

For immediate Release

Task sensitive embOS plug-in for ARM cores and IAR tool chain released

Hilden, Germany – September 10, 2008 - SEGGER Microcontroller, a leading manufacturer of middleware, debug probes and flash programming solutions for embedded systems, today announced the availability of a new task sensitive plug-in for the IAR Embedded Workbench[®] for ARM[®].

The plug-in not only visualizes the state and stack usage of all tasks and other OS-objects such as mailboxes, timers and semaphores in the system. The new version of the plug-in also allows selecting and analyzing any task. This means the register, call stack, source code and disassembly windows, which normally show the state of the running task only, switch to the state of the selected task. This works for all tasks in the system, including tasks which have not been started, the running task as well as blocked and interrupted tasks.

The new plug-in enhances the debugging and testing capabilities of embOS for ARM significantly, reducing overall development and debugging time. It works with all builds: Source code, library and trial builds. For detailed information about task sensitivity, refer to: http://www.segger.com/embos iar plugin.html

XIAR Embedded Workbench IDE						_ 🗆 ×
<u>File Edit View Project D</u> ebug Disassembly <u>R</u> DI embOS <u>I</u> ools <u>W</u> indow <u>H</u> elp						
	• 1		D 🐢 👍 🏠 📴 🕅	👂 🕁 🕁		
5 • 5 • 5 5 5 5 X						
	1	1 .	1	1	1	
* Prio Id Name	Status	Timeout	Stack Info	Run count	Time slice	Events
115 0x20A5E8 IP_RxTask 110 0x20A5A4 IP Task	Waiting (event object)	9 (98363)	296 / 512 @ 0x209D90 592 / 768 @ 0x209A90	79 8987	0/2 0/2	0×0 0×0
101 0x20A3A4 IP_Task 101 0x20A560 IP FTP Server	Waiting (event object)	9 (90303)	504 / 8192 @ 0x209A90	8770	0/2	0x0
💾 🕈 100 0x20A62C IP_FTPServerChild	Ready		1744 / 2400 @ 0x209130	8	0/2	0×0
Tidle Idle						
	🕶 🗙 Register	¥ I	Disassembly			×
RTUSINIT_AT915AMI/X256.c [US_IP_FTPServer.c] USKemic		•				<u>^</u>
1030 * 1031 * Function description	CPU Registers	<u> </u>		Memory		
1032 *	R0 =		00009154 480B 00009156 7800	LDR LDRB	RO, [PC,#0×0 RO, [RO, #0]	
1033 * OS_Deactivated() is called fr 1034 * a cooperative task switch.	R2 =		00009158 1C40 0000915A 490A	ADD LDR	RO. RO. #1	1000
1035 ×	- K3 -		0000915C 7008	STRB	R1, [PC,#0×0 R0, [R1, #0]	20]
1036 * Notes 1037 * It is important that OS_Regio		00000000 002080C4	OS_Switch(); 0000915E F7FF	; pre BL/BLX	Switch needs	to be call
1038 * It is important, that OS_Swit	R6 = 0×1	00204E64	00009160 FC95 OS_RegionCnt = r;	BL' //	<pre>?Veneer (3) A call of OS_</pre>	
1039 * and OS_UpdateAT() is called. 1040 * From that point, interrupts 1		00204D08	c) 00009162 4808	LDR	R0, [PC,#0×0	20]
1041 * This is required to ensure, t	R9 = 0×0	CCC0009	00009164 7004 OS_RESTORE_I();	STRB //	R4, [RO, #O] From now on,	interrupts
1042 * the activation of high priori 1043 */	KI0 - 0X	CCC000A	00009166 4808 00009168 7800	LDR LDRB	RO, [PC,#0×0 RO, [RO, #0]	20]
1044 void OS_Deactivated(void) { 1045 OS_U8 r;	R11 = 0×0	CCC000B	0000916A 21F0	MOV	R1, #240	
1046 r = OS RegionCnt++;	R13 (SP) = $0x$		0000916C 4208 0000916E D002	TST BEQ	RO, R1 0×009176	
1047 OS_Switch(); //		00009163 0000003F	OS_RESTORE_I(); 00009170 2099		From now on,	interrupts
1048 1049 OS_RegionCnt = r;		FFFFFFF	1	MOV /	RO, #153	
1050 OS_RESTORE_I<>; //		00009162				
1051			Call Stack			×
1053 /************************************	• R10_fiq = 0×	00000000	♦ OS_Deactivated ()			
1055 * OS_GetTime32	$R11_fiq = 0x$ $R12_fiq = 0x$		OS DeactivateP (0x0020	80C4. 'H' (0×48))	
1056 */ 1057 #ifndef OS_GetTime32	$R13_fiq = 0 \times$	00200130	OS_EVENT_Wait (0×0020	· ·	, ,	
1058 OS_I32 OS_GetTime32(void) {	$R14_fiq = 0xi$		IP_OS_WaitItemTimed(0)	
1059 OS_I32 t; 1060 OS_DI<>;			IP_OS_WaitItem (0x0020			
1061 t = (OS_I32)OS_Time;	R14_svc = 0x		t_accept (2116872, 0x00) FTPServerParentTask (8100)	
1062 OS_RESTORE_I<>; 1063 return t;			MainTask ()	/		
1064 >	R14_abt = 0x	0000000	[OS_ReturnFromTask + 0]		
1065 #endif 1066	$ \pm SPSR_abt = 0xi$					
1067/************************************	R13_irq = 0x0 R14_irq = 0x0					
1068 *		Þ				
		()				
Ready						NI /

"The new plug-in allows in-depth analysis of suspended tasks, which is especially important in complex applications with a lot of tasks. It makes the combination of embOS for ARM, IAR Embedded Workbench and J-Link the best debugging solution for embedded systems that I am aware of", says Armin Winter, embOS product manager at SEGGER.



About embOS[®]

embOS is a priority-controlled multitasking system, designed to be used as an embedded operating system for the development of real-time applications for all popular CPUs. embOS is a high-performance real time OS that has been optimized for minimum memory consumption in both RAM and ROM, as well as high speed and versatility. Nested interrupts are supported, causing zero additional interrupt latency.

embOS comes with embOSView, a run-time task-level profiling tool. embOS is provided as full source code and comes with a simple licensing model without royalties. The user manual with full product specifications and a trial version are available at:

http://www.segger.com/embos.html

About SEGGER

SEGGER Microcontroller develops and distributes hardware and software development tools as well as software components. All software components are ANSI "C" compliant and can be used in embedded systems including industries such as telecom, medical technology, consumer electronics, automotive industry and industrial automation. SEGGER software products include: embOS (RTOS), emWin (GUI), emFile (File System), emUSB (USB device stack) and embOS/IP (TCP/IP stack). Besides the highly efficient software products, SEGGER also provides embedded hardware tools such as the well-known JTAG emulator J-Link, J-Trace and the Flasher (stand alone programmer). SEGGER's intention is to cut software development time for embedded applications by offering affordable, flexible and easy-to-use tools and software components allowing developers to focus on their applications.

Contact information:

Ivo Geilenbrügge, Marketing Manager Tel: +49-2103-2878-0 E-mail: info@segger.com

Issued on behalf of:

SEGGER Microcontroller GmbH & Co. KG In den Weiden 11 40721 Hilden Germany

All product and company names mentioned herein are the trademarks of their respective owners. All references are made only for explanation and to the owner's benefit.