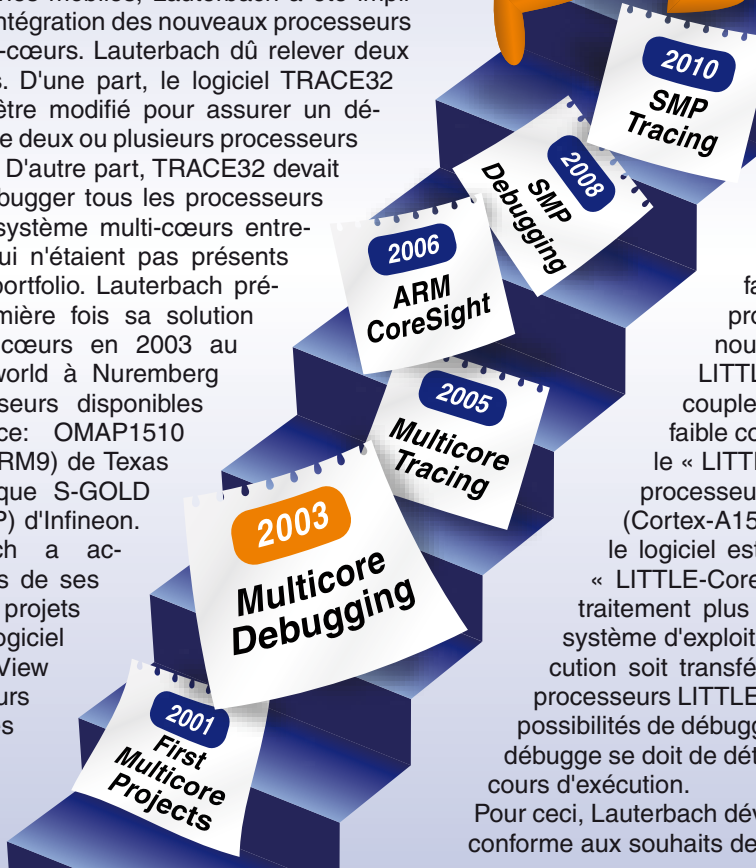




**Débugge
Multi-cœurs**



Dans le cadre de partenariats stratégiques avec les fabricants de téléphones mobiles, Lauterbach a été impliqué dès 2001 sur l'intégration des nouveaux processeurs base d'ASICs multi-cœurs. Lauterbach dû relever deux défis très différents. D'une part, le logiciel TRACE32 PowerView devait être modifié pour assurer un débogge sans conflit de deux ou plusieurs processeurs connectés en série. D'autre part, TRACE32 devait être capable de débogger tous les processeurs présents dans un système multi-cœurs entre-autres les DSPs qui n'étaient pas présents jusque-là dans le portfolio. Lauterbach présente pour la première fois sa solution de débogge multi-cœurs en 2003 au salon embedded world à Nuremberg pour deux processeurs disponibles dans le commerce: OMAP1510 (TMS320C55x et ARM9) de Texas Instruments ainsi que S-GOLD (ARM9 et OAK DSP) d'Infineon. Depuis, Lauterbach a accompagné plusieurs de ses clients dans leurs projets multi-cœurs. Le logiciel TRACE32 PowerView a été alors plusieurs fois adapté à ces systèmes à cœurs multiples qui offrent en effet de complexes



possibilités de débogge et de trace. Quels sont désormais les défis de Lauterbach pour l'année 2013 ? Beaucoup de designers de systèmes SMP souhaitent des processeurs plus puissants avec une consommation d'énergie plus faible. Pour répondre à cette problématique, ARM propose un nouveau système appelé « big.LITTLE » qui offre la possibilité de coupler un processeur ayant une faible consommation d'énergie à savoir le « LITTLE-Core » (Cortex-A7) avec un processeur puissant appelé « big-Core » (Cortex-A15). Le principe est simple : le logiciel est généralement exécuté sur le « LITTLE-Core ». Dès qu'une puissance de traitement plus importante est nécessaire, le système d'exploitation veille alors à ce que l'exécution soit transférée vers le « big-Core ». Les processeurs LITTLE- et big-Core offrent différentes possibilités de débogge et de trace. De plus, l'outil de débogge se doit de détecter quel est le processeur en cours d'exécution. Pour ceci, Lauterbach développera en 2013 une solution conforme aux souhaits de ses clients.

CONTENU NEWS 2013

Les stratégies multi-cœurs TRACE32	2	Nouveaux OS cibles supportés	7
Couverture de code	4	µTrace pour le Cortex™-M	8
Nouveaux processeurs supportés	6		
Débugge UEFI pour ARM	7		

Les stratégies multi-cœurs TRACE32

Depuis plus de 10 ans, Lauterbach offre des solutions de débogage et de trace pour les systèmes multi-cœurs.

Flexibilité

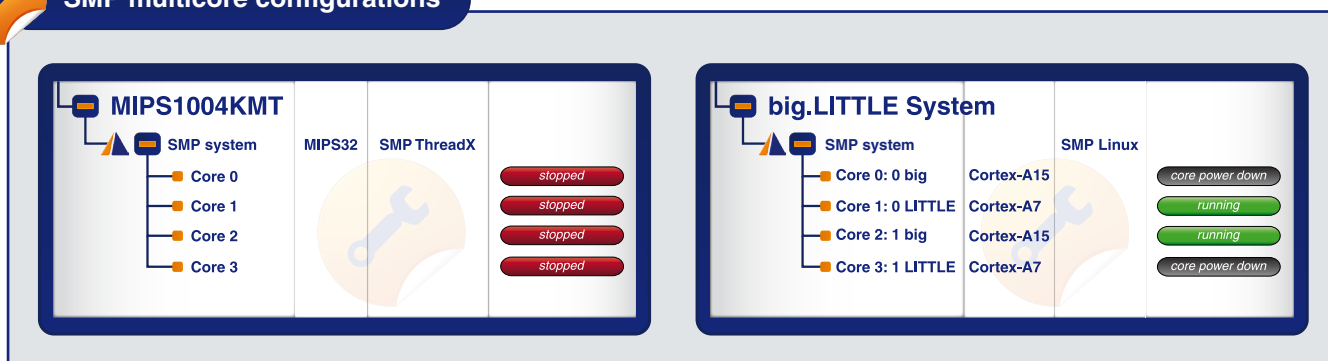
Lauterbach a toujours accordé beaucoup d'importance sur la flexibilité du logiciel et du matériel TRACE32.

Chaque combinaison de processeurs, chaque topologie multi-cœur et toute infrastructure de débogage et de trace peut être supportée par TRACE32. Cette flexibilité implique aussi en effet que TRACE32 inclut le débogage et la trace des systèmes AMP et SMP.

Les tableaux en pages 2 et 3 offrent un aperçu des différences les plus importantes entre ces deux systèmes.



SMP multicore configurations



Système SMP – Symmetric MultiProcessing

Conception du système	Un système SMP se constitue de deux ou plusieurs cœurs qui sont généralement identiques ou qui disposeraient de jeux d'instructions compatibles.
Répartition des tâches/ Système d'exploitation	Un seul système d'exploitation SMP répartit les tâches entre les cœurs (de façon dynamique ou bien statique).
Nombre des interfaces TRACE32	Une seule interface TRACE32 est utilisée pour le débogage d'un système SMP. Cette interface contrôle tous les cœurs et affiche toutes les informations correspondantes.
Démarrage et arrêt synchrone des cœurs	Tous les cœurs sont démarrés et arrêtés de façon synchrone.
Onchip Breakpoints	Les points d'arrêts on-chip sont parallèlement programmés dans les registres de débogage de tous les cœurs.
Filtre et déclencheur de trace	Les filtres et déclencheurs de trace sont parallèlement programmés dans les registres de trace de tous les cœurs.
Affichage des informations de trace	Les informations de trace peuvent être affichées pour tous les cœurs ensemble ou bien séparément pour chaque cœur.
Profilage de code	Les mesures des temps d'exécution peuvent être effectuées pour chaque cœur ou bien pour tout le système.

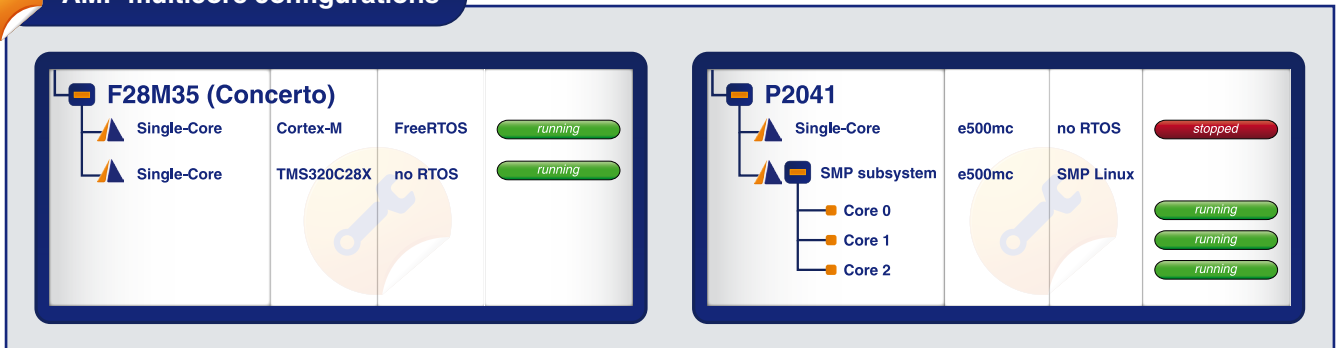
Plus de 70 architectures supportées

Un des principes de Lauterbach est d'offrir des outils pour une multitude d'architectures embarquées. Chaque nouveau cœur est intégré dans TRACE32, de façon à ce qu'il puisse être débogué comme un sous-système mono-cœur du système AMP. Le débogue SMP pour une

architecture est intégré dès que le mode multi-cœur est disponible. L'adaptation de l'OS-Awareness joue dans ce contexte un rôle particulièrement important. Il faut notamment pouvoir détecter comment le système d'exploitation assigne les tâches aux processeurs, soit dynamiquement au cours de l'exécution soit de façon totalement statique ou en partie statique.



AMP multicore configurations



Système AMP – Asymmetric MultiProcessing

Conception du système	Un système AMP se constitue de plusieurs sous-systèmes qui se composent d'un seul cœur ou bien d'un système SMP.
Répartition des tâches/ Système d'exploitation	Les tâches sont assignées aux sous-systèmes de façon statique lors de la phase de conception. Le système d'exploitation ne contrôle pas plus d'un sous-système.
Nombre des interfaces TRACE32	Deux ou plusieurs instances de TRACE32 sont démarrées pour le débogue d'un système SMP. Chaque instance contrôle un sous-système et visualise les informations correspondantes.
Démarrage et arrêt synchrone des cœurs	Tous les sous-systèmes peuvent être démarrés et arrêtés de façon synchrone (configurable).
Points d'arrêt on-chip	Les points d'arrêt on-chip sont programmés indépendamment pour chaque sous-système.
Filtre et déclencheur de trace	Les filtres et déclencheurs de trace sont programmés indépendamment pour chaque sous-système.
Affichage des informations de trace	Les informations de trace sont affichées pour chaque sous-système dans l'interface TRACE32 correspondante.
Profilage de code	Les mesures des temps d'exécution d'un sous-système peuvent être effectuées dans l'interface TRACE32 correspondante. Un timestamp global permet de visualiser la relation temporelle entre les différents sous-systèmes.

Couverture de code : génération de la documentation



Depuis novembre 2012, TRACE32 Powerview offre de nouvelles possibilités de documentation des résultats de l'analyse de couverture de code.

Ceci comprend entre-autre une fonction de commentaire ainsi que l'enregistrement des résultats en format XML.

Record

64-bit host

File on hard-disk

Review

address	tree	coverage	executed
P:00001130--000032F6	\jpeg	partial	25.026%
P:000032F8--00003899	\j_dapimin	partial	42.960%
P:000032F8--0000345F	\jpeg_createDecompress	partial	68.888%
P:00003460--0000348F	\jpeg_destroy_decompress	never	0.000%
P:00003490--0000348F	\jpeg_abort_decompress	ok	100.000%
P:000034C0--0000376B	\default_decompress_param	never	29.824%
P:0000376C--000038A7	\jpeg_consume_input	partial	58.227%
P:000038A8--00003987	\jpeg_read_header	partial	42.857%
P:00003988--000039F3	\jpeg_input_complete	never	0.000%
P:000039F4--00003A5F	\jpeg_has_multiple_scans	never	0.000%
P:00003A60--00003B9F	\jpeg_finish_decompress	partial	53.750%
P:00003BA0--00005393	\jquant2	never	0.000%

Couverture de code basée sur la trace

L'assurance qualité de produits ayant une notion de sécurité requière parfois strictement une preuve de « *Statement Coverage* » et de « *Decision Coverage* ». Pour beaucoup de systèmes embarqués, le code, hautement optimisé, doit obligatoirement être testé en temps réel. L'instrumentation du code ou bien le changement du comportement temporel du système ne sont pas autorisés.

Pour permettre à ses clients d'avoir une preuve de « *Statement Coverage* » et de « *Decision Coverage* », Lauterbach offre une couverture de code basée sur la trace. Pour cela, le processeur ou bien le système mul-

ti-cœurs utilisé doit répondre aux exigences suivantes : les cœurs utilisés doivent disposer d'une logique de trace qui produit des informations sur l'exécution code. En même temps le processeur ou système multi-cœurs doit disposer d'un port de trace avec suffisamment de bande passante pour que l'outil externe puisse récupérer, sans perte, les données produites.

Pour un débit de transmission moyen au niveau du port de trace allant jusqu'à 60 Mo/s, il est possible de transmettre les données de trace « à la volée » au cours de l'enregistrement vers le PC. Ainsi, il est possible de saisir, pour chaque test unitaire, plusieurs téraoctets d'informations de trace.

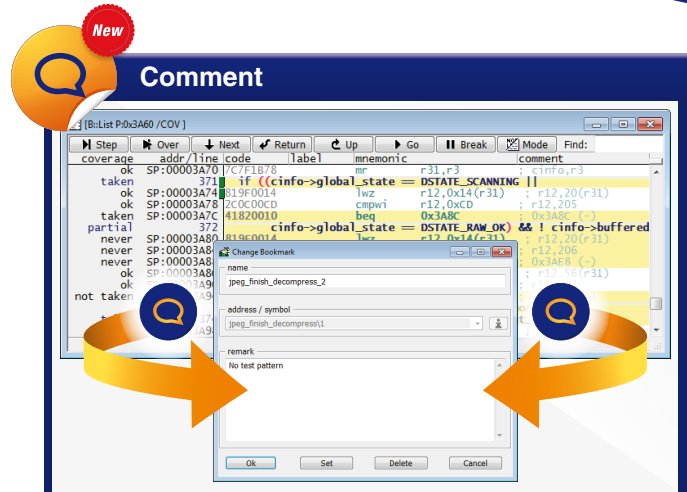
Puisque les informations de trace sont disponibles en assembleur, les preuves suivantes sont possibles :

- **Object Statement Coverage** : chaque ligne assembleur a été exécutée au moins une fois lors du test système.
- **Object Branch Coverage** : teste les branchements conditionnels : la condition a été exécutée (taken) au moins une fois ou bien n'a pas été exécutée (not taken).

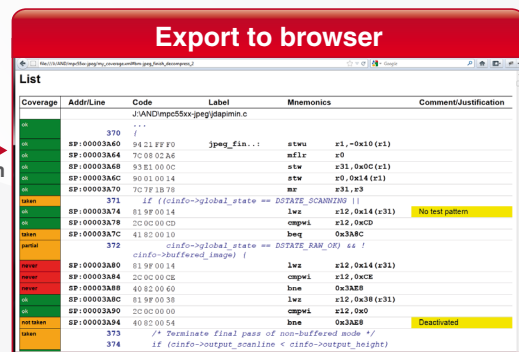
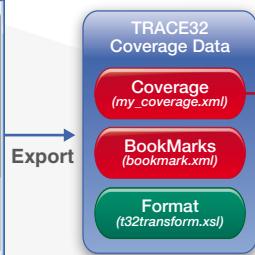
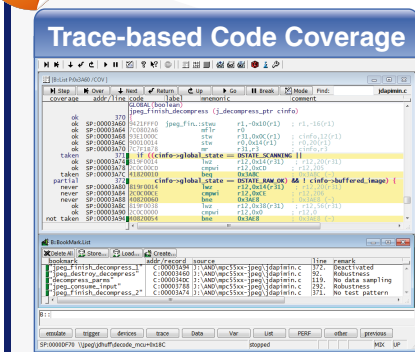
La couverture de code exécuté ainsi que la « *Decision Coverage* » pour les lignes de code haut niveau peuvent être facilement déduites à partir de cette analyse.

La fonction de commentaire

Afin de prouver qu'un système embarqué réalise correctement et complètement toutes les exigences (*Requirements*), on exécute généralement des tests



Export



unitaires sur le code permettant de constituer la base du système de test. Pour pouvoir collecter les données nécessaires pour la couverture de code, l'outil de trace enregistre au cours du test système toutes les informations sur l'exécution du code (**Record**). Les données de trace enregistrées sont alors gérées par le logiciel TRACE32 PowerView dans une base de données à partir de laquelle de nombreuses fonctions de vérification de couverture de code sont possibles (**Review**).

Le testeur doit prendre par la suite une des décisions suivantes pour toutes les instructions qui sont marquées comme « pas complètement exécutées » :

- Dans le cas où le code non exécuté doit couvrir un « *Requirement* », un nouveau test doit être mis en place afin de couvrir l'itération suivante.
- Dans le cas où le code non exécuté doit couvrir un « *Requirement* », qui ne peut pas cependant être testé dans la configuration système actuelle, la nouvelle fonction de commentaire de TRACE32 peut être utilisée afin de justifier la non-exécution de cette section de code (**Comment**).

- S'agit-il d'un « code mort » ? Dans ce cas, le code non-exécuté doit être supprimé du logiciel.

Export

Après la fin du test système, les résultats de l'analyse de couverture de code doivent bien sûr être documentés. TRACE32 inclut désormais la possibilité d'enregistrer les résultats en format XML (**Export**). Les éléments suivants peuvent être exportés :

1. Le code assembleur avec le code haut niveau ainsi que les marqueurs de l'analyse de couverture de code (my_coverage.xml).
2. Les commentaires expliquant pourquoi les sections de code non-exécutées au cours des tests sont toutefois autorisées (bookmark.xml).

Pour une représentation intuitive des résultats de couverture de code dans un navigateur Web, Lauterbach fournit un fichier de mise en forme (t32transform.xml). Si nécessaire, les résultats peuvent être enregistrés, à partir du navigateur, en format PDF.

New

Processeurs

Altera	Cortex-A/-R • Cyclone V SoC
Analog Devices	Cortex-M • ADuCM36x
AppliedMicro	PPC40x • PPC405EX, PPC405EXr PPC44x • SMP pour APM PacketPro
ARM	Cortex-A5x (ARMv8) • Cortex-A53 • Cortex-A57
Atmel	Cortex-M • ATSAM4
Axis	MIPS32 • ARTPEC-4
Broadcom	MIPS32 • BCM47186 • BCM6318, BCM6828 • BCM7346, BCM7356 • BCM7418, BCM7425
BroadLight	MIPS32 • BL25580
CEVA	CEVA-X • CEVA-XC323 TeakLite-III • CEVA-TeakLite-4
Energy Micro	Cortex-M • EFM32LGxxx, EFM32WGxxx • EFM32ZGxxx
Freescale	ColdFire+/V1 • MCF51AC/AG/CN/EM • MCF51JE/JM/MM/QE • MCF51JF/JU/QM/QU Cortex-A/-R • Vybrid F Series Cortex-M • Kinetis L • Vybrid Series MPC85XX/QorIQ e500 • P1010, P1012, P1014 • P1017, P1021, P1023 QorIQ 32-Bit • P2040, P2041 QorIQ 64-Bit • B4220, B4420, B4860 • P5021, P5040, T10XX • T2080, T2081, T4160, T4240

Freescale (cont.)	PX-Series • PxD1005, PxD1010, PxD2020 • Pxn2020, Pxn2120, Pxr40xx • Pxs2005, Pxs2010, Pxs30xx Qorivva MPC5xxx • MPC5743K, MPC5744K • MPC5744P, MPC5746M, • MPC5748G, MPC5777M S12Z • S12ZVH, S12ZVM StarCore • B4220, B4420, B4860
Hilscher	ARM9 • NETX 51
Infineon	Cortex-M • XMC4000 Family • TC2D5T/D7T, TC2D5TE/D7TE • TC275T/277T, TC275TE/277TE TriCore • TC2D5T/D7T, TC2D5TE/D7TE • TC275T/277T, TC275TE/277TE
Intel®	Atom™/x86 • Atom Z2460/CE2600/N2800 • Core i3/i5/i7 3rd Generation
Marvell	ARM11 • MV78130v6, MV78160v6 • MV78230v6, MV78260v6 Cortex-A/-R • MV78130v7, MV78160v7 • MV78230v7, MV78260v7
Mobileye	MIPS32 • EyeQ3
NEC	MIPS32 • EMMA3 Series
NVIDIA	Cortex-A/-R • Tegra 3
NXP	Beyond • JN5168 Cortex-M • LPC43xxx, LPC800
Renesas	Cortex-A/-R • R-Car H1 MIPS32 • RT3352 RH850 • RH850/E1x, RH850/F1x RL78 • RL78D1A/F1x/G1x/I1A/Lxx RX • RX630, RX631, RX63N SH • SH7267

New

Processeurs

Renesas (cont.)	V850 <ul style="list-style-type: none"> • V850E2/Fx4-L • V850E2/Mx4 Multicore
Samsung	Cortex-A/-R <ul style="list-style-type: none"> • Exynos 4212, Exynos 4412 • Exynos 5250 • S5PV210
Sigma Designs	MIPS32 <ul style="list-style-type: none"> • SMP8634, SMP8654
ST-Ericsson	Cortex-A/-R <ul style="list-style-type: none"> • DB8540 MMDSP <ul style="list-style-type: none"> • DB8540
STMicro-electronics	Cortex-A/-R <ul style="list-style-type: none"> • SPEAr1310, SPEAr1340 Cortex-M <ul style="list-style-type: none"> • STM32 F3, STM32 F4

STMicro-electronics
(cont.)

SPC5xx

- SPC56AP60, SPC56AP64
- SPC560P54, SPC560P60
- SPC574K70, SPC574K72
- SPC574L74, SPC57EM80
- SPC57HM90

Synopsys

ARC

- ARC-EM 1.1

Texas Instruments

Cortex-A/-R

- RM4 Series

Cortex-M

- F28M35 Concerto
- LM4F Series

MSP430

- MSP430FR5xx

TMS320C28X

- C28346
- F28022, F28027, F28M35

TMS320C55X

- C5535

TMS320C6x00

- C6655, C6657, C6713



Débugge UEFI pour ARM

Lauterbach a élargi en 2012 son support pour le débuge du logiciel intermédiaire UEFI qui inclut désormais :

- UEFI BIOS InsydeH2O pour Atom et x86
- UEFI BIOS Intel BLDK pour Atom et x86 New
- UEFI BIOS TianoCore pour ARM/Cortex New

Le débuge UEFI est activé dans TRACE32 par le chargement de ce qu'on appelle une extension TRACE32. Plus d'informations concernant le débuge UEFI sont disponibles à l'adresse suivante : www.lauterbach.com/uefi.html

Intégration de systèmes d'exploitation

- FreeRTOS pour Beyond et Coldfire
- Linux pour Beyond et x86 64-bit
- OSEK/ORTI SMP
- QNX pour x86
- Quadros pour CEVA-X
- RTX-ARM v4
- SMX pour Coldfire
- SYS/BIOS pour TMS320C6x00
- VxWorks pour x86
- µC/OS-II pour TMS320C28X
- µC/OS-III pour SH

New

Systèmes d'exploitation

DEOS pour PowerPC	disponible
Linux pour ARMv8 (64-Bit)	prévu
OKL4 5.0 pour ARM	disponible
Windows Standard (XP, Vista, Win7, Win8) pour x86 32/64-Bit	prévu
µT-Kernel pour ARM	disponible



Venez nous rendre visite pour voir toutes les nouveautés Lauterbach :

Paris - Porte de Versailles - Stand D7

Parc des Exposition de Tarbes



µTrace pour le Cortex™-M

Lauterbach développe actuellement un nouvel outil de débogue et de trace ultra compétitif ciblant uniquement la famille de microprocesseurs Cortex-M. Il sera disponible à la vente aux cours du deuxième trimestre 2013. Le choix des architectures Cortex-M est lié à l'importance que prend aujourd'hui ce type d'architecture sur le marché de l'embarqué. Le nouvel outil Lauterbach appelé « µTrace » est donc une solution compacte intégrant toutes les fonctionnalités très haut de gamme « made in » Lauterbach :

Les Caractéristiques du µTrace

- Support pour plus que 1000 différents processeurs de la famille Cortex-M
- Interface USB 3 vers le PC
- Interfaces JTAG, Serial Wire Debug et cJTAG
- 256 MByte de mémoire de trace
- Connecteur Half-Size à 34 broches vers le matériel cible. Des adaptateurs pour plusieurs autres types de connecteurs sont disponibles.
- Plan de tension 0.3 V - 3.3 V, tolérant les 5 V

Les Caractéristiques de débogue

- Débogue C/C++
- Points d'arrêt simples et avancés
- Écriture et lecture de la mémoire au cours de l'exécution

- Programmation de la mémoire Flash
- Débogue OS-aware
- Débogue multi-cœurs de deux ou plusieurs cœurs Cortex-M

Les Caractéristiques de trace

- Trace ETMv3 à 4-bit en mode continu
- ITM via TPIU et Serial Wire Output
- Trace multi-cœurs
- Trace infinie par la diffusion des informations de trace « à la volée » vers le PC
- Analyse des temps d'exécution pour les tâches et les fonctions
- Analyse de couverture de code
- Évaluation des données de trace lors de l'enregistrement (Real-time Profiling)
- Mesure d'énergie grâce à la sonde Analog Probe intégrée

Le µTrace sera contrôlé, comme tous les autres produits Lauterbach, par l'interface graphique TRACE32 PowerView.



INFORMEZ NOUS

Votre adresse e-mail a changé ou vous ne recevez plus notre bulletin d'information, dites-le nous à l'adresse suivante :

info_fr@lauterbach.fr

