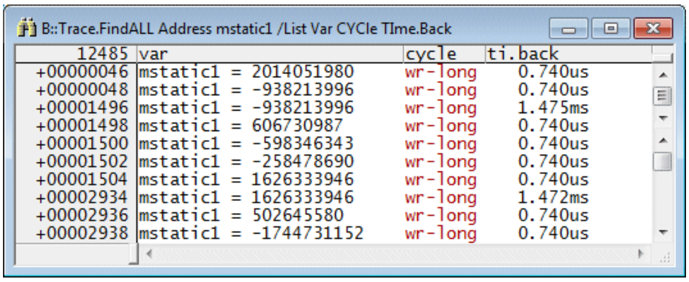
; find all occurrences of the specified search event
; use the option /List to format the result
Trace.FindAll [<*record_number*> | <*record_range*>] <*items*> … **/List** <*items*> …

```
; if nothing is specified the default display is used
Trace.FindAll Address mstatic1
```


B::Trace.FindALL Address mstatic1

```
    12485 run address         cycle    data             symbol                       ti.back
+00000041        D:40004058 wr-long          00000000 \\diabc\diabc\mstatic1
+00000042        D:40004058 wr-long          9401FFC2 \\diabc\diabc\mstatic1             0.740us
+00000044        D:40004058 wr-long          BC05FF46 \\diabc\diabc\mstatic1             0.740us
+00000046        D:40004058 wr-long          780BFE8C \\diabc\diabc\mstatic1             0.740us
+00000048        D:40004058 wr-long          C813FD94 \\diabc\diabc\mstatic1             0.740us
+00001496        D:40004058 wr-long          C813FD94 \\diabc\diabc\mstatic1             1.475ms
+00001498        D:40004058 wr-long          2429FAEB \\diabc\diabc\mstatic1             0.740us
+00001500        D:40004058 wr-long          DC55F599 \\diabc\diabc\mstatic1             0.740us
+00001502        D:40004058 wr-long          F097ED9E \\diabc\diabc\mstatic1             0.740us
+00001504        D:40004058 wr-long          60EFE2FA \\diabc\diabc\mstatic1             0.740us
```

```
; advise TRACE32 to display the result as follows:
; variable display, cycle type, timestamp
Trace.FindAll Address mstatic1 /List Var TIme.Back
```


B::Trace.FindALL Address mstatic1 /List Var CYCle TIme.Back

```
    12485 var                       cycle    ti.back
+00000046 mstatic1 = 2014051980     wr-long    0.740us
+00000048 mstatic1 = -938213996     wr-long    0.740us
+00001496 mstatic1 = -938213996     wr-long    1.475ms
+00001498 mstatic1 = 606730987      wr-long    0.740us
+00001500 mstatic1 = -598346343     wr-long    0.740us
+00001502 mstatic1 = -258478690     wr-long    0.740us
+00001504 mstatic1 = 1626333946     wr-long    0.740us
+00002934 mstatic1 = 1626333946     wr-long    1.472ms
+00002936 mstatic1 = 502645580      wr-long    0.740us
+00002938 mstatic1 = -1744731152    wr-long    0.740us
```

Related TRACE32 Functions

The **Trace.Find** command can be used together with the following functions:

FOUND()	Returns TRUE if search item was found.
TRACK.RECORD()	Returns the number of the record in which the search item was found.
TRACK.ADDRESS()	Returns the address listed in the record in which the search item was found.
ADDRESS.OFFSET()	Extracts the hex-address from the address object returned by TRACK.ADDRESS().

```
Trace.Find (-2000.)--(0.) DATA.L 0x28

IF FOUND()
    Trace.REF TRACK.RECORD()
```

```
Trace.Find , DATA.L 0x28

IF FOUND()
    PRINT "0x28 written to address " ADDRESS.OFFSET(TRACK.ADDRESS())
```

Convert Setting in Trace Find Dialog to a TRACE32 Command

Push button	Direction Radio button	Search procedure
Find First	**Down**	Trace.Find *<expert_items>*
Find First	**Up**	Trace.Find *<expert_items>* /Back
Find Here	**Down**	Trace.Find *<record_number> <expert_items>*
Find Here	**Up**	Trace.Find *<record_number> <expert_items>* /Back
Find Next	**Down**	Trace.Find
Find Next	**Up**	Trace.Find
Find All	—	Trace.FindAll *<expert_items>*

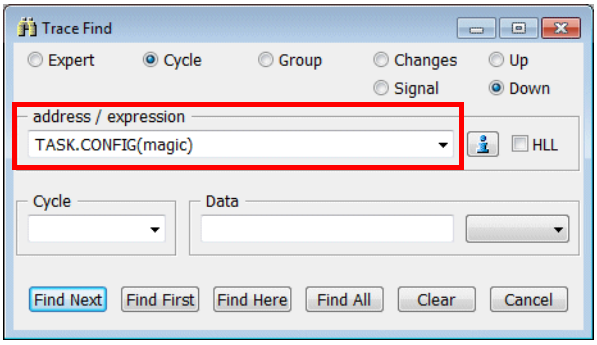
Use Cases

Find Task Switches

There are two methods to generate task switch information if **OS-aware tracing** is used:

- Trace packets for the write accesses to the variable identified by TASK.CONFIG(magic).
- Task switch packets.

Address TASK.CONFIG(magic)



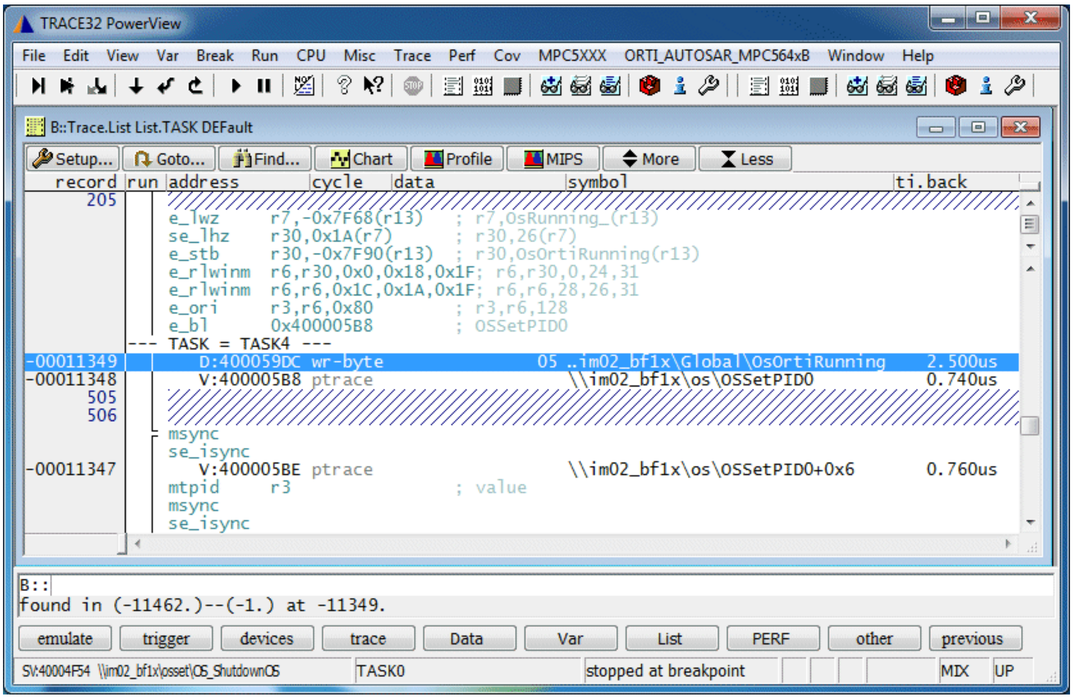
Expert [✓ Cycle] Group Changes Up
Signal [✓ Down]

address / expression: TASK.CONFIG(magic)

HLL [ ]

Cycle:

Data:



TRACE32 PowerView
File Edit View Var Break Run CPU Misc Trace Perf Cov MPC5XXX ORTI_AUTOSAR_MPC564xB Window Help
B::Trace.List List.TASK DEFault
Setup... Goto... Find... Chart Profile MIPS More Less
emulate trigger devices trace Data Var List PERF other previous

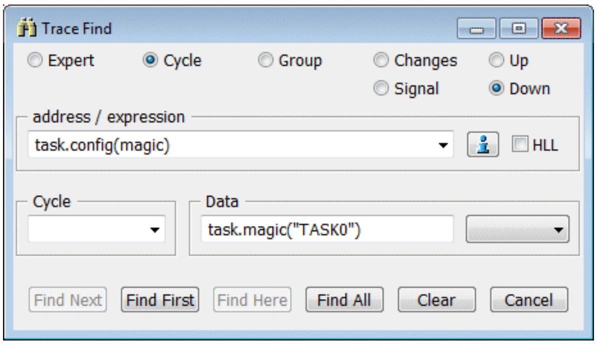
```
   record run address         cycle    data          symbol                              ti.back
      205
              e_lwz       r7,-0x7F68(r13)     ; r7,OsRunning_(r13)
              se_lhz      r30,0x1A(r7)        ; r30,26(r7)
              e_stb       r30,-0x7F90(r13)    ; r30,OsOrtiRunning(r13)
              e_rlwinm    r6,r30,0x0,0x18,0x1F; r6,r30,0,24,31
              e_rlwinm    r6,r6,0x1C,0x1A,0x1F; r6,r6,28,26,31
              e_ori       r3,r6,0x80          ; r3,r6,128
              e_bl        0x400005B8          ; OSSetPID0
          --- TASK = TASK4 ---
-00011349         D:400059DC wr-byte          05 ..im02_bf1x\Global\OsOrtiRunning    2.500us
-00011348         V:400005B8 ptrace              \\im02_bf1x\os\OSSetPID0            0.740us
      505
      506
              msync
              se_isync
-00011347         V:400005BE ptrace              \\im02_bf1x\os\OSSetPID0+0x6        0.760us
              mtpid       r3                  ; value
              msync
              se_isync
```

```
B::
found in (-11462.)--(-1.) at -11349.
```

```
SV:40004F54 \\im02_bf1x\osset\OS_ShutdownOS   TASK0   stopped at breakpoint   MIX   UP
```

```
Trace.List List.TASK DEFault

Trace.Find Address TASK.CONFIG(magic)

; SMP systems: find task switch on core 0
; Trace.Find Address TASK.CONFIG(magic[0])
```

The following command allows to search for the entry to a specified task:



address / expression: task.config(magic)

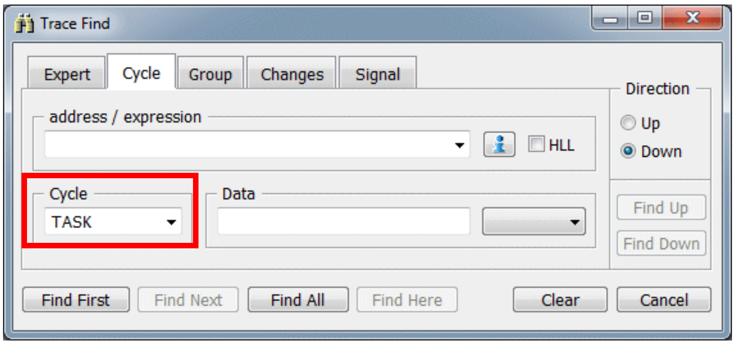
Cycle:

Data: task.magic("TASK0")

```
Trace.Find Address TASK.CONFIG(magic) Data.any TASK.MAGIC("TASK0")
```

Task Switch Packets

If a Context ID or ownership packet is decoded and if its value is assignable to a task, the "**task**" cycle type and the task name is displayed. The displayed data value is a TRACE32 internal value.

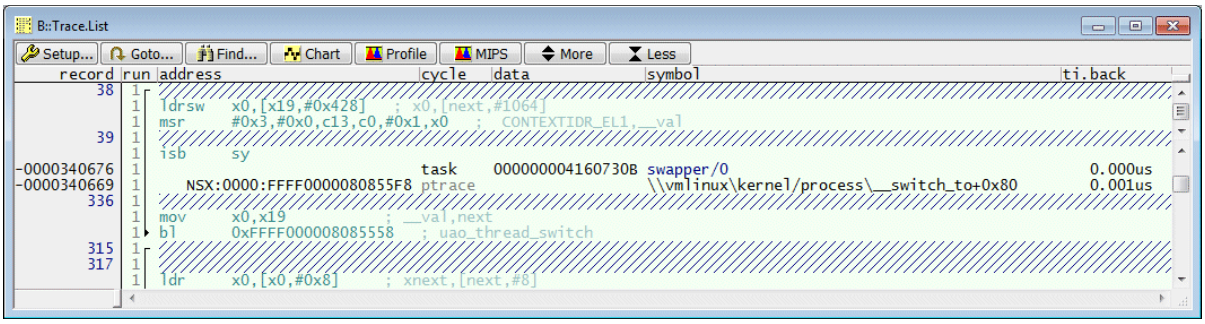


address / expression
HLL

Direction
Up [✓ Down]

Cycle: TASK

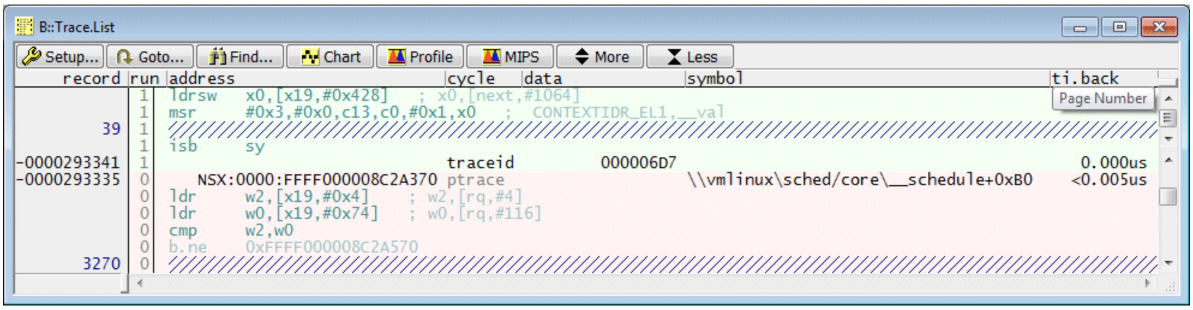
Data



```
      record run address                   cycle    data              symbol                                    ti.back
          38  1
              1   ldrsw    x0,[x19,#0x428]    ; x0,[next,#1064]
              1   msr      #0x3,#0x0,c13,c0,#0x1,x0    ;  CONTEXTIDR_EL1,__val
          39  1
              1   isb      sy
-0000340676   1                            task     000000004160730B swapper/0                                 0.000us
-0000340669   1      NSX:0000:FFFF0000080855F8 ptrace                \\vmlinux\kernel/process\__switch_to+0x80  0.001us
         336  1
              1   mov      x0,x19             ; __val,next
              1   bl       0xFFFF000008085558  ; uao_thread_switch
         315  1
         317  1
              1   ldr      x0,[x0,#0x8]       ; xnext,[next,#8]
```

```
Trace.List

Trace.Find CYcle TASK
```

If a Context ID or ownership packet is decoded and if its value can not be assigned to a task or any other protocol-specific content, the cycle type "**traceid**" and the packet content is displayed.



```
      record run address                   cycle    data              symbol                                    ti.back
              1   ldrsw    x0,[x19,#0x428]    ; x0,[next,#1064]
              1   msr      #0x3,#0x0,c13,c0,#0x1,x0    ;  CONTEXTIDR_EL1,__val
          39  1
              1   isb      sy
-0000293341   1                            traceid           000006D7                                        0.000us
-0000293335   0      NSX:0000:FFFF000008C2A370 ptrace                \\vmlinux\sched/core\__schedule+0xB0     <0.005us
              0   ldr      w2,[x19,#0x4]      ; w2,[rq,#4]
              0   ldr      w0,[x19,#0x74]     ; w0,[rq,#116]
              0   cmp      w2,w0
              0   b.ne     0xFFFF000008C2A570
        3270  0
```

Both cycle types can be found with the following command:

```
Trace.Find CYcle TASK OR CYcle TRACEID
```

Trace.Find Keyword Reference

In the context of this reference, Match means that the trace record meets the conditions and will be found by **Trace.Find** and listed by **Trace.FindAll**.

<find-expr>:	*<or-expr>* [**OR** *<find-expr>*]
<or-expr>:	*<and-expr>* [*<or-expr>*]
<and-expr>:	*<simple-expr>* *<op-expr>* **(** *<find-expr>* **)**
<simple-expr>:	*<digital-channel> <digital-value>* *<analog-channel>* **Greater** *<float>* *<analog-channel>* **Smaller** *<float>* *<analog-channel>* **INBound** *<float> <float>* *<analog-channel>* **BEYONDBound** *<float> <float>* *<analog-channel>* **Valid** *<word-channel> <word-expr>* see **<trace>.Find** reference for more options
<op-expr>:	**NOT** *<and-expr>* **APPEAR** *<and-expr>* **WITHIN** *<offset> <and-expr>* **CHANGE** *<item>* **APPEAR** *<and-expr>* **WITHIN** *<offset> <and-expr>* **AT** *<offset>* **(** *<find-expr>* **)**
<igital-value>:	**OFF | ON | Low | High | 0 | 1**
<word-expr>:	*<word-list>* **!** *<integer>* **!** *<integer-range>*

<word-list>:	*<integer>* [**|** *<word-list>*}
	<integer-range> [**|** *<word-list>*}

<offset>:	*<integer>* *<time>*

APPEAR Match when condition becomes a match

Format: **APPEAR** *<and-expr>*

Match if the specified condition is a match in the current trace record, but no match in the preceding record (edge detector).

Example: Find all records in which signal eXt.2 has a rising edge while signal eXt.3 is high.

```
Probe.FindAll APPEAR eXt.2 High eXt.3 High
```

AT Combine with condition at other record

Format: **AT** *<offset>* **(** *<find-expr>* **)**

Match if the condition is a match in a trace record relative to the current record. The offset can be specified either in records or as a time and may be positive or negative.

Note: It is possible to leave out the opening and closing parentheses in simple expressions. However, without these, the **AT** operator will have a different operator precedence than the other “prefix” operators **NOT**, **WITHIN** and **APPEAR**. These semantics are considered deprecated.

Example: Find all records in which the signal x.2 is low, but high two records later.

```
Probe.FindAll eXt.2 Low At 2 ( eXt.2 High )
```

CHANGE Match when selected item changed

Format: **CHANGE** *<item>*

Match if the value of the specified item changed compared to the preceding trace record.

NOT — Negate condition

Format: **NOT** *<and-expr>*

Invert the argument condition. The entire expression is considered a match if the argument condition does not match.

OR — OR condition

Format: **OR *<condition>***

Construct logical OR of multiple conditions. Note that multiple conditions concatenated without an operator are ANDed together, with the AND having a higher precedence.

WITHIN — Time restriction for condition

Format: **WITHIN** *<offset>* *<and-expr>*

The expression is considered a match if the argument *<and-expr>* matches for at least one record within a certain number of records or a certain time from the current record. The offset can be positive or negative.

Example: Look for setup or hold violations of a data bus relative to a clock edge:

```
Probe.FindAll APPEAR eXt.CLK 1 AT -20ns ( WITHIN 50ns CHANGE Word.DATA )
```

For each rising edge if the eXt.CLK signal, the expression is considered a match if a setup time of 20 ns or a hold time of 30 ns is violated.