JTAGjet-Trace

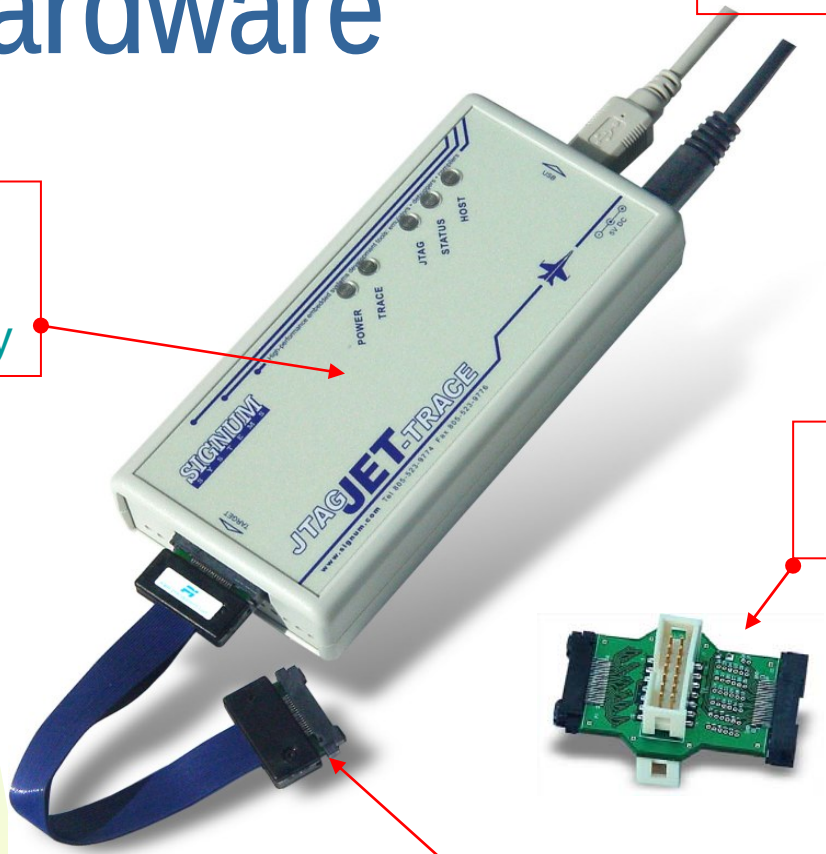
Signum Systems Corp.

Oct 1, 2010

JTAGjet-Trace Hardware

USB 2.0 High-Speed connection (480 Mbps)

Up to 4M frames deep (18 MBytes) ETM trace memory



ETM to JTAG adapter optional

38-pin Mictor connector contains all ETM and JTAG signals

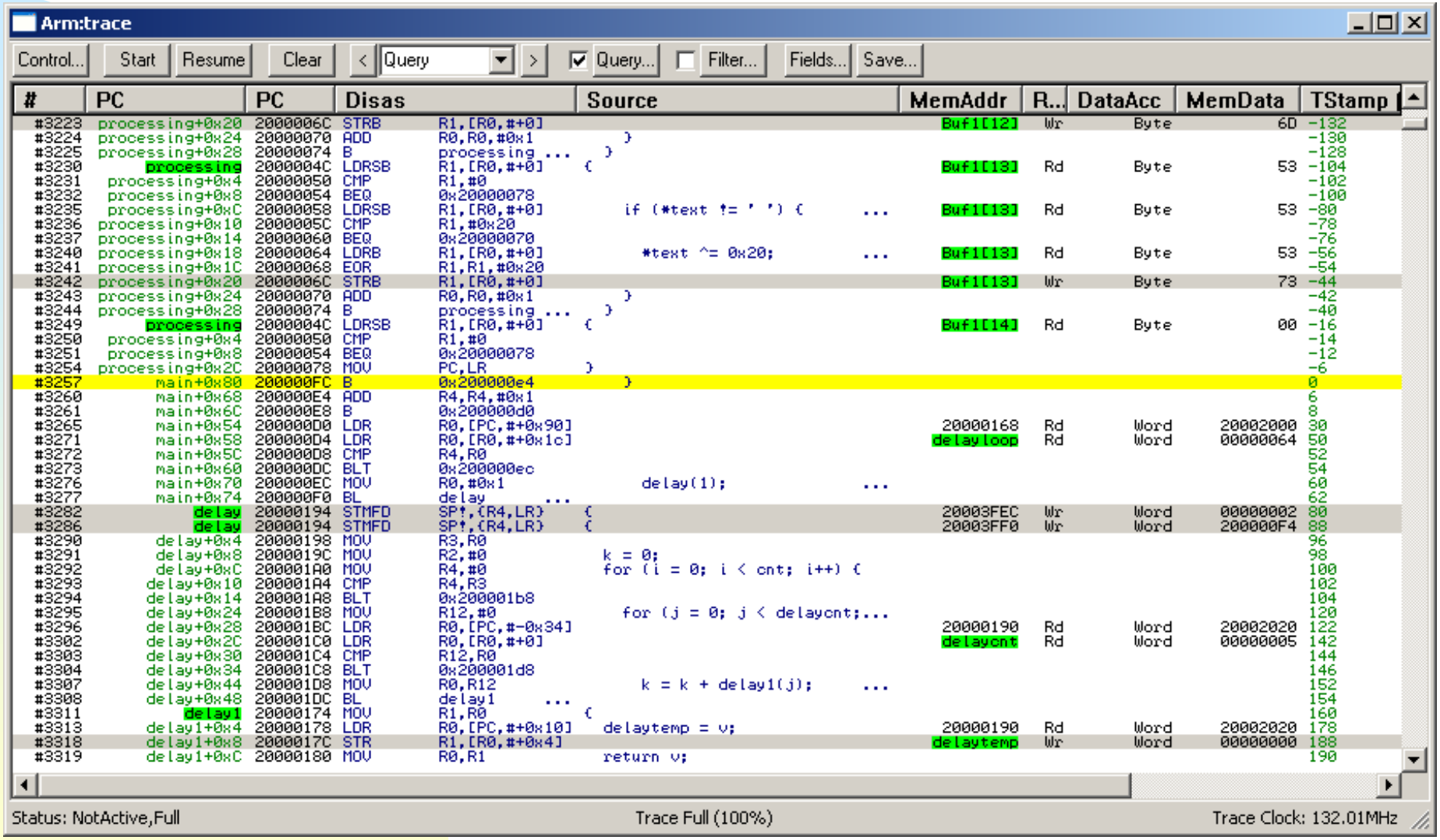
JTAGjet-Trace Features

- Up to 1.6 GHz CPU speed (200 MHz Trace CLK)
- Non-intrusive, real-time tracing
- Supports all ARM7, ARM9, ARM11 & Cortex ETMs
- Up to 18MB trace memory (4M x 36bits wide)
- 56-bit cycle accurate time stamp
- Traces all or in-range of PC addresses
- Trace on / off from specific PC address
- Traces all or range of data variables (bandwidth allowed)
- Traces data variable values
- Powerful trace post-filtering & timing display
- Compatible with all major C/C++ compilers/debuggers

Compatible ARM Debuggers

- Chameleon (Signum Systems) is included
- uVision (Keil-ARM)
- Code Composer Studio (Texas Instruments)
- RealView & ADS (ARM Ltd.)
- eBinder (eSol)
- Multi-2000 (Green Hills Software)
- GDB & Insight (GNU / Eclipse)
- EWARM (IAR)
- XRAY & EDGE (Mentor Graphics)
- Code Warrior (MetroWerks)
- CodeSourcery

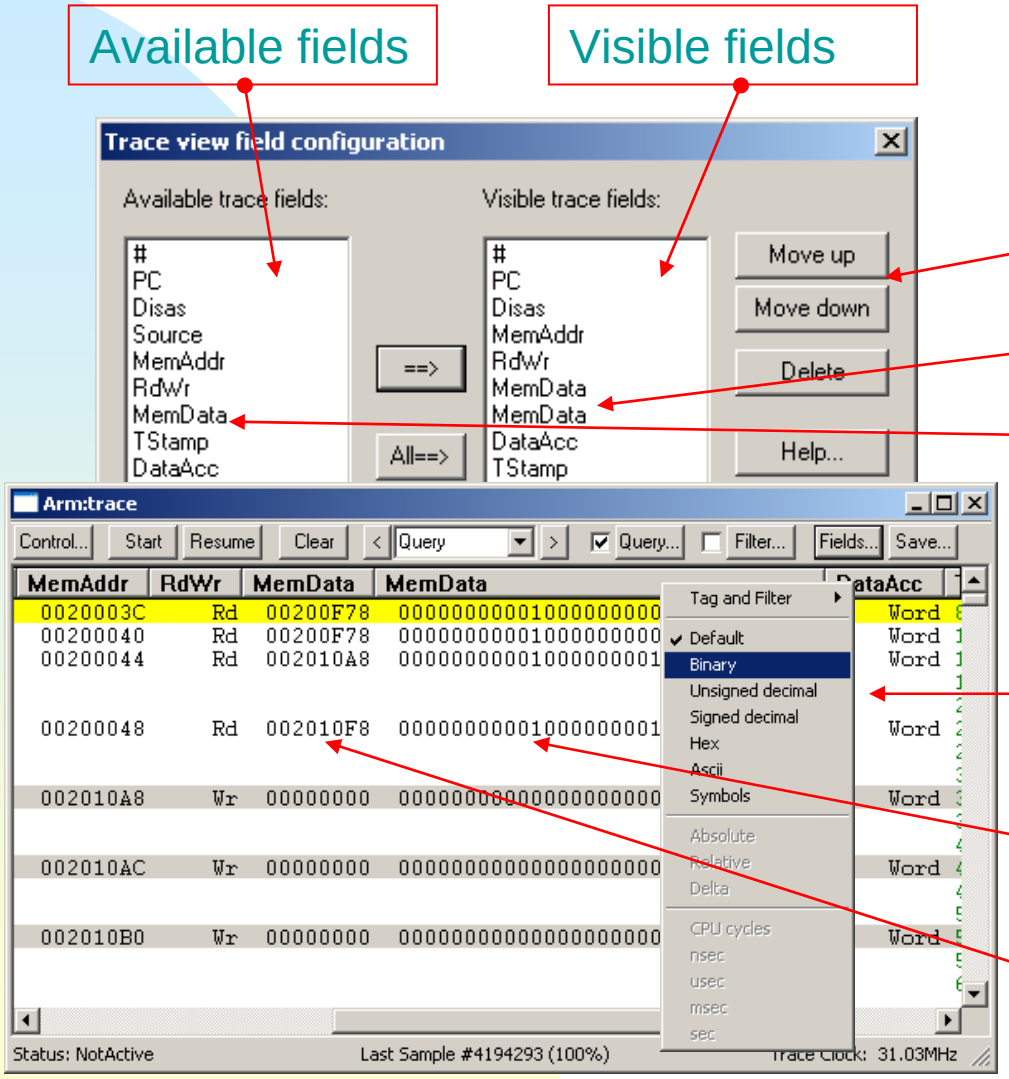
Trace Display -Overview



#	PC	PC	Disas		Source	MemAddr	R..	DataAcc	MemData	TStamp
#3223	processing+0x20	2000006C	STRB	R1,[R0,#+0]		Buf1[12]	Wr	Byte	6D	-132
#3224	processing+0x24	20000070	ADD	R0,R0,#0x1	}					-130
#3225	processing+0x28	20000074	B	processing ...	}					-128
#3230	processing	2000004C	LDRSB	R1,[R0,#+0]	{	Buf1[13]	Rd	Byte	53	-104
#3231	processing+0x4	20000050	CMP	R1,#0						-102
#3232	processing+0x8	20000054	BEQ	0x20000078						-100
#3235	processing+0xC	20000058	LDRSB	R1,[R0,#+0]	if (*text != ' ') { ...	Buf1[13]	Rd	Byte	53	-80
#3236	processing+0x10	2000005C	CMP	R1,#0x20						-78
#3237	processing+0x14	20000060	BEQ	0x20000070						-76
#3240	processing+0x18	20000064	LDRB	R1,[R0,#+0]	*text ^= 0x20; ...	Buf1[13]	Rd	Byte	53	-56
#3241	processing+0x1C	20000068	EOR	R1,R1,#0x20						-54
#3242	processing+0x20	2000006C	STRB	R1,[R0,#+0]		Buf1[13]	Wr	Byte	73	-44
#3243	processing+0x24	20000070	ADD	R0,R0,#0x1	}					-42
#3244	processing+0x28	20000074	B	processing ...	}					-40
#3249	processing	2000004C	LDRSB	R1,[R0,#+0]	{	Buf1[14]	Rd	Byte	00	-16
#3250	processing+0x4	20000050	CMP	R1,#0						-14
#3251	processing+0x8	20000054	BEQ	0x20000078						-12
#3254	processing+0x2C	20000078	MOV	PC,LR	}					-6
#3257	main+0x80	200000FC	B	0x200000e4	}					0
#3260	main+0x68	200000E4	ADD	R4,R4,#0x1						6
#3261	main+0x6C	200000E8	B	0x200000d0						8
#3265	main+0x54	200000D0	LDR	R0,[PC,#+0x90]		20000168	Rd	Word	20002000	30
#3271	main+0x58	200000D4	LDR	R0,[R0,#+0x1c]		delayloop	Rd	Word	00000064	50
#3272	main+0x5C	200000D8	CMP	R4,R0						52
#3273	main+0x60	200000DC	BLT	0x200000ec						54
#3276	main+0x70	200000EC	MOV	R0,#0x1	delay(1); ...					60
#3277	main+0x74	200000F0	BL	delay ...						62
#3282	delay	20000194	STMFD	SP!,{R4,LR}	{	20003FEC	Wr	Word	00000002	80
#3286	delay	20000194	STMFD	SP!,{R4,LR}	{	20003FF0	Wr	Word	200000F4	88
#3290	delay+0x4	20000198	MOV	R3,R0						96
#3291	delay+0x8	2000019C	MOV	R2,#0	k = 0;					98
#3292	delay+0xC	200001A0	MOV	R4,#0	for (i = 0; i < cnt; i++) {					100
#3293	delay+0x10	200001A4	CMP	R4,R3						102
#3294	delay+0x14	200001A8	BLT	0x200001b8						104
#3295	delay+0x24	200001B8	MOV	R12,#0	for (j = 0; j < delaycnt;...					120
#3296	delay+0x28	200001BC	LDR	R0,[PC,#-0x34]		20000190	Rd	Word	20002020	122
#3302	delay+0x2C	200001C0	LDR	R0,[R0,#+0]		delaycnt	Rd	Word	00000005	142
#3303	delay+0x30	200001C4	CMP	R12,R0						144
#3304	delay+0x34	200001C8	BLT	0x200001d8						146
#3307	delay+0x44	200001D8	MOV	R0,R12	k = k + delay1(j); ...					152
#3308	delay+0x48	200001DC	BL	delay1 ...						154
#3311	delay1	20000174	MOV	R1,R0	{					160
#3313	delay1+0x4	20000178	LDR	R0,[PC,#+0x10]	delaytemp = v;	20000190	Rd	Word	20002020	178
#3318	delay1+0x8	2000017C	STR	R1,[R0,#+0x4]		delaytemp	Wr	Word	00000000	188
#3319	delay1+0xC	20000180	MOV	R0,R1	return v;					190

Trace - Field Selection and Format

Fields visible in the trace window can be customized by user.



Order of fields can be defined

MemData field used twice

Fields available to display

Display format of each field may be customized too.

Field format selection - click on column header

Field in binary format (useful when displaying bit-register values)

Field in hex format (default)

Trace - Controls

Saving

Start/Stop

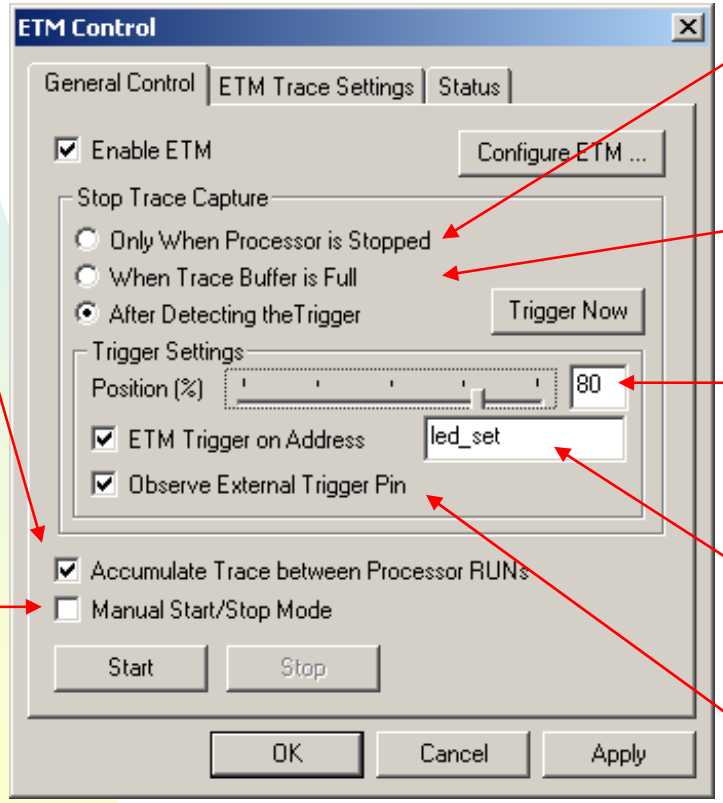
Clear buffer

Search&Tag

Post-filtering

Fields

Resume (start without clear).



Capture will only stop if CPU stops.

Capture will stop when trace buffer is full - CPU will continue to run.

Do not clear buffer before each CPU GO. Useful in source-level stepping.

80% of buffer will have trace before trigger, 20% will have trace after it.

Do not start trace when CPU starts. Start must be used manually. Useful for reset sequence tracing.

ETM trigger will happen on led_set function.

Observe the trigger pin on ETM connector.

SIGNUM SYSTEMS

www.signum.com

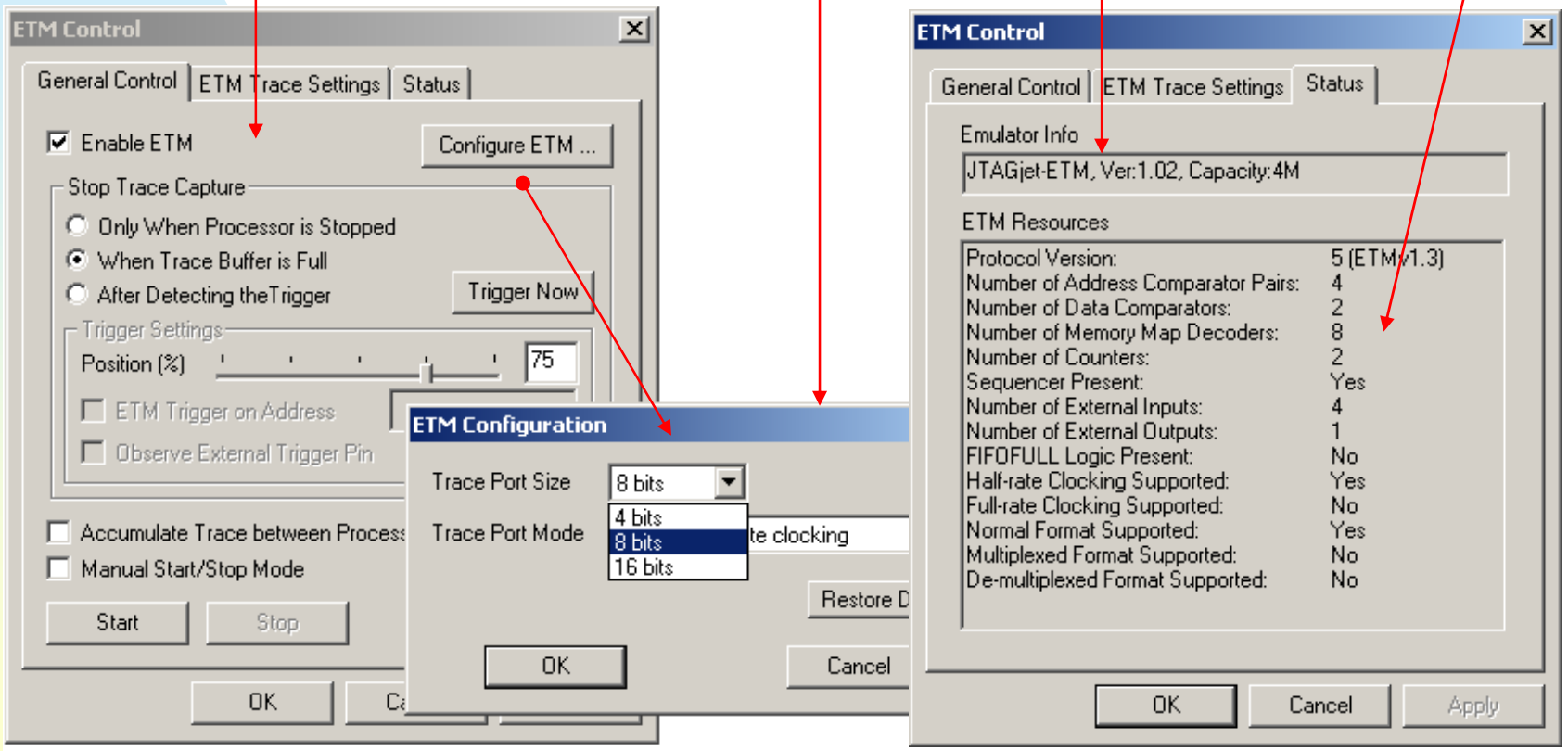
Trace - ETM Controls and Info

Emulator capture and trigger controls

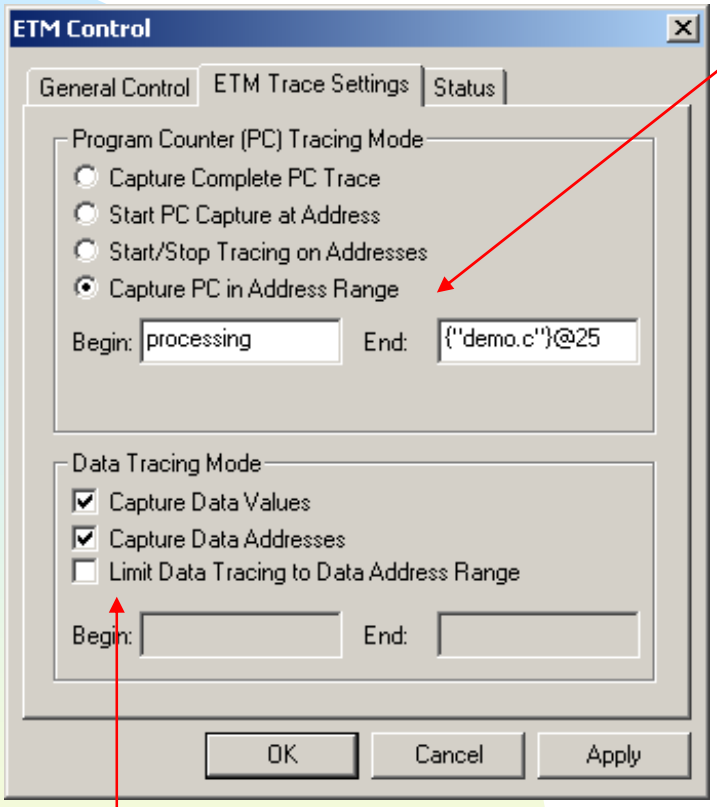
ETM port mode selection

Emulator info & trace size

On-chip ETM capabilities

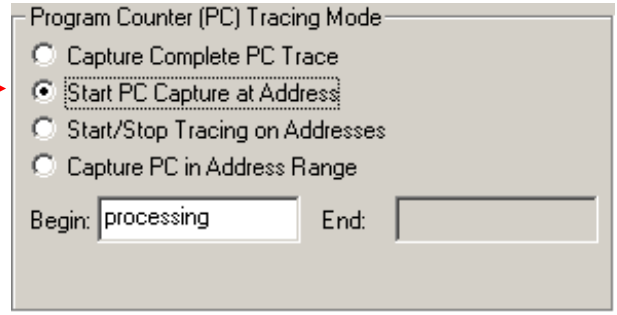


Trace - ETM Pre-Filtering

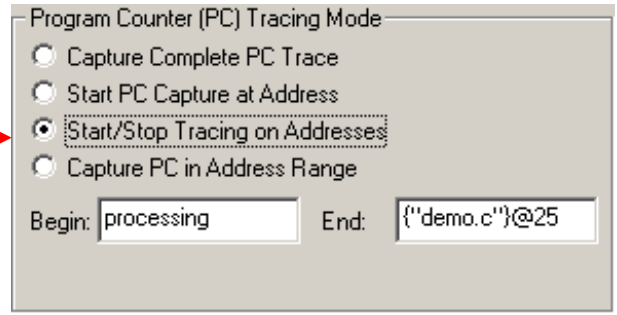


Trace PC in range only. Every call/jump outside range will stop tracing & resume on any return.

Do not trace PC until the code at 'Begin' executes. Useful to start trace on a rare event.

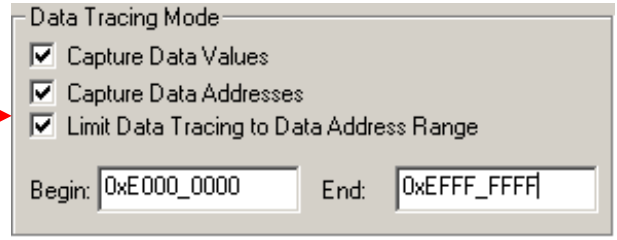


Start/stop mode starts on 'Begin' and stops on 'End' (latch mode)

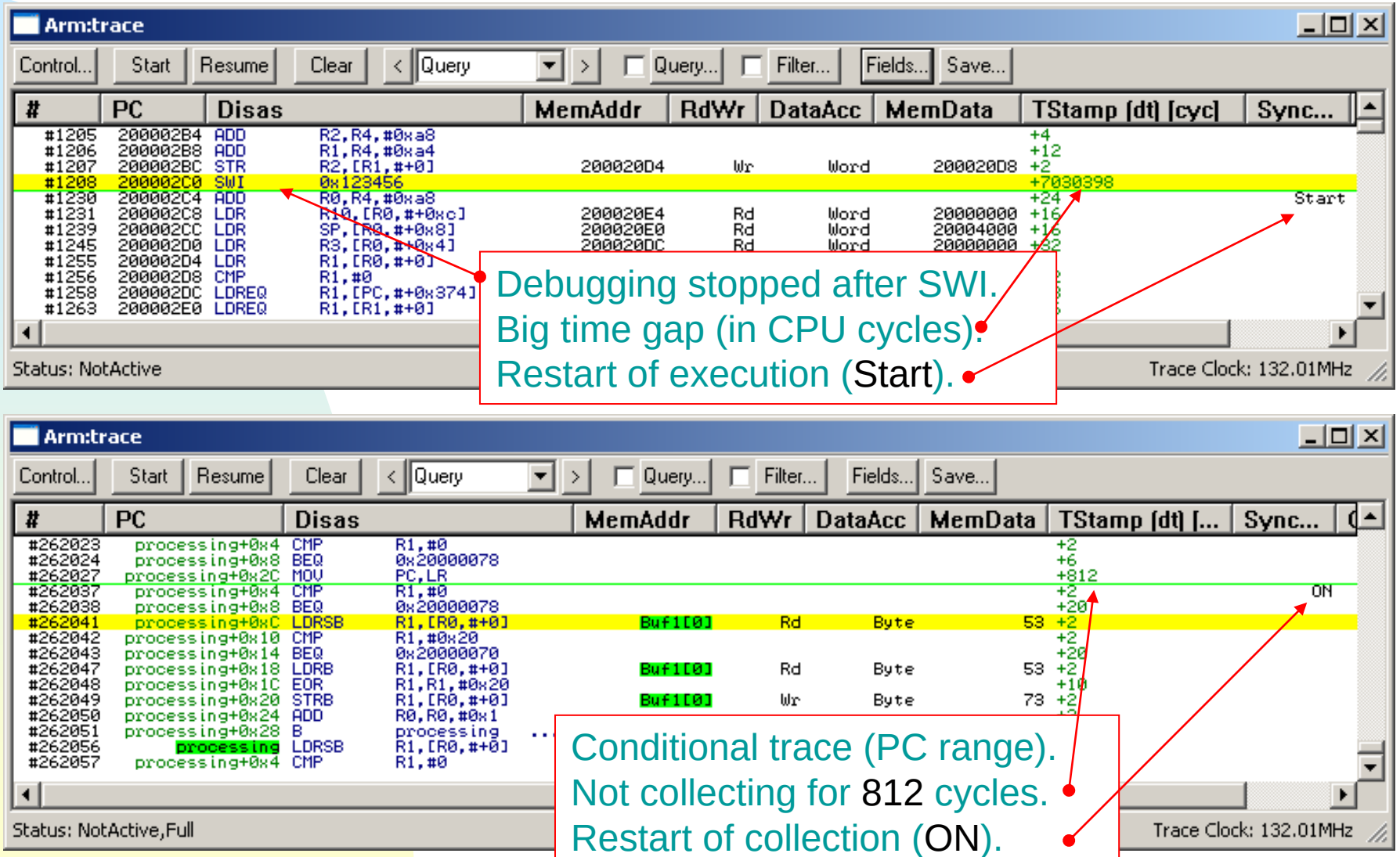


Data tracing of Values and/or Addresses

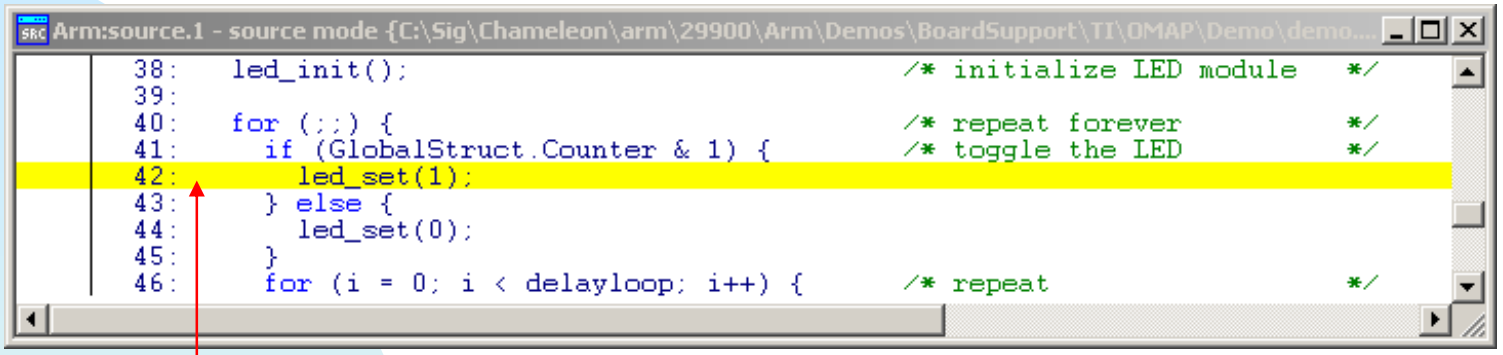
Limit data trace in range (very useful in 4-bit mode to prevent FIFO overflows). This range covers all on-chip registers.



Trace - Discontinuity



Trace - Synchronization with Source

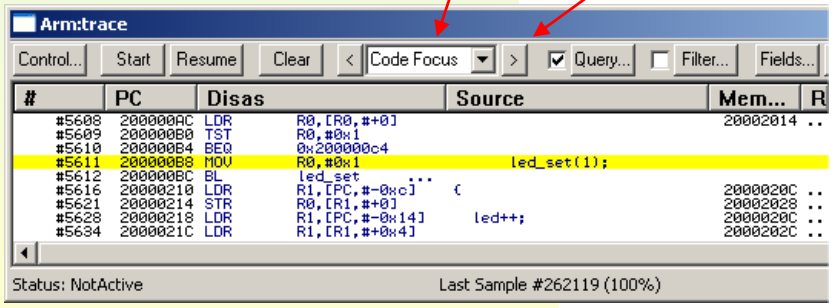


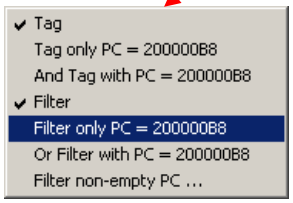
1) Click on source line in Source Window.

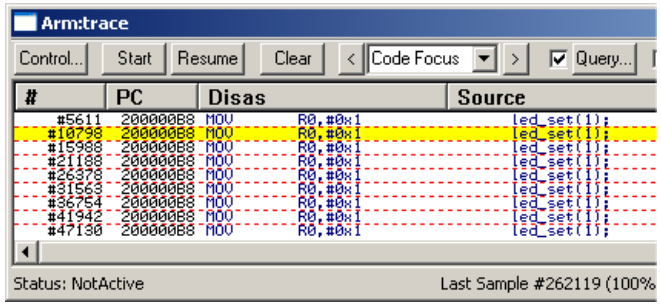
2) Set Code Focus as a search mode

3) Use **<** and **>** buttons to search for this line in trace (backward, forward)

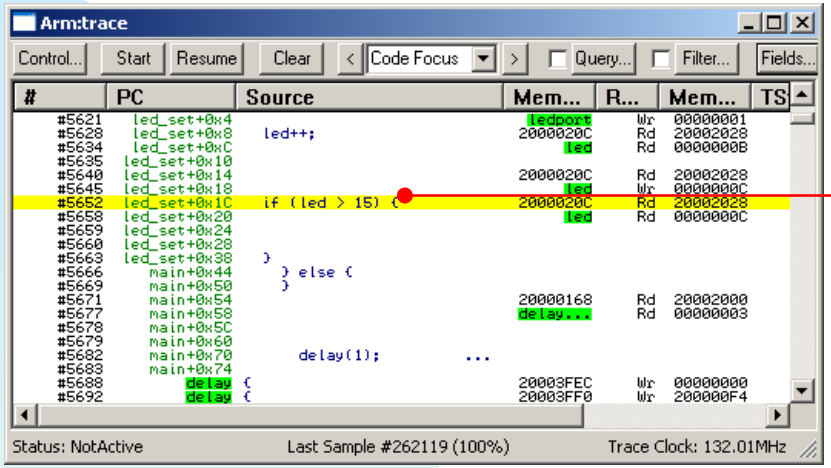
4) Use filtering to see only such lines in trace

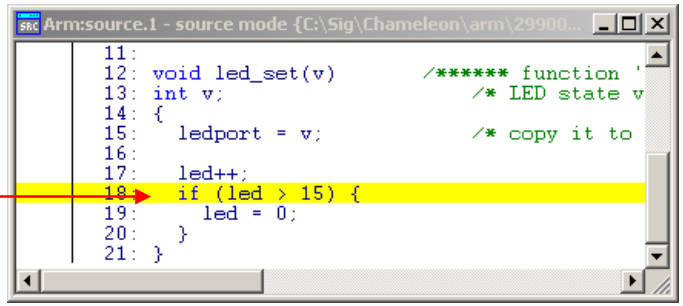




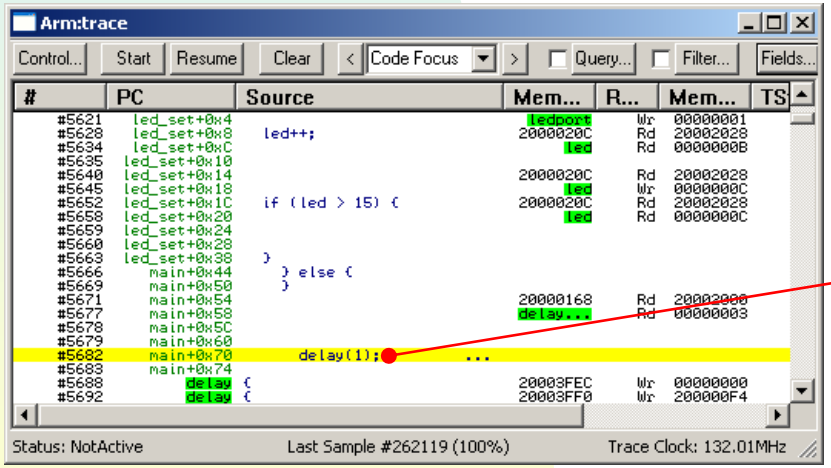


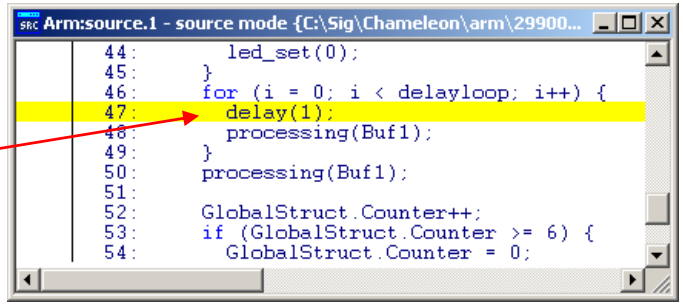
Trace - Synchronization to source



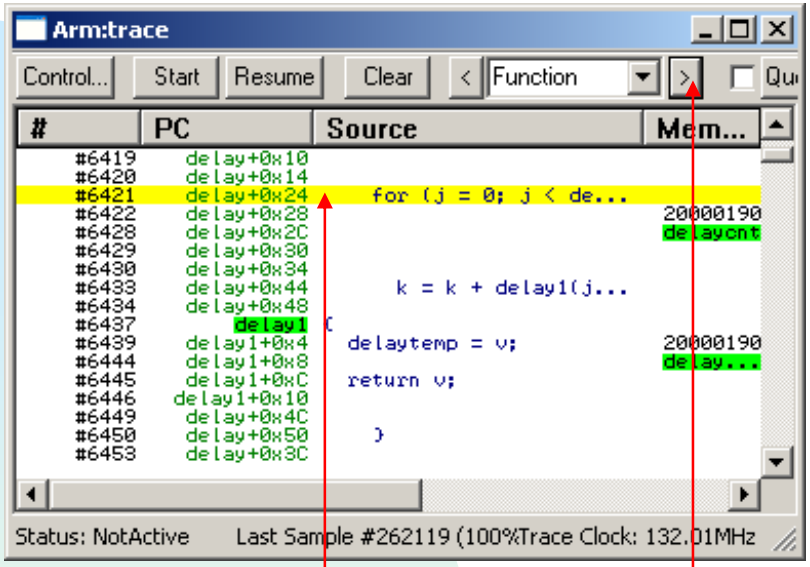


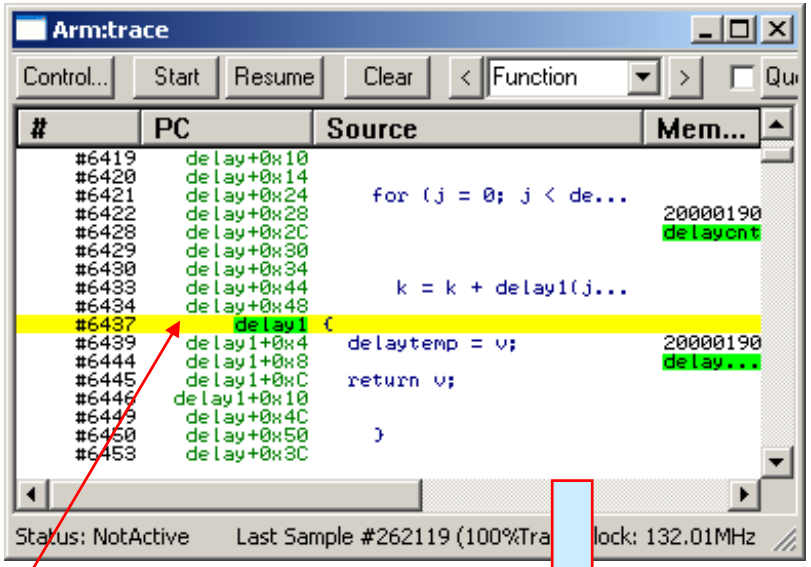
Click on any trace frame and the Source Window will show the corresponding source line instantly. Scroll back and forth the trace to see the program flow.

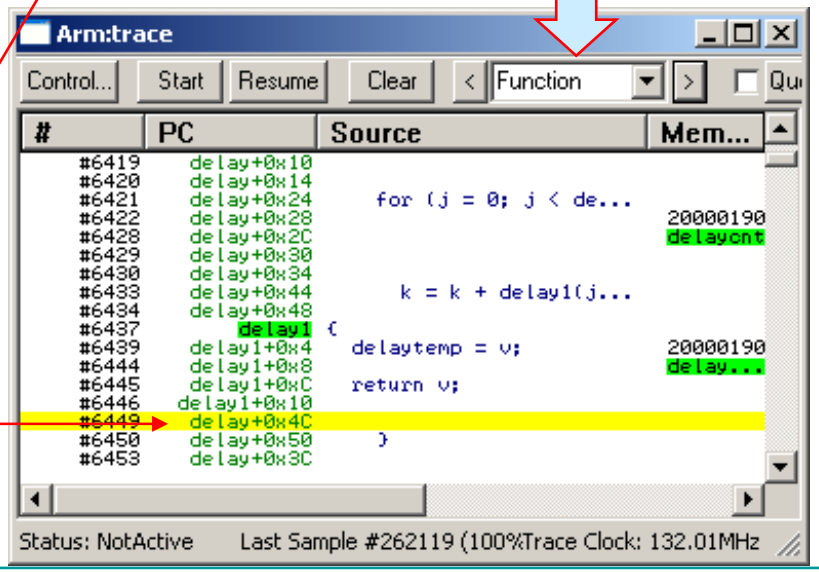




Trace - Function Navigation





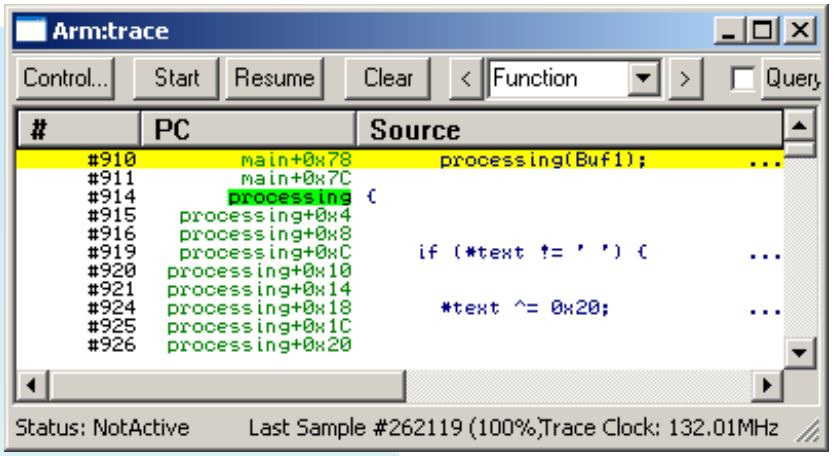


Initial position
(in function delay)

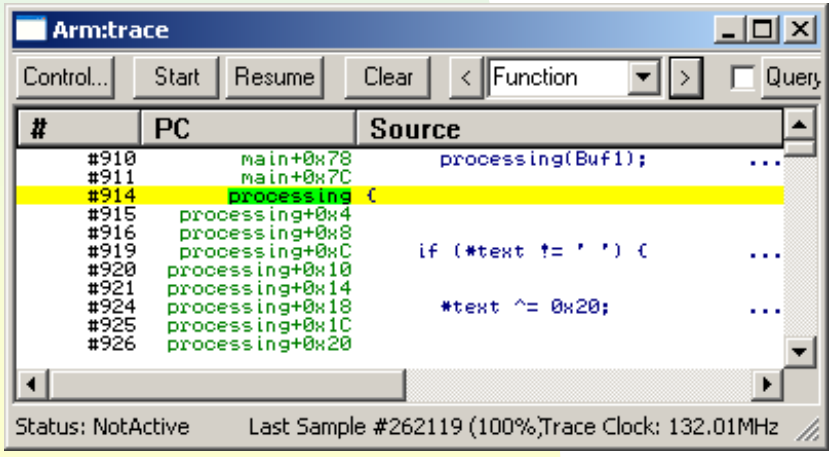
After 1-st click on **>**
(in function delay1)

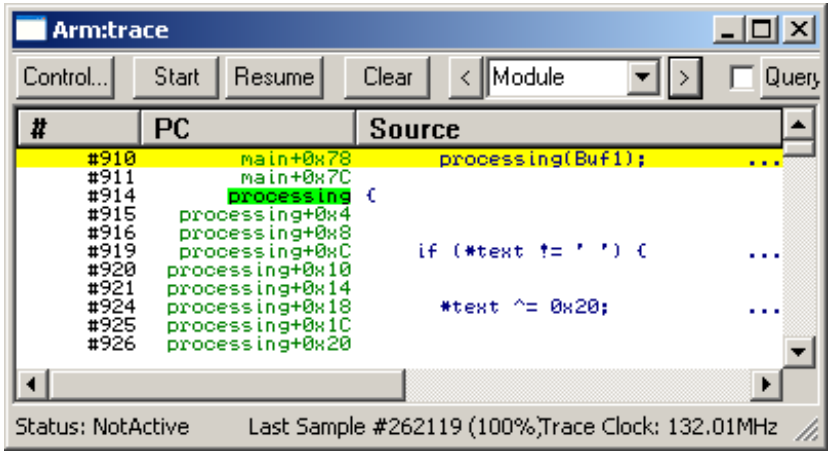
After 2-nd click on **>**
(back in function delay -
after return from delay1)

Trace - Function/Module Navigation

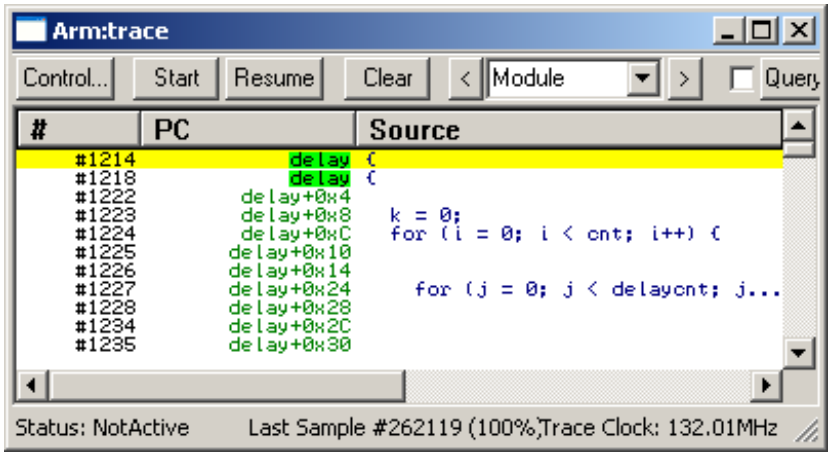


Function - function processing found (in same module as main).



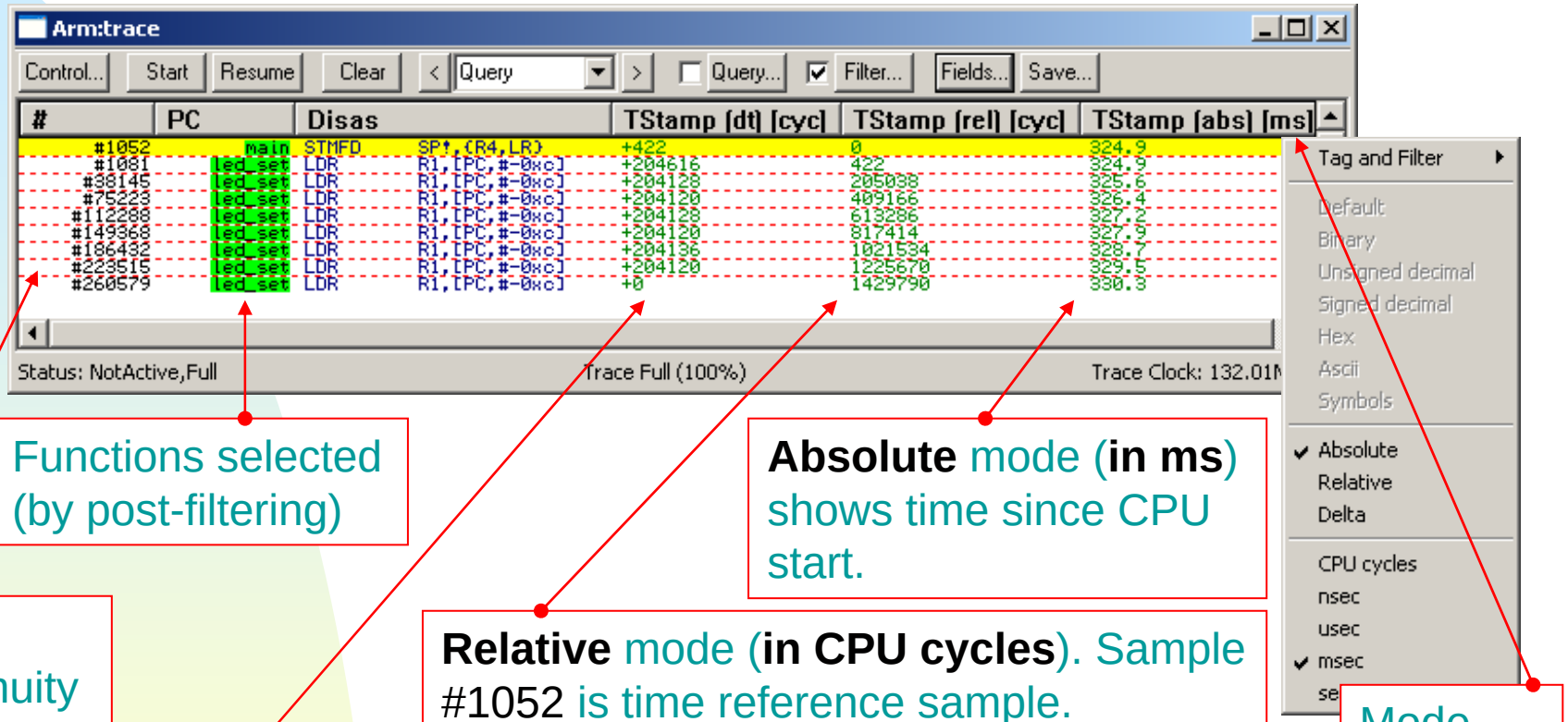


Module - function processing skipped (in same module as main). delay in different module found.



Trace - Timestamp Display

Timestamp is captured with CPU cycle accuracy - captured time can be shown in **3 modes (Absolute, Relative & Delta)** and as **CPU cycles** or **time units**.



Functions selected (by post-filtering)

Absolute mode (**in ms**) shows time since CPU start.

Red discontinuity marker (between post-filtered samples)

Relative mode (**in CPU cycles**). Sample #1052 is time reference sample.

Delta mode (**in CPU cycles**) shows distance in time to next displayed sample 422 cycles from main to led_set. About 204120 cycles between led_set calls.

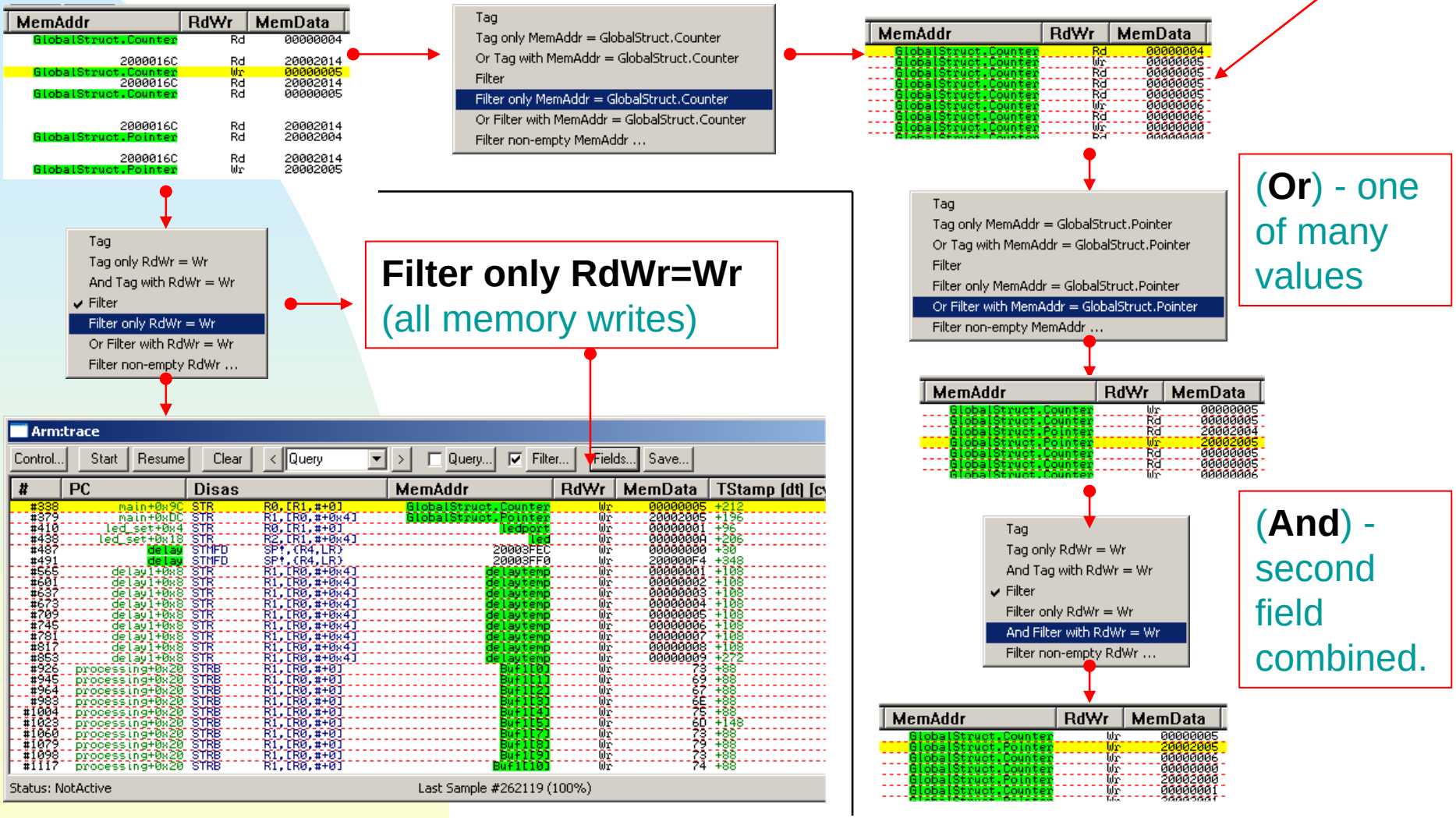
Mode change (click on column)

Trace - Quick Filtering (only/and/or)

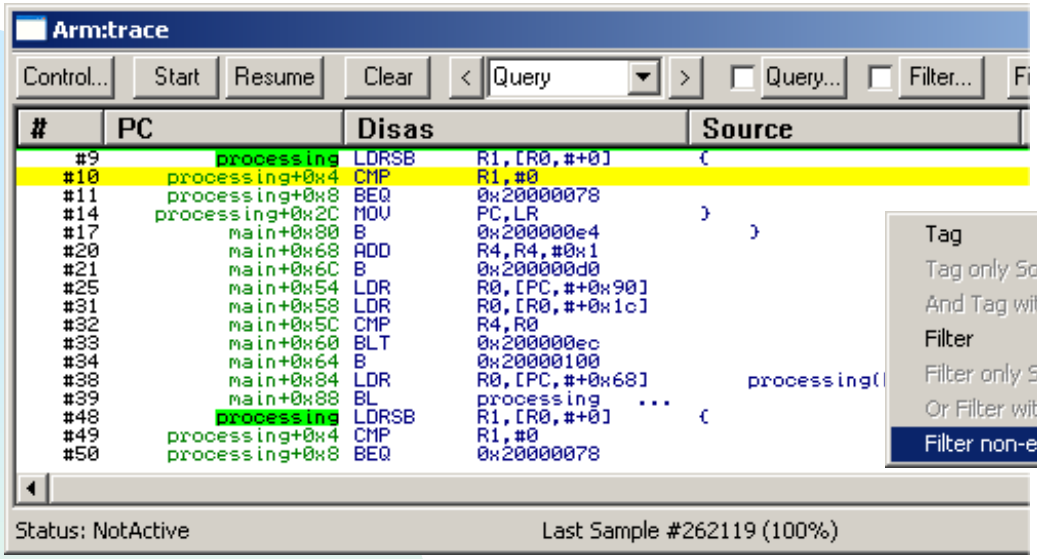
Unfiltered data

(**only**) - one value

– – – – - removed samples here



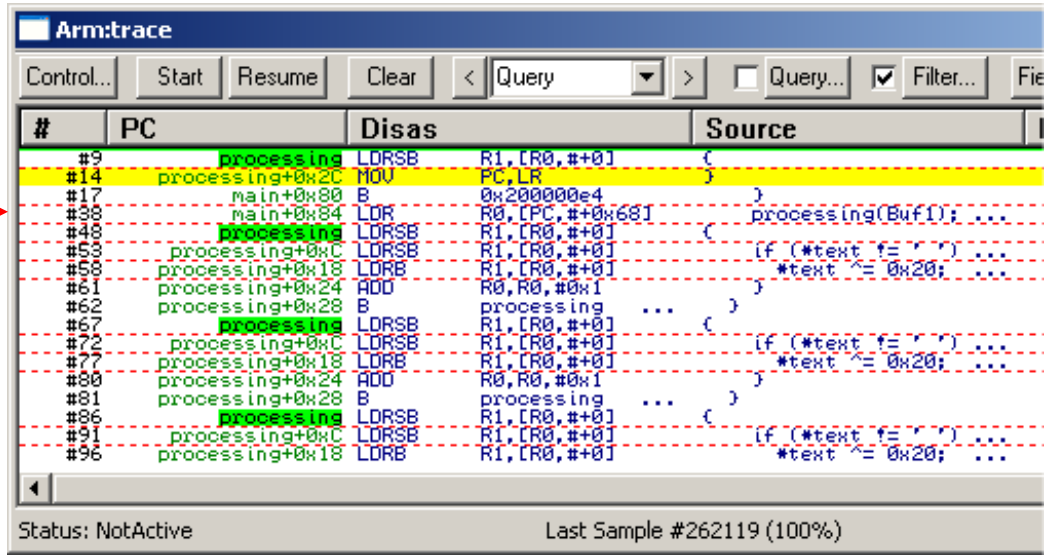
Trace - Quick Filtering (sources)



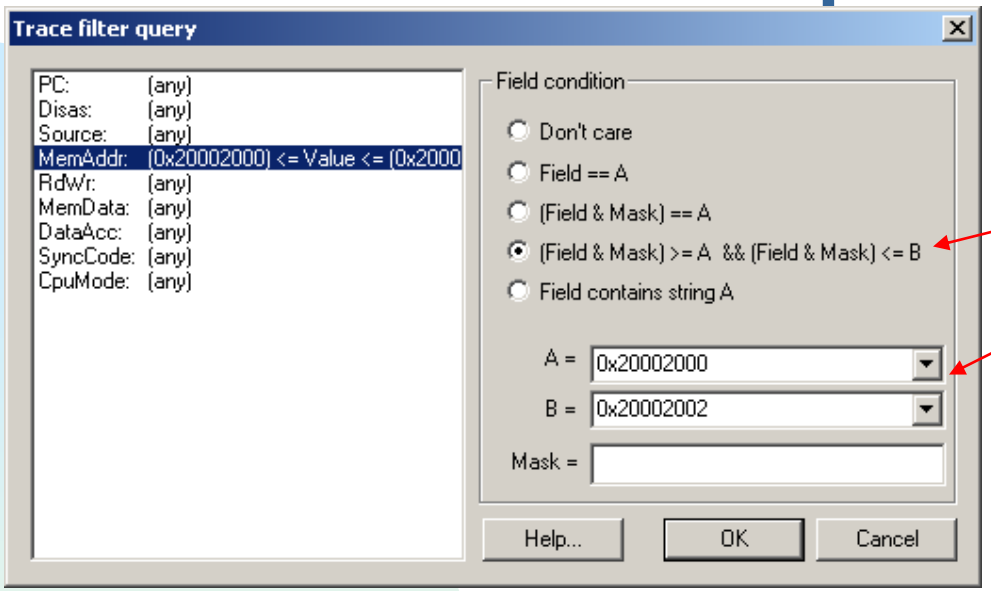
Before:
Code without sources visible.

Double-click popup menu

After:
Only code with sources visible.



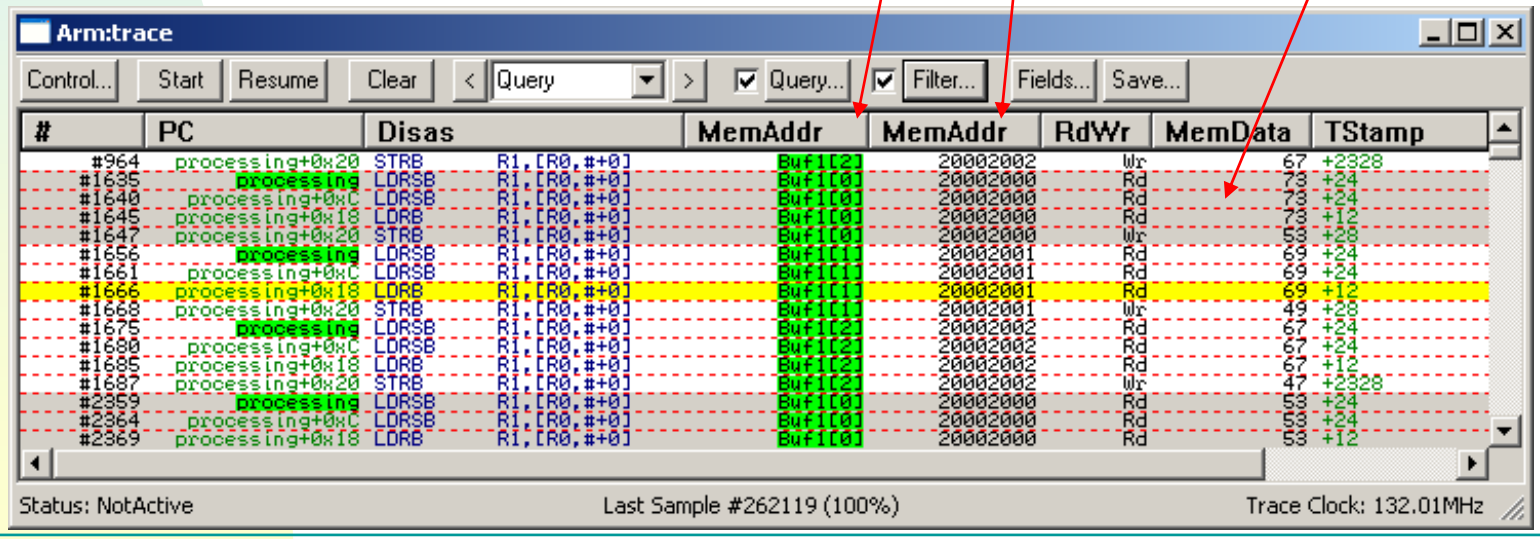
Trace - Complex Filtering (1)



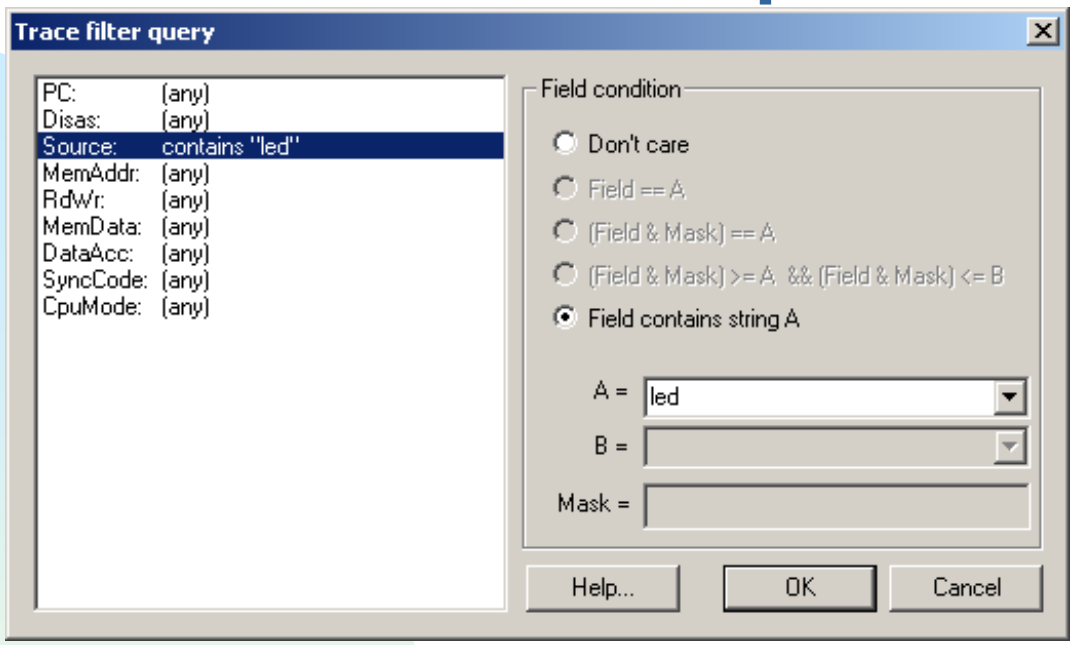
Display samples with memory address in specified range.

Buf[0] tagged.

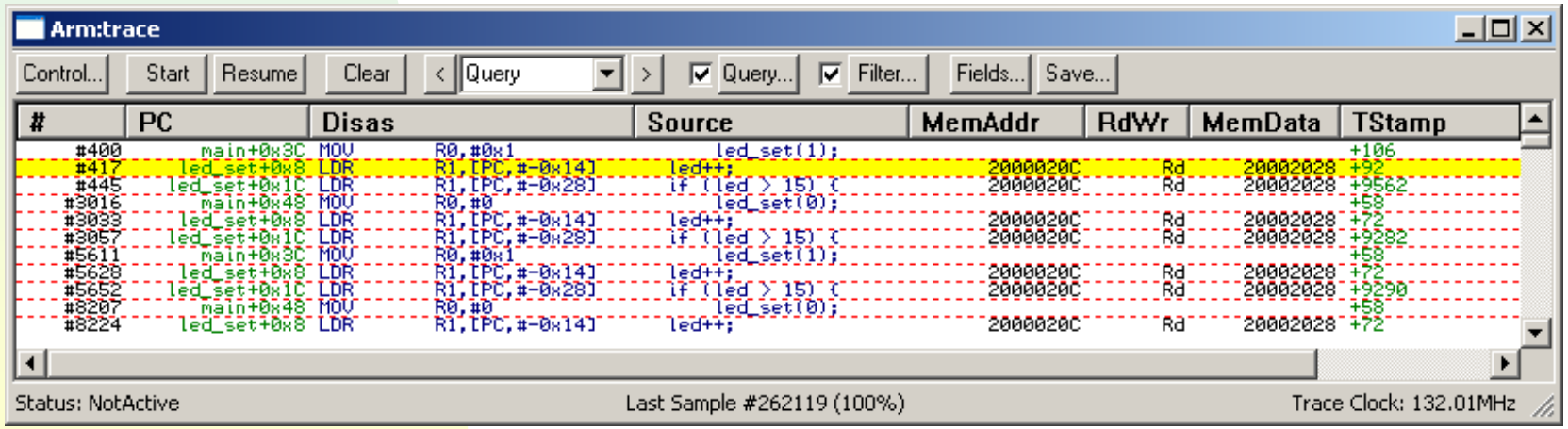
MemAddr as symbols and number.



Trace - Complex Filtering (2)

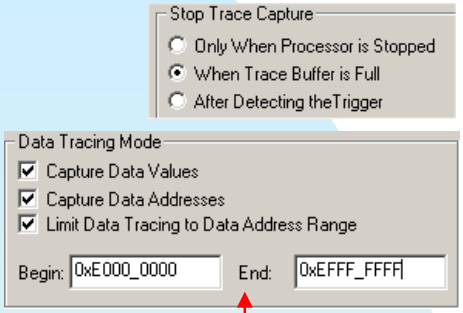


Display only samples with text "led" in Source field.



Tracing CPU Initialization

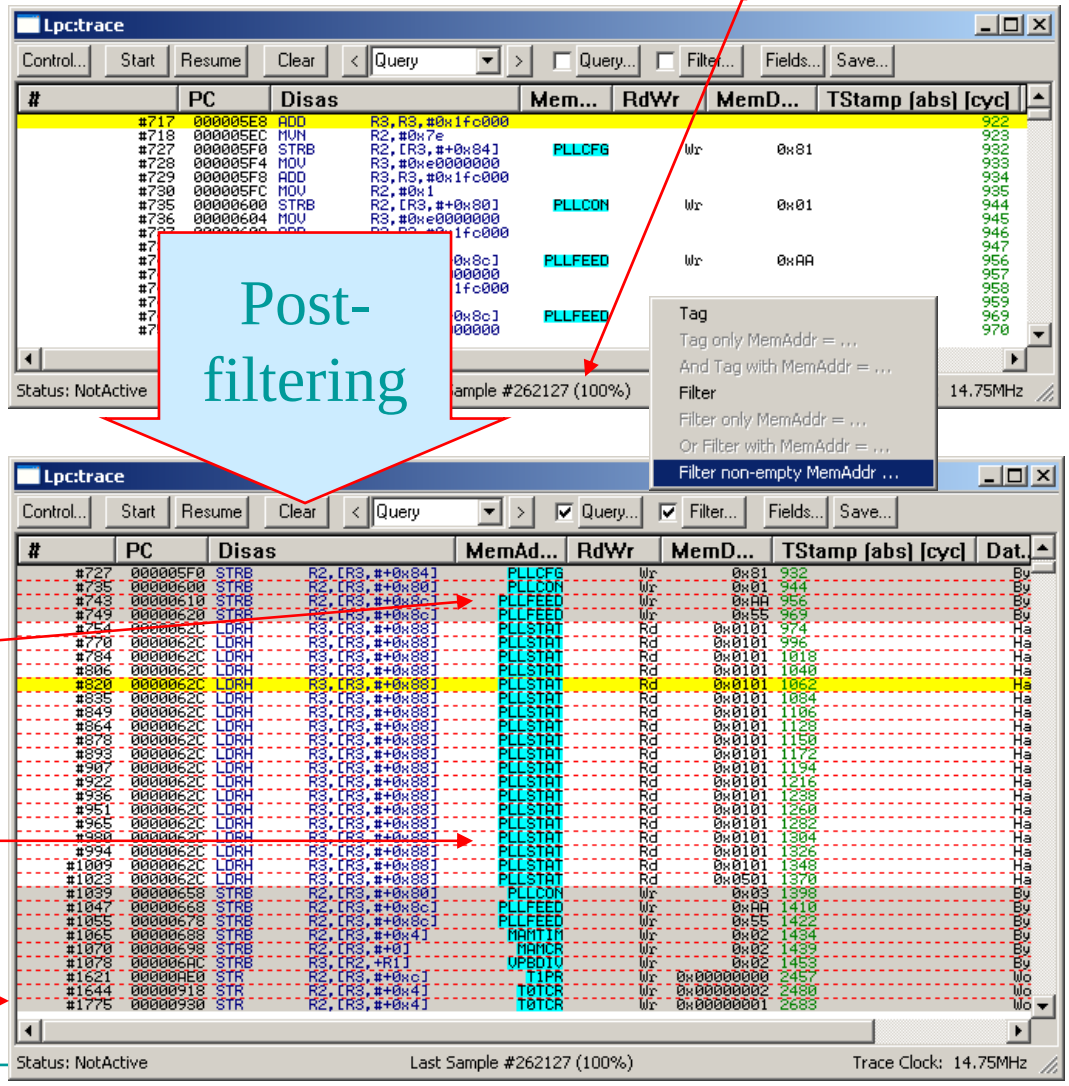
ETM is disabled on power-up and CPU reset. Start from 0 needed.



Start from 0

Data trace for registers only!! (very important in 4-bit ETM).

Full trace captured, CPU still running.



Post-filtering

All register accesses visible (PLL initialization).

Loop while waiting for PLL setup (reading PLLSTAT)

All writes tagged (RdWr=Wr)

Trace - Save to EXCEL File

Limiting range of records to save.

Bookmark

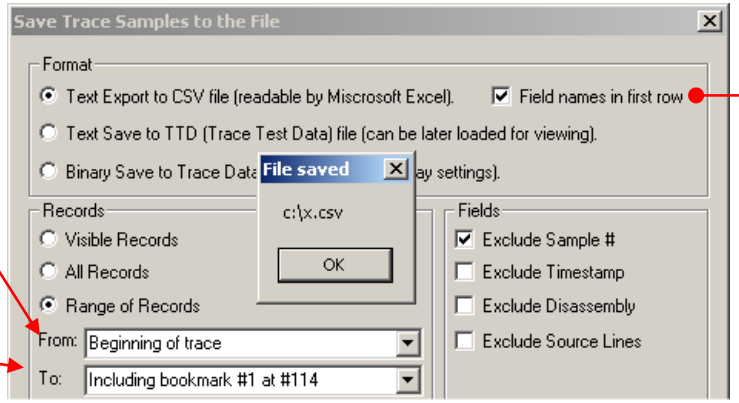
Field names in first row.

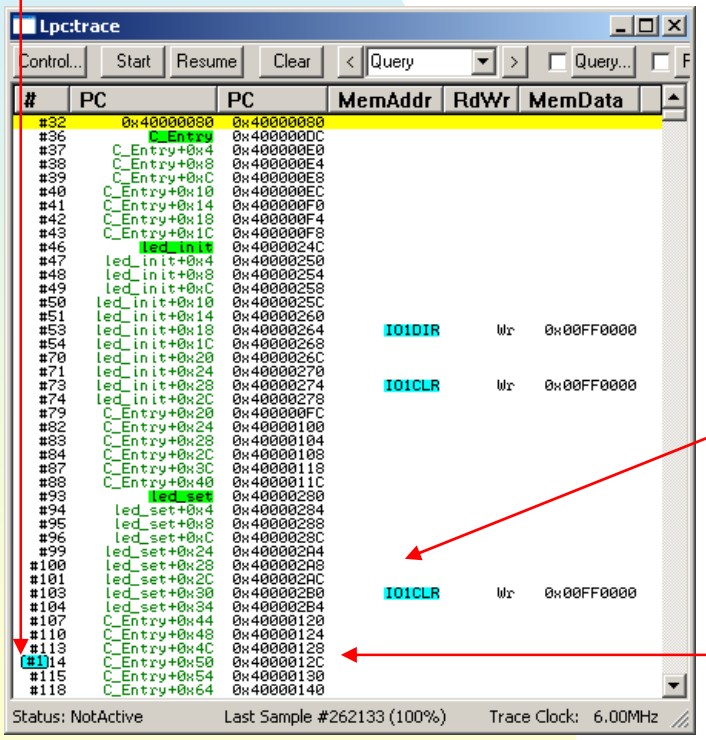
Saved file (shown in split view).

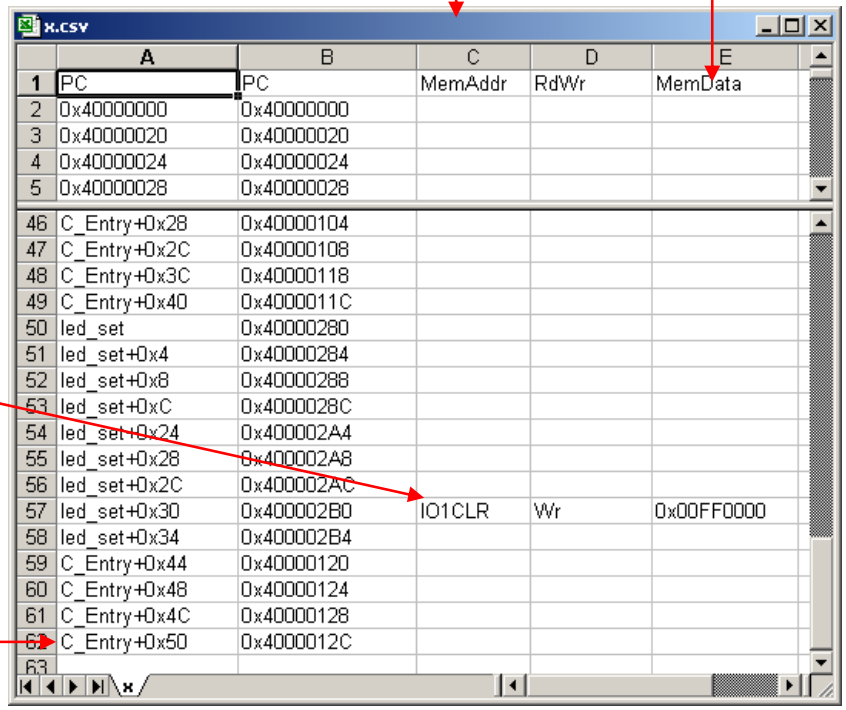
Save to CSV

Register name

Last saved sample **(Bookmark #1)**





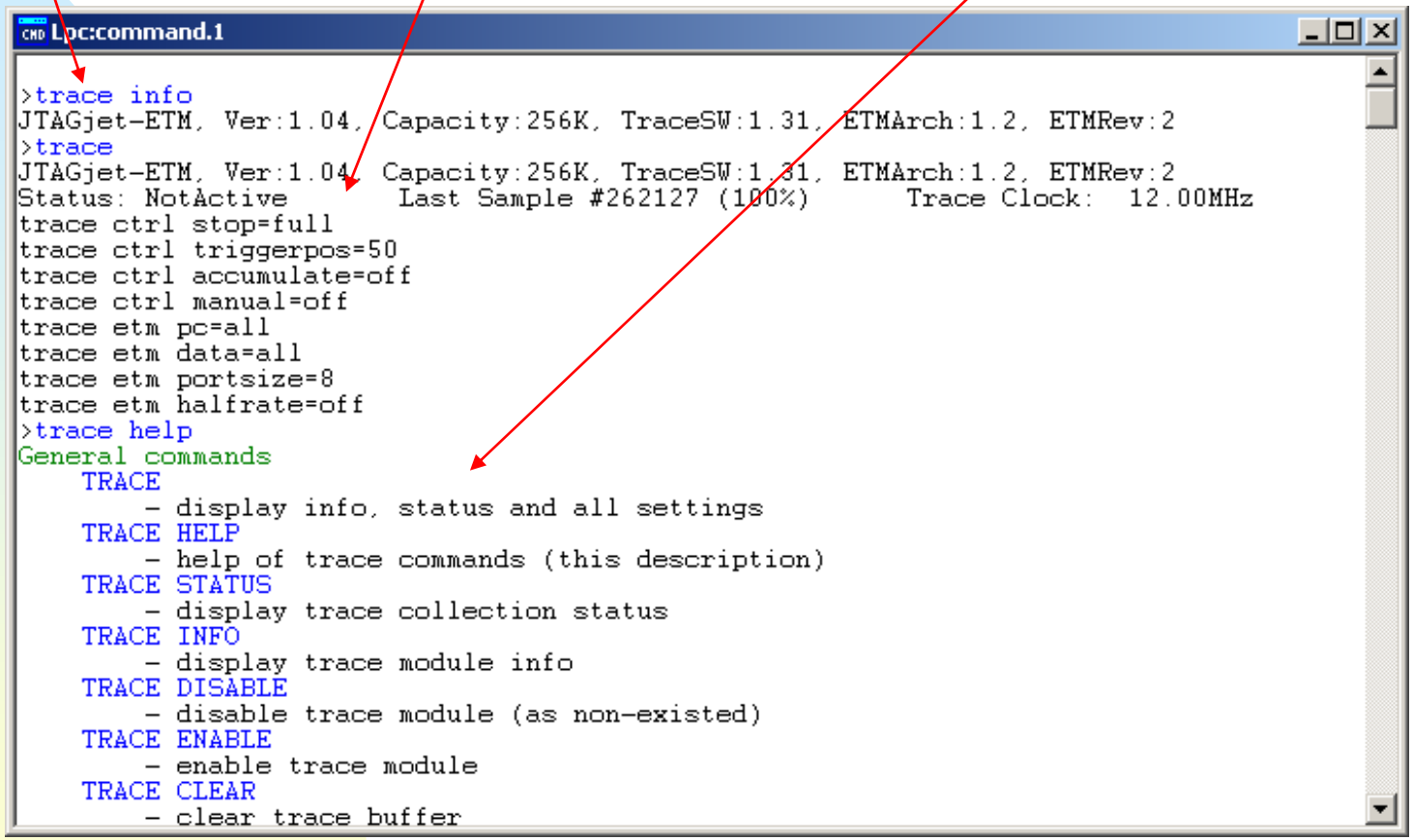


Trace - Command Line Access

Short info

Complete settings display

Command help



Trace - Integration with External Debuggers (CCS/ADS/GNU/etc.)