

Lauterbach APIs

Lauterbach stellt mit seinen APIs Software-Schnittstellen zur Verfügung, die es erlauben:

- ein TRACE32-Entwicklungswerkzeug durch eine externe Applikation zu steuern.
- den Funktionsumfang eines TRACE32-Entwicklungswerkzeugs mittels einer externen Applikation zu erweitern.

Im Folgenden werden die einzelnen TRACE32 APIs kurz vorgestellt.

TRACE32 API

Die TRACE32 API ist die Standard-API zur Steuerung eines TRACE32-Entwicklungswerkzeugs durch eine externe Applikation. Folgende funktionspezifische APIs sind Teil der TRACE32 API:

- Die **Visual Basic API** stellt eine vorcompilierte DLL zur Verfügung, die es ermöglicht, ein TRACE32-Entwicklungswerkzeug über ein Visual Basic Programm zu steuern.
- Die **FDX API** stellt C-Bibliotheken bereit, um einer externen Applikation, die auf dem Host läuft, einen sehr schnellen Datenaustausch mit der Applikation auf dem Zielsystem zu ermöglichen. Das TRACE32-Entwicklungswerkzeug dient dabei als Kommunikationsschnittstelle.
- Die **JTAG API** stellt C-Bibliotheken zur Verfügung, die einer externen Applikation erlauben über eine TRACE32-ICD Debugger Hardware direkt mit einem JTAG TAP-Controller im Zielsystem zu kommunizieren. Damit kann man ohne eigenes Hardware-Interface beispielsweise Boundary Scan Testprogramme oder Basisfunktionen eines Debuggers für einen weiteren Core in der JTAG Kette des Mikrocontrollers implementieren.

Die folgenden APIs stellen eigenständige Software-Schnittstellen zur Verfügung:

Simulator API

Die Simulator API erlaubt es, für die von Lauterbach angebotenen Instruktion Set Simulatoren Timer, externe Kommunikationsschnittstellen und andere Peripheriekomponenten zu definieren. So kann der Instruktion Set Simulator auch für den Test von Programmen eingesetzt werden, die Timer bedienen oder über Kommunikationsschnittstellen Daten austauschen.

Communication API

Für eine ganze Reihe von Mikrocontroller bietet Lauterbach ROM Monitor-basierte Debugger an. Diese Debugger kommunizieren mit dem im Zielsystem installierten ROM Monitor über eine RS232 Schnittstelle bzw. über die Schnittstelle einer Lauterbach EPROM Simulator Hardware.

Die Communication API erlaubt nun, einen ROM Monitorbasierten Debugger durch eine externe Applikation so zu erweitern, dass er auch über andere Kommunikationsschnittstellen wie beispielsweise Ethernet oder CAN betrieben werden kann.

Kernel Awareness API

TRACE32-Entwicklungswerkzeuge unterstützen standardmäßig die meisten auf dem Markt gängigen Echtzeit-Betriebssysteme. Hauptfunktionen dieser Unterstützung sind:

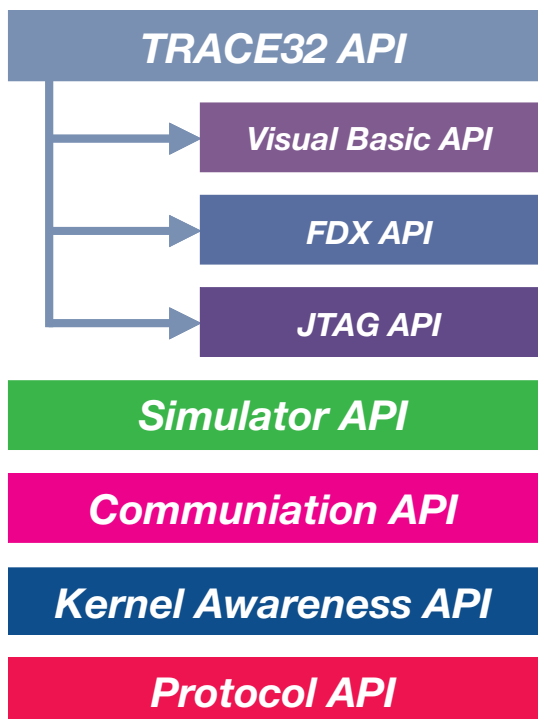
- Anzeige von Betriebssystem-Ressourcen während des Debugging
- Umfassende Möglichkeiten zur Analyse des Laufzeitverhaltens der einzelnen Tasks

Die Kernel Awareness API erlaubt den Nutzern eines TRACE32-Entwicklungswerkzeugs diese Betriebssystemunterstützung auch an eigene, proprietäre Echtzeit-Betriebssysteme anzupassen.

Protocol API

Die TRACE32-Produkte PowerProbe und PowerIntegrator ermöglichen die Aufzeichnung des Zeitverhaltens beliebiger Kommunikationsschnittstellen sowie beliebiger Busprotokolle. Die Auswertung und Darstellung wichtiger Kommunikations- und Busprotokolle wird von der TRACE32-Software bereits unterstützt.

Die Protocol API ermöglicht die Umsetzung von Roh-Tracedaten in höhere Protokolle über eine externe Applikation auch dann, wenn die TRACE32 Software noch keine Anpassung für das Kommunikations- bzw. Busprotokoll bietet. So kann das im Trace aufgezeichnete Zeitverhalten intuitiv und schnell analysiert werden.



Andrea Martin, Februar 2005