

Produktübersicht

Kithara RealTime Suite

Die Echtzeitlösung für Automatisierung, Kommunikation, Bildverarbeitung, Automotive und Datenspeicherung

NEU!

Timer Module

Hochgenaue Ermittlung der Systemzeit sowie Kurzzeitverzögerungen: [Seite 15](#)

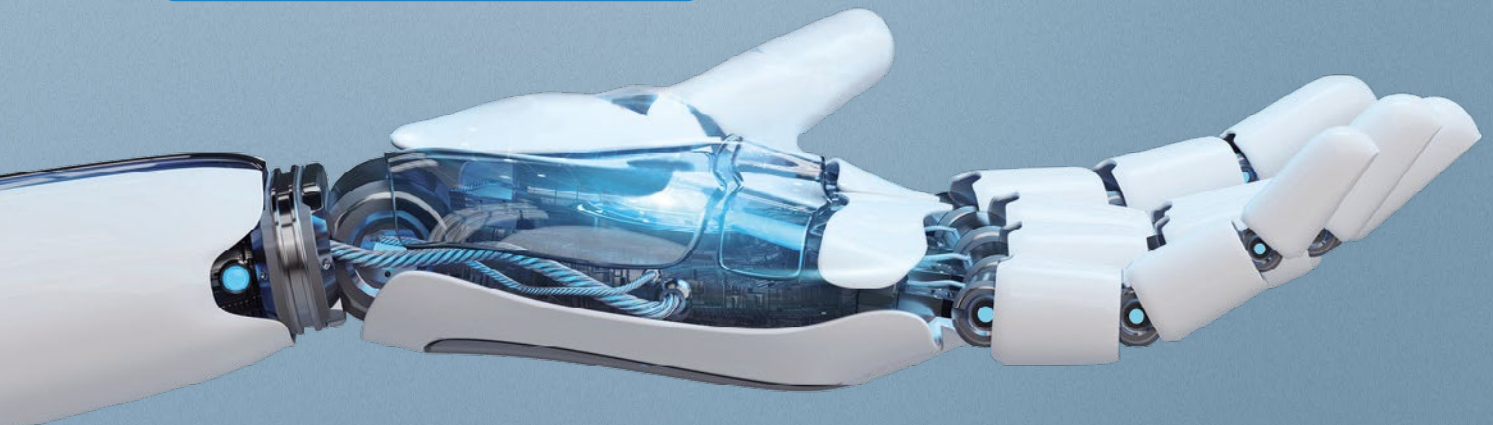
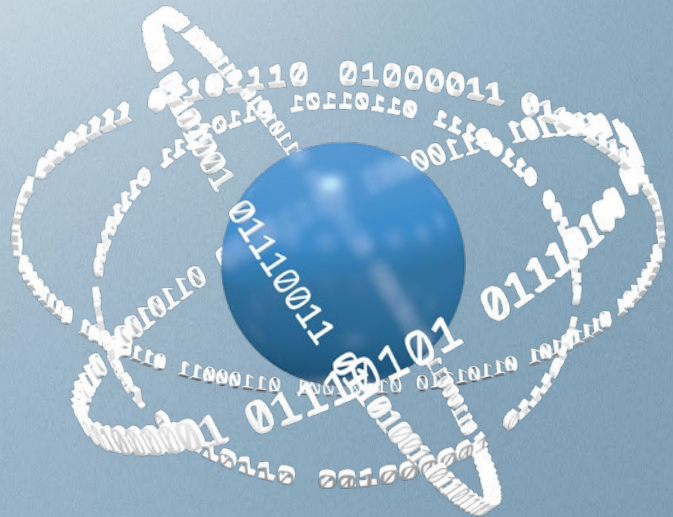
RealTime Tasking Module

Hochfrequente Echtzeit-Timer-Routinen und prioritätsgesteuertes, präemptives Echtzeit-Multitasking: [Seite 16](#)

Timer
Module



RealTime
Tasking
Module





Präzises Timing

Echtzeit für Windows

In der modernen Welt ist Zeit ein entscheidender Faktor. Automatisierte Abläufe auf industriellem Niveau erfordern eine präzise Kommunikation zwischen Anwendung und Maschine. Hochleistungsgeräte erzeugen enorme Datenmengen, die sofort verarbeitet werden müssen.

Kithara entwickelt Software zur mikrosekundengenauen Steuerung, Überwachung und Auswertung von Prozessen für Industrie und Forschung. Unsere Produkte finden sich beispielsweise in der Automobilbranche, der Luftfahrtindustrie, bei Nahrungsmittelherstellern sowie im Gesundheitswesen. Auch zum hochaktuellen Thema Autonomes Fahren liefert Kithara einen wichtigen Beitrag.

Das Problem: Betriebssysteme wie Microsoft Windows entziehen laufenden Anwendungen häufig die Priorität, um andere Programme und Prozesse, etwa zur Systemwartung, auszuführen. Dadurch kommt es zu Unterbrechungen im Ablauf der Anwendungen.

Kithara schafft einen eigenen, geschützten Bereich, der vom Betriebssystem unabhängig ist und somit nicht gestört wird.

Kithara RealTime Suite

Windows-Echtzeiterweiterung für PC-gestützte Automatisierung

Kithara RealTime Suite ist eine modulare Echtzeiterweiterung für Windows, die hardwarenahe Programmierung, Kommunikation, Automatisierungsprotokolle und Bildverarbeitung in einem leistungsfähigen Echtzeitsystem vereint. Durch nahtlose Integration der Module erhalten Sie alle benötigten Komponenten wie aus einem Guss.

Die leistungsfähigen Echtzeit-Ethernet-Treiber sind Grundlage für die Socket-Kommunikation über TCP und UDP, einschließlich der Erfassung von Bildern mit GigE-Vision-kompatiblen Kameras, sowie für moderne Industrial-Ethernet-Protokolle. Die Software wird durch Echtzeittreiber für CAN- und UART-Interfaces ergänzt. Zu den unterstützten Automatisierungsprotokollen gehört eine umfassende EtherCAT®-Master-Implementierung inklusive Distributed Clock, Safety-over-EtherCAT®, Hotplug-Fähigkeit und Kabelredundanz.

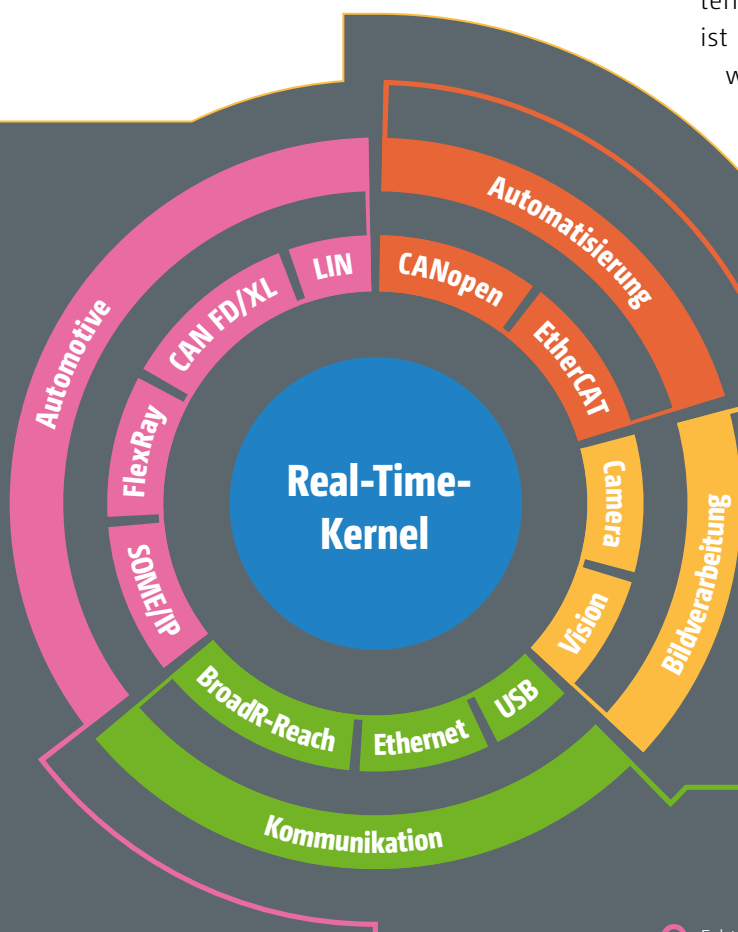
Außerdem wird ein CANopen®-Master über verschiedene CAN-Interfaces bereitgestellt, der auch in EtherCAT integriert werden kann.

Die hardwarenahe Programmierung erlaubt direkte I/O- und Speicherzugriffe, eine Interruptbehandlung und ein komfortables High-Level-Interface für Multifunktionskarten.

Zur Weiterverarbeitung der über GigE Vision oder USB3 Vision® erfassten Bilddaten innerhalb des Echtzeitsystems stehen umfangreiche Bildverarbeitungsbibliotheken wie Halcon oder OpenCV zur Verfügung.

Für schnelle Echtzeitdatenspeicherung werden SSDs mit NVMe-Schnittstelle (auch im RAID-0-Verbund) unterstützt. Datensätze lassen sich dabei auch unmittelbar in hierarchisch strukturierten Dateien im MDF- oder PCAPng-Format speichern.

Die Software ist einsteigerfreundlich und intuitiv. Durch die Benutzung der gewohnten Programmiersprache und -umgebung ist Kithara RealTime Suite auf einfache Anwendbarkeit ausgelegt. Das Betriebssystem Windows dient gleichzeitig als Entwicklungsplattform und Host-System zur Ausführung, ein gesondertes Echtzeitsystem erübrigt sich. Dadurch werden Zeitaufwand und Entwicklungskosten minimiert.



Wichtige Standards für Produktionsautomatisierung und Qualitätskontrolle | Seite 20

Funktionen für Industriekameras und Bildverarbeitungsbibliotheken | Seite 28

Unterstützung aller relevanten Hardwarechnittstellen | Seite 36

Echtzeitanbindung von modernen Automobil-Bussystemen | Seite 50

Unsere Produkte für Ihr Echtzeitprojekt

Windows eignet sich optimal für die Realisierung von industriellen Anwendungen, denn Windows ist ...

- populär: Eine weitverbreitete, intuitive und vertraute Benutzeroberfläche
- modern: Zugriff auf neueste Kommunikationsschnittstellen
- produktiv: Leistungsfähige Entwicklerwerkzeuge
- günstig: Niedrige Einarbeitungs- und Trainingskosten
- zukunftssicher: Langlebig durch hohen Marktanteil und ständige Weiterentwicklung

Die bei Windows fehlende Echtzeitfähigkeit wird durch Kithara RealTime Suite ergänzt und ermöglicht erst die Ausführung zeitkritischer Anwendungen durch ...

- deterministisches Verhalten
- extrem niedrige maximale Antwortzeiten
- gewohnte Programmiersprache und Tools
- leistungsfähige und einfach anwendbare Funktionen
- schnelle Einarbeitung und somit kostengünstige Entwicklung
- breite Unterstützung industrieller Protokolle und Standards

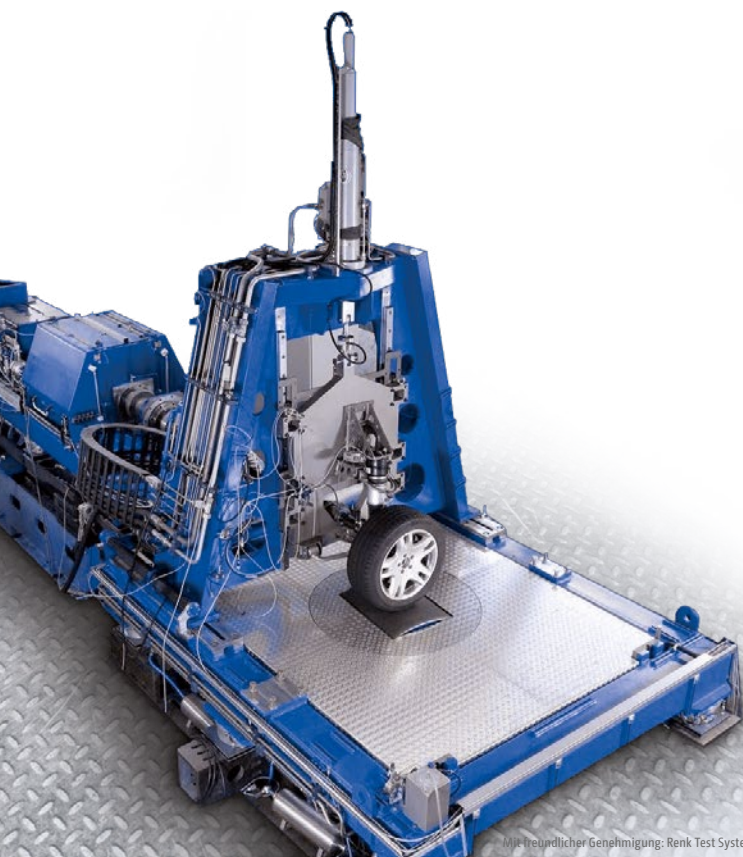
Kithara RealTime Suite verwirklicht Echtzeitsteuerung und Visualisierung innerhalb desselben Systems. Der Echtzeitkern Ihrer Anwendung läuft auf eigenen CPU-Kernen, deren Anzahl sich frei konfigurieren lässt. Auf den verbleibenden CPU-Kernen übernimmt Windows weiterhin die Visualisierung und Nutzerinteraktion. So laufen beide Bereiche unabhängig voneinander, ohne sich zu beeinflussen.

Die Kombination aus Windows und Kithara RealTime Suite erlaubt die Realisierung sehr vielfältiger Lösungen, unter anderem in folgenden Anwendungsgebieten:

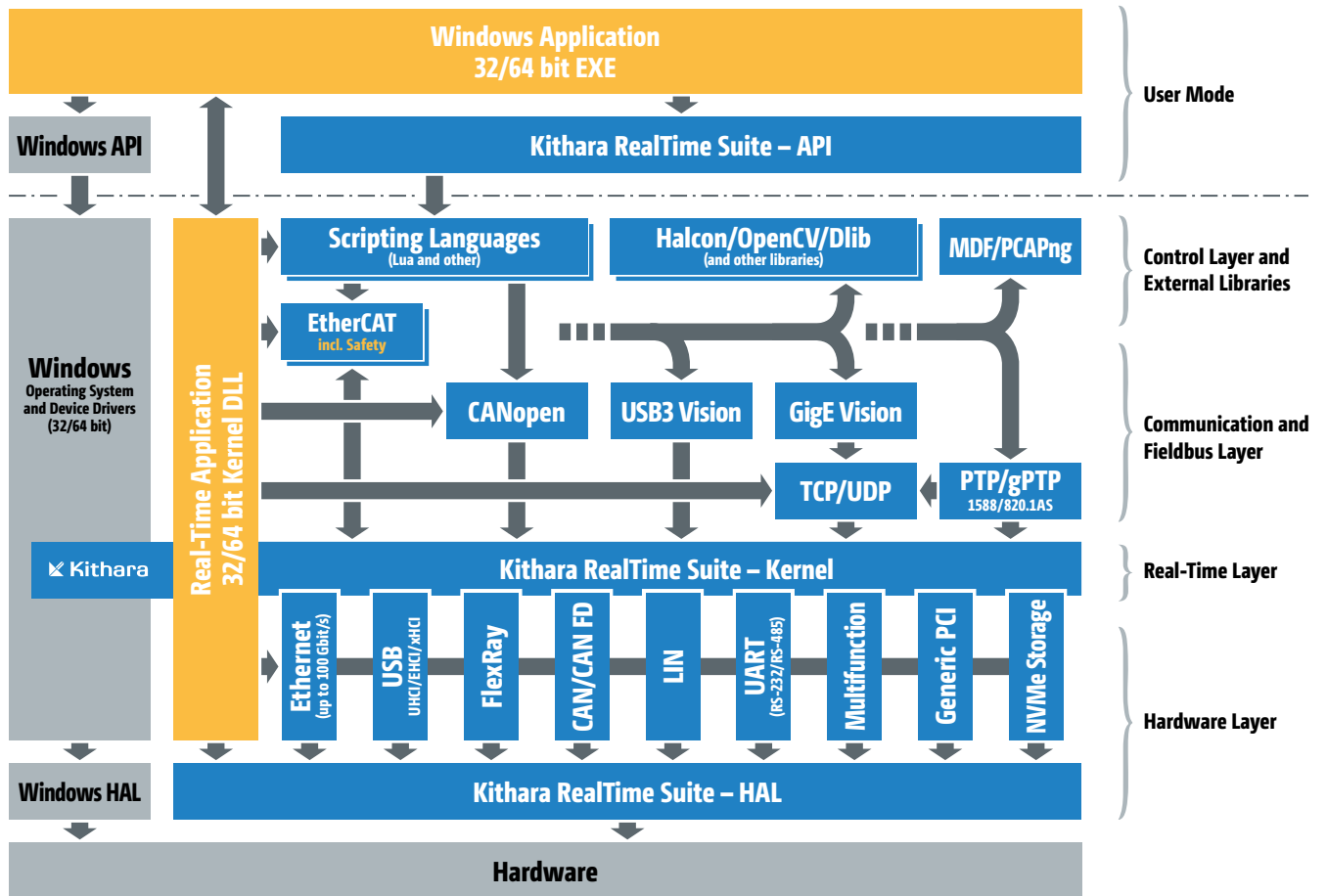
- Maschinenbau, Sondermaschinen und Prüfstände
- Produktionsautomatisierung und Qualitätskontrolle
- Laborautomatisierung und mobile Anwendungen
- Mess- und Prüfplätze im Fahrzeugbau und der Luftfahrtindustrie
- Medizinische Geräte
- Labore in Wissenschaft und Forschung

Fazit

Kithara RealTime Suite vereinfacht Ihr Projekt erheblich. Sie benötigen kein gesondertes Echtzeitsystem, somit entfällt eine aufwendig herzustellende Kommunikation zwischen verschiedenen Systemen. Stattdessen nutzen Sie die Vorteile einer homogenen Lösung und profitieren von einem erheblich vereinfachten Entwicklungsaufwand, einheitlichen Programmierwerkzeugen und Toolchain. Das verkürzt die Produkteinführungszeit, senkt die Kosten und führt zu einem niedrigeren Ressourcenverbrauch. Somit sind Kithara RealTime Suite und Windows die ideale Basis für erfolgreiche Echtzeitlösungen!



Das folgende Schema zeigt anhand eines Schichtmodells, wie Kithara RealTime Suite mit den verschiedenen Software- und Hardwarekomponenten eines Systems verknüpft ist.



- ① Kommunikation und Synchronisation zwischen der Anwendung und dem Echtzeitteil erfolgen über Shared Memory, Pipes, Mailslots, Sockets, Events oder Semaphore.
- ② Der EtherCAT-Master ist eine leistungsfähige Industrial-Ethernet-Steuerungslösung für vielfältige Automatisierungsaufgaben. Auch Feldbus-Protokolle, wie etwa ein CANopen-Master, werden unterstützt.
- ③ Bilddaten von GigE-Vision- und USB3-Vision-Kameras können in Echtzeit empfangen und mit Halcon und anderen Bibliotheken verarbeitet werden.
- ④ Die prioritätsgesteuerte preemptive Realtime-Multitasking-Umgebung erlaubt die Zuweisung der dedizierten CPU-Kerne und stellt hohe Frequenzen bereit, kombiniert mit einem geringen Jitter.

Wie funktioniert Echtzeit unter Windows eigentlich?

Für wen Echtzeit kein Fremdwort ist, weiß, wie wichtig diese heute für viele informationstechnische Bereiche ist. Ob bei parallel operierenden Servomotoren eines Roboterarms, präzisen Mess-Applikationen oder kameragestützten Fahrassistenzsystemen – die Reaktionszeit zwischen Hardware- und Software-Komponenten ist in vielen modernen Anwendungen ein kritischer Faktor. Was früher mit speziellen Echtzeitbetriebssystemen (Real-Time Operating System, kurz: RTOS) oder Hardware (DSP bzw. FPGA) erreicht wurde, lässt sich mittlerweile ebenso auch auf handelsüblichen Windows-PCs umsetzen. Windows ist jedoch kein Echtzeitsystem, da es laufende Anwendungen häufig unterbricht, um andere Programme und Prozesse, etwa zur Systemwartung, auszuführen. Wie können diese vermeintlichen Gegensätze also zusammengebracht werden? Wie funktioniert Echtzeit unter Windows?

Echtzeit und Windows – die Lösung

Die einzige Voraussetzung für Echtzeit auf Windows-PCs ist ein Prozessor mit mindestens zwei Kernen. Die grundlegende Funktionsweise besteht darin, Windows anzuweisen, nur eine begrenzte Anzahl von CPU-Kernen eines Mehrkernprozessors zu benutzen. Auf den freien Kernen bootet daraufhin das Echtzeitsystem, welches fortan wie ein eigenständiges RTOS funktioniert, während Windows auf den

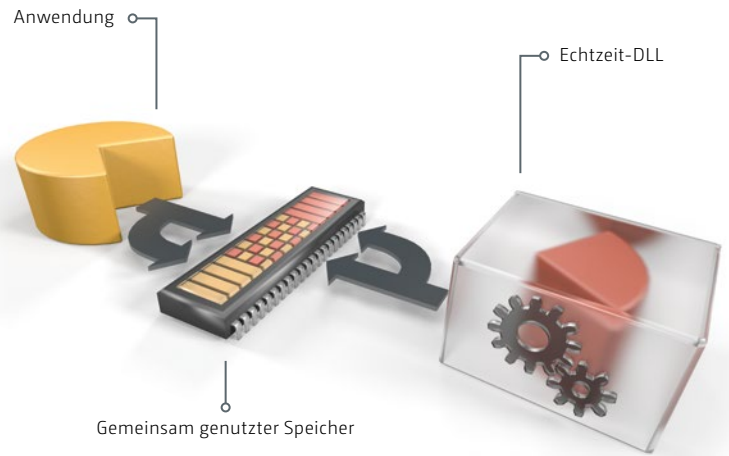
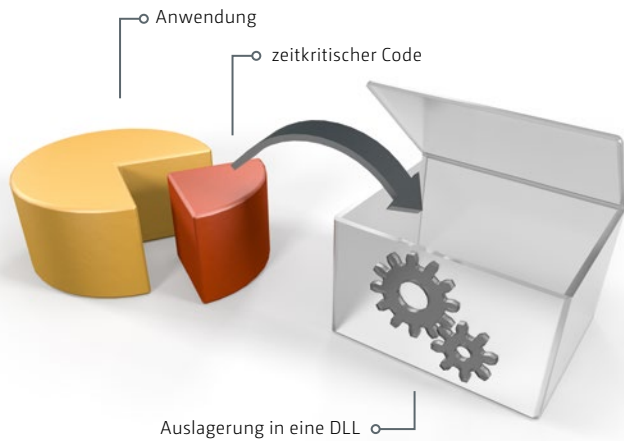
übrigen CPU-Kernen seine volle Funktionalität behält. Windows und das Echtzeitsystem laufen ab diesem Zeitpunkt also gleichzeitig, parallel auf einem einzigen Rechner, ohne dass sie sich gegenseitig einschränken. Die Anweisung an Windows, auf weniger Kernen zu booten, kann relativ einfach mit dem Konfigurationsprogramm Msconfig vorgenommen werden. Eine Anleitung dazu finden sie hier: [Setting up Dedicated CPUs](#).

Durch diese Vorgehensweise erhält das Echtzeitsystem einen geschützten Bereich und garantiert somit, dass Windows-Prozesse nicht die Priorität über zeitkritische Operationen bekommen und sich negativ auf die Echtzeitleistung auswirken. Ziel ist es, speziell „harte“ Echtzeiteigenschaften auch auf Windows-PCs zu erreichen. Doch was bedeutet „hart“ in diesem Zusammenhang?

Was ist der Unterschied zwischen „weicher“ und „harter“ Echtzeit?

Bei Echtzeit unterscheidet man zwischen unterschiedlichen Anforderungen und wie mit dem Zeitrahmen zur Ausführung von Operationen umgegangen wird. Während „weiche“ Echtzeit lediglich einen ungefähren Mittelwert an Reaktionszeiten anstrebt, wird bei der „harten“ Echtzeit garantiert, dass eine vorgeschriebene Zeitspanne nicht überschritten wird. Durch den erhöhten Anspruch ist das Erreichen von „harten“ Echtzeiteigenschaften damit wesentlich komplexer und aufwendiger. Jedoch ist dieses deterministische Zeitverhalten vor allem in automatisierten industriellen Bereichen, speziell bei Mess-, Steuer- und Regelungsanwendungen, unabdingbar.





Wie können Windows- und Echtzeit-Kontext nun aber zusammengebracht werden?

Das Echtzeitsystem ist als ein Gerätetreiber implementiert und stellt ein eigenes API bereit, mit dem der Anwender die zeitkritischen Code-Teile seiner Windows-Applikation in eine DLL auslagert. Diese DLL wird dann in den Echtzeitkontext des RTOS geladen. Damit Windows und DLL aus ihren unterschiedlichen Kontexten miteinander kommunizieren, kann – neben indirekten Mitteln wie Pipes oder Sockets – ein geteilter Speicherbereich erstellt werden, auf den beide Zugriff haben. Trotz der Verwendung eines Echtzeitbetriebssystems bleibt Windows dabei als Programmieroberfläche und Host-System zur Ausführung erhalten.

Warum gerade Windows als RTOS?

Gegenüber sowohl reinen Echtzeitbetriebssystemen als auch Echtzeit-Hardwarelösungen bietet ein Windows-PC spezifische Vorteile. So ist Windows eine vertraute Grafik- und Benutzeroberfläche mit breiter, stetig aktualisierter Treiberunterstützung sowie einem großen Spektrum an lauffähigen Programmen. Reine RTOS hingegen sind stark spezialisiert und dadurch oft eingeschränkt, was Funktionsumfang und Bedienbarkeit angeht. Hardware-Lösungen wie FPGAs oder DSPs können zwar je nach Anforderung angepasst werden, deren Programmierung ist jedoch auch komplex und zeitaufwändig. Um „harte“ Echtzeit auch unter Windows umsetzen zu können, benötigt man lediglich einen Dual-Core-PC.

Zusammenfassung

Windows eignet sich sehr gut für industrielle Anwendungen durch:

- weitverbreitete, intuitive und vertraute Benutzeroberfläche
- Zugriff auf neueste Kommunikationsschnittstellen
- leistungsfähige Entwicklerwerkzeuge
- niedrige Einarbeitungs- und Trainingskosten
- Langlebigkeit durch hohen Marktanteil und ständige Weiterentwicklung

Die bei Windows fehlende Echtzeitfähigkeit wird durch Echtzeitsysteme wie Kithara Real-Time Suite ergänzt und ermöglicht erst die Ausführung zeitkritischer Anwendungen durch:

- deterministisches Verhalten dank hochfrequenter Echtzeit-Timer
- extrem niedrige maximale Antwortzeiten
- gewohnte Programmiersprache und Tools
- leistungsfähige und einfach anwendbare Funktionen
- schnelle Einarbeitung und somit kostengünstige Entwicklung
- breite Unterstützung industrieller Protokolle und Standards

Windows und Echtzeit stellen keinen Widerspruch dar. Erst das nahtlose Zusammenspiel beider Welten ermöglicht optimale Lösungen und macht den Einsatz in industriellen Bereichen oft überhaupt erst möglich.

Möchten Sie sich selber ein Bild von Echtzeit unter Windows machen? Testen Sie einfach die **kostenfreie Demoversion** von Kithara RealTime Suite mit vollem Funktionsumfang und zahlreichen Beispielen, die Ihnen den Einstieg erleichtern.

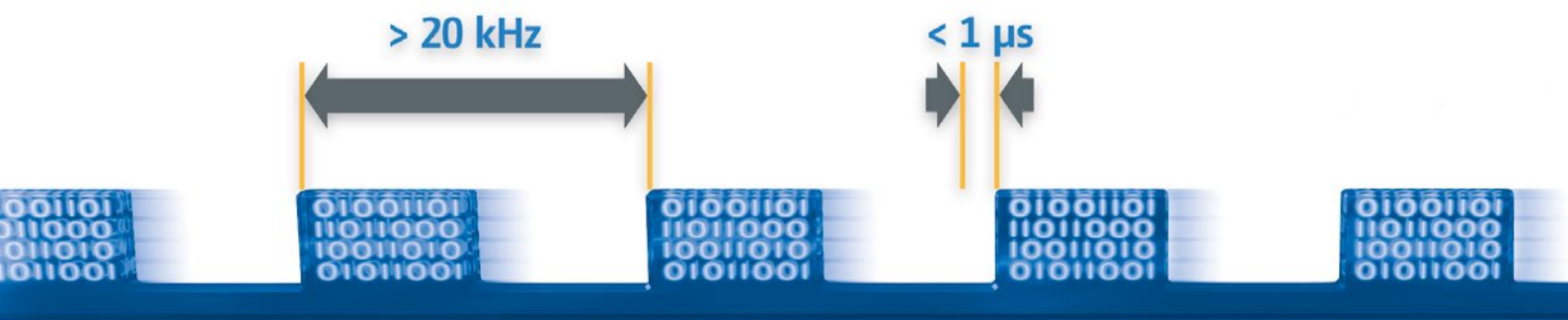
„Harte“ Echtzeit für die Windows-Plattform

Aufgrund des modular aufgebauten Systems mit seinen breit gefächerten Anwendungsmöglichkeiten ist Kithara RealTime Suite die optimale Basis zur Entwicklung leistungsfähiger Applikationen für Automatisierung, Hardwarekommunikation und industrielle Bildverarbeitung, da diese Bereiche zumeist von zeitkritischen Anwendungen abhängig sind.

Kithara RealTime Suite nutzt den sicheren Dedicated Mode, dabei werden einzelne logische CPUs (CPU-Kerne) völlig autonom und ohne jegliche Windows-Beeinflussung betrieben. Mit entsprechender Hardware lassen sich auf diese Weise präzise zyklische Timeraufrufe mit

Frequenzen im hohen zweistelligen Kilohertzbereich realisieren, bei Abweichungen (Jitter) von nur wenigen Mikrosekunden.

Beim Echtzeitstart analysiert das System zunächst die Hardware, beispielsweise alle verfügbaren Zeitgeber, und kalibriert sie. Doch es sind nicht nur einzelne Timer-Routinen mit hoher Frequenz ausführbar. Das preemptive Echtzeit-Multitaskingsystem stellt Tasks (Threads) mit bis zu 255 Prioritätsstufen bereit, so dass jeweils die Aktion mit der höchsten Priorität zuerst ausgeführt wird. Weniger wichtige Aktionen werden sofort unterbrochen und erst nach Beendigung der höher priorisierten Aktion fortgesetzt.

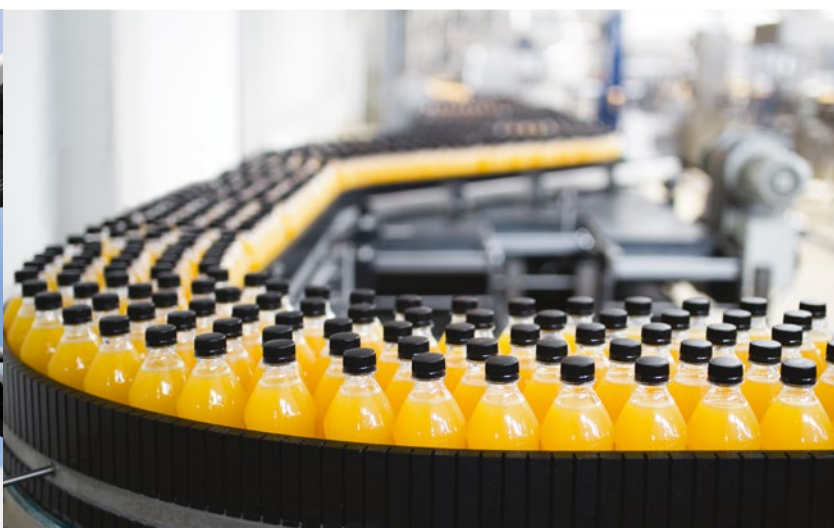


Zur Synchronisation zwischen Kernel-DLL und der Windows-Anwendung werden Events, Semaphore, Mutexe, Daten- und Message-Pipes, Sockets sowie Shared Memory zur Verfügung gestellt. Die einzelnen Tasks lassen sich gezielt den verschiedenen logischen CPUs zuordnen, um Leistung und Integrität spezieller Tasks sowie die Skalierbarkeit des Gesamtsystems weiter zu optimieren.

Um den Echtzeitkern von Kithara RealTime Suite gruppieren sich verschiedene weitere Module zur Anbindung externer Geräte und Systeme über unterschiedlichste Kommunikationsverbindungen und Zugangswege sowie für spezielle Aufgaben. So gibt es dank Storage Module schnelle Datenspeicherung auf SSDs

mit einer Dauertransferrate von mehreren Gigabyte pro Sekunde. Die erfassten Messdaten lassen sich auch im zukunftssicheren MDF-Format in Terabyte-großen Dateien speichern.

Mit Kithara Kernel Tracer und Kithara Performance Analyzer stehen außerdem leistungsfähige Diagnose- und Programmierwerkzeuge zur Verfügung, mit denen der Entwicklungsprozess noch flexibler und effizienter gestaltet werden kann.



Kundenspezifische Software-Zusammenstellung

Stellen Sie sich Ihren eigenen, maßgeschneiderten Custom Driver entsprechend den benötigten Funktionen einfach selbst zusammen oder lassen Sie sich kostenlos von uns beraten. Dieser Custom Driver bietet Ihnen so die höchste Flexibilität und hat folgende Eigenschaften:

- Kithara RealTime Suite ist modular aufgebaut und ermöglicht ein individuell maßgeschneidertes Echtzeitbetriebssystem
- Sie können für die Entwickler- und Runtime-Dateien einen eigenen Namen festlegen.
- Stabiles API – keine Anpassung bei neuen Versionen erforderlich.
- Mit dem Erwerb der Module erhalten Sie automatisch eine Entwicklerlizenz.
- Günstige Staffeldingungen für Runtime-Lizenzen je nach Bedarf.
- Mehrere Updates über einen Zeitraum von 12 Monaten nach Ersterwerb
- Update-Service jeweils um weitere 12 Monate verlängerbar.
- Jederzeit um zusätzliche Module erweiterbar.
- Einmalig 10 Stunden Entwickler-Support inklusive, darüber hinaus verschiedene Support-Pakete erhältlich (20, 50, 100 Stunden)
- **Support** wird im übersichtlichen Kithara-Ticketsystem bearbeitet, ermöglicht schnellen und direkten Kontakt mit Entwicklern

Unser qualifiziertes Support-Team garantiert Ihnen den bestmöglichen Service bei der Einbindung der Software in Ihr Projekt und unterstützt Sie durch unseren Entwickler-Support (auch über Kithara-Software hinaus).

Plattformen

Echtzeitfähigkeit wird nur auf der Kernel-Ebene erzielt. Dazu wird eine Programmiersprache und -umgebung benötigt, die nativen Maschinencode erzeugt, wie etwa C/C++, Delphi und andere. Dennoch unterstützt Kithara RealTime Suite verschiedene Plattformen, beispielsweise auch die .NET-Umgebung. Die Lösung besteht darin, den zeitkritischen Code in eine DLL zu verlagern, die mit den Funktionen von Kithara RealTime Suite direkt auf die Kernel-Ebene geladen wird und dadurch in den Echtzeit-Kontext gelangt. Sofort verwendbare Programmgerüste für die genannten Plattformen befinden sich in jeder Software-Lieferung.

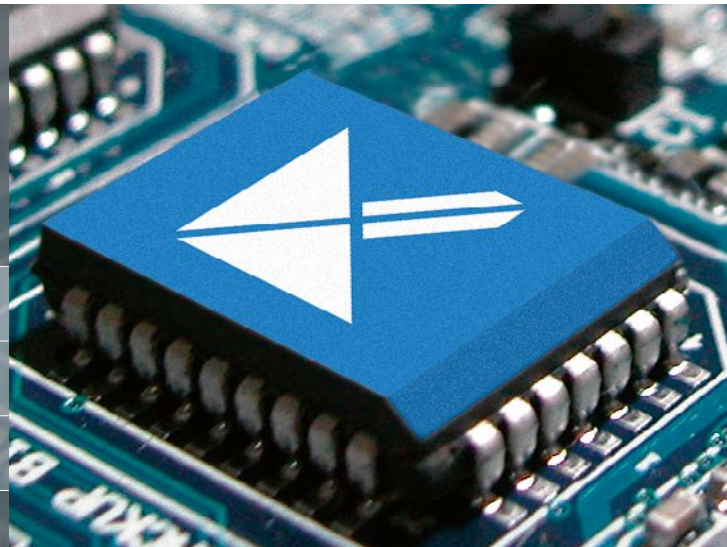
Betriebssystem

Die Software unterstützt folgende Betriebssysteme:

- Windows 10 und 11 (32 und 64 Bit), Dedicated Mode
- Windows Server 2016, 2019 und 2022, Dedicated Mode

Systemvoraussetzungen

Kithara RealTime Suite unterstützt eine breite Palette von Hardware- und Software-Kombinationen sowie industriellen Standards.



Hardware

Die Software läuft auf folgender Hardware:

- CPU: AMD (ab Athlon) oder Intel (ab Pentium 2), 32 oder 64 Bit
- Dual- oder Multi-Core; mit Multi-Core optional Hyperthreading verwendbar, derzeit bis 48 logische CPUs, darüber auf Anfrage
- ACPI erforderlich (Advanced Control and Power Interface)
- Der PC muss PAE (Page Address Extension) unterstützen
- Gigabit-Ethernet bis 100 GBit/s, fast alle Controller von Intel und RealTek werden unterstützt

Compiler

Sofort verwendbare Programmgerüste für folgende Programmiersprachen/Compiler sind Teil der Software-Lieferung (auch weitere Programmiersprachen können auf Anfrage unterstützt werden):

- Visual Studio 20xx C++ mit MFC-Oberfläche (mitgelieferte Projektdateien: VS 2010/12/13/15/17/19/22)
- C++Builder (Embarcadero, ehem. Borland) mit VCL-Oberfläche
- Delphi (Embarcadero, ehem. Borland) mit VCL-Oberfläche
- Visual Studio 20xx C# mit WPF-Oberfläche (inkl. C++-DLL für Echtzeit-Ausführung, mitgelieferte Projektdateien: VS 2010/12/13/15/17/19/22)

Für die Code-Ausführung im Echtzeitkontext ist im Prinzip jeder Compiler verwendbar, der eine DLL mit nativem Maschinencode erzeugen kann. Die Windows-Anwendung kann jedoch auch mit anderen Programmiersprachen erzeugt werden, zum Beispiel mit C#.

Pakete



Für die häufigsten Anwendungsbereiche haben wir die wichtigsten Module zu Paketen zusammengestellt. Sie decken bereits den Bedarf unserer meisten Kunden ab, enthalten zusätzliche Tools und sind günstiger als die einzelnen Module.

Kithara RealTime EtherCAT – EtherCAT in Echtzeit unter Windows

Kithara RealTime EtherCAT ist eine Echtzeiterweiterung für Windows-Betriebssysteme, speziell fokussiert auf Automatisierung mit EtherCAT in Echtzeit.

Kithara RealTime EtherCAT Vision Set - EtherCAT und Vision in Echtzeit unter Windows

Kithara RealTime EtherCAT Vision Set ist eine Echtzeiterweiterung für Windows-Betriebssysteme, speziell fokussiert auf Automatisierung sowie Bilderfassung und -verarbeitung.

Kithara RealTime Vision – Echtzeit-Bilderfassung und -verarbeitung unter Windows

Kithara RealTime Vision ist eine Echtzeiterweiterung für Windows-Betriebssysteme, speziell fokussiert auf Bilderfassung mit GigE Vision und USB3 Vision sowie Bildverarbeitung in Echtzeit.

Kithara RealTime Automotive – Echtzeit für den Automobilbereich

Die erfolgreichsten Bussysteme in der Automobiltechnik, zuständig für Fahrzeugsicherheit, Fahrkomfort und Fahrerassistenzsysteme: CAN/CAN FD, FlexRay und LIN.

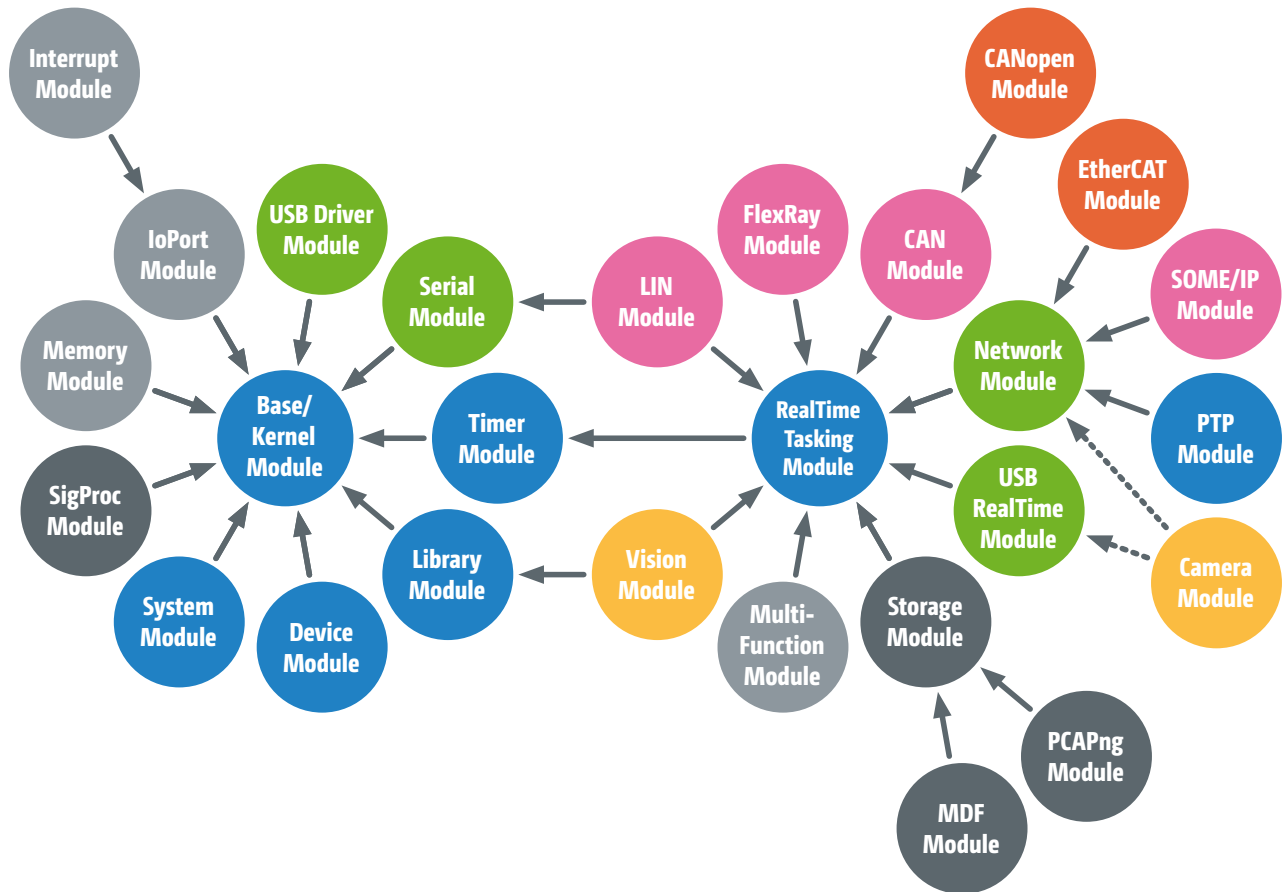
Pakete	RealTime EtherCAT	RealTime CANopen	RealTime Vision	RealTime Automotive
Basisfunktionen				
Base/Kernel Module	✓	✓	✓	✓
System Module	✓	✓	✓	✓
Library Module	✓	✓	✓	✓
Echtzeit-System				
Timer Module	✓	✓	✓	✓
RealTime Tasking Module	✓	✓	✓	✓
Storage Module				
PCAP Module / MDF Module				
PTP Module				
Hardware-Zugriffe				
IoPort Module				
Memory Module				
Interrupt Module				
MultiFunction Module				
Kommunikation				
Network Module	✓		✓	
USB RealTime Module			✓	
Serial/UART Module				✓
Device Module				
Development Tools				
Kernel Tracer	✓	✓	✓	✓
Performance Analyzer	✓	✓	✓	✓
Master Monitor	✓			
Automatisierung				
EtherCAT Module	✓			
CANopen Module		✓		
Bildverarbeitung				
Camera Module			✓	
Dlib Extension			✓	
Halcon Extension*			✓	
OpenCV Extension*			✓	
Automotive				
FlexRay Module				✓
CAN Module		✓		✓
CAN FD				✓
LIN Module				✓
BroadR-Reach (100BASE-T1/1000BASE-T1)				

Wählen Sie das passende Paket und kombinieren Sie es mit weiteren Modulen, falls gewünscht. Eine detaillierte Auflistung aller verfügbaren Module finden Sie nachfolgend.

* Die Halcon-Lizenz ist gesondert bei der Firma MVTec zu erwerben. OpenCV ist eine freie Programmbibliothek.

Module

Kithara RealTime Suite besteht aus verschiedenen Modulen. Die Software kann kundenspezifisch aus diesen Modulen zusammengestellt oder fertige Pakete mit weiteren Modulen ergänzt werden. Nachfolgende Grafik illustriert die Abhängigkeit der Module voneinander:



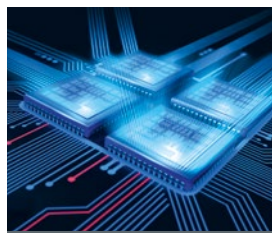
Basisfunktionen

Base/Kernel Module

Allgemeine Verwaltung, Ausführung auf Kernel-Ebene, Echtzeit-Speichermanagement, Debug-Hilfen, Systeminformationen, grundlegende Ressourcen, Device-Handling

Allgemein

Das Base/Kernel Module stellt generell die Grundlage für jedes Echtzeitsystem von Kithara RealTime Suite dar und ist daher immer erforderlich. Es ermöglicht das Öffnen des Echtzeittreibers von der Windows-Applikation aus sowie die Ausführung von Echtzeit-Code auf der Kernel-Ebene, um somit den Echtzeitkontext zu erreichen. Es erlaubt die allgemeine Verwaltung und stellt Funktionen für Debug-Hilfen und zur Ermittlung von Systeminformationen und Treiberversion bereit. Außerdem stehen Mechanismen für grundlegende Ressourcen bereit, wie etwa Anwendungs-Threads, Events, Callbacks, Shared Memory, Daten- und Message-Pipes und



Base/Kernel Module
System Module
Timer Module
RealTime Tasking Mod.
PTP Module

schnelle Mutex-Objekte. Zusätzlich werden eine Echtzeit-Speicherverwaltung, Memory-Kopierfunktionen und ein generischer WDM-Treiber für Plug-and-Play-Installationen zur Verfügung gestellt. Die integrierte KiK64-Funktion erlaubt die Ausführung von 32-Bit-Code auf 64-Bit-Systemen (nur mit der 64-Bit-Version von Kithara RealTime Suite).

Features

- Funktionen zum Öffnen des Treibers
- Ausführung von Echtzeit-Code auf der Kernel-Ebene
- Ermittlung von Fehlerbeschreibungen
- Debug-Hilfen
- Ermittlung von Systeminformationen
- Anwendungs-Threads
- Funktionen für Event- und Callback-Objekte, schnelle Mutex-Objekte
- Funktionen für Shared Memory
- Daten- und Message-Pipes mit automatischer Entkopplung zwischen schreibender und lesender Seite
- Device-Informationen, Driver-Management
- Echtzeit-Speicherverwaltung
- Memory-Kopierfunktionen für Echtzeitkontext
- Generischer WDM-Treiber für Plug-&-Play-Installation enthalten
- Generierte Log-Messages kompatibel zu Kithara Kernel Tracer
- KiK64-Funktion: 32-Bit-Code auf 64-Bit-System ausführbar (nur mit der 64-Bit-Version von Kithara RealTime Suite)
- Nur Sprachen, die nativen Maschinencode erzeugen können (C/C++ oder Delphi)
- Unterstützt die folgenden Betriebssysteme: Windows 10 und 11 (32 und 64 Bit) sowie Windows Server 2016, 2019 and 2022, jeweils Dedicated Mode
- Unterstützung von Multikernprozessoren, Hyperthreading und NUMA-Multiprozessor-PCs
- Base/Kernel Module Addons: Kernel CPU Extension, Kernel Memory Extension



Systemvoraussetzungen

- Folgende Systemvoraussetzungen müssen gegeben sein:
- Unterstützung des CMPXCHG8B/CMPXCHG16B CPU-Befehls

Erweiterungen

Kernel NUMA Extension

Verwendung von Mehrsocket-Systemen mit „Non-Uniform Memory Access“-Architektur in Echtzeit, automatische Optimierung der Speicherallokation.

Thunderbolt Extension

Unterstützung von über Thunderbolt angeschlossenen Geräten.

System Module

Abfangen von Systemereignissen wie Schutzverletzungen und Systemabstürzen auf der Kernel-Ebene (zum Beispiel FailSafe-Handler/„BlueScreen-Handler“)

Allgemein

Das System Module dient dem Abfangen von Systemereignissen wie Schutzverletzungen und Systemabstürzen auf der Kernel-Ebene. Dazu werden vorher registrierte Handler (Callback-Funktionen oder Echtzeit-Tasks) gerufen. Damit lassen sich beispielsweise FailSafe-Handler („BlueScreen-Handler“) realisieren und das System im Fehlerfall in eine definierte Ausgangsposition bringen.

Features

- Abfangen von Systemereignissen mit Callback-Funktionen oder Echtzeit-Tasks, zum Beispiel bei Schutzverletzungen und Systemabstürzen auf der Kernel-Ebene
- Realisierung von beispielsweise FailSafe-Handlern („BlueScreen-Handlern“)

Echtzeit-System

Timer Module

Hochgenaue und kalibrierte Ermittlung der Systemzeit sowie Kurzzeitverzögerungen

NEU!

Allgemein

Das Timer Module bildet mit seinen Zeitgeberfunktionen die Basis für alle Echtzeitaufgaben. Es stellt einen kalibrierten Zugang zu allen Hardware-Zeitbasen im System bereit. Zeitangaben lassen sich in beliebige, auch anwenderspezifische Zeitformate umrechnen. Kurzzeitverzögerungen sind je nach zugrundeliegender Hardware bis auf wenige Nanosekunden genau. Zusätzlich lassen sich einfache Timer ohne Echtzeit in Millisekundenauflösung auf Basis von Windows-Mechanismen erstellen.

Features

- Erkennung aller im System befindlichen Hardware-Zeitbasen
- Kalibrierung der verschiedenen Zeitgeber
- Langzeitsynchronisation bei Systemuhren
- Ermittlung der Systemzeit in verschiedenen Formaten, Auflösung bis ca. 0,1 μ s
- Hochgenaue Kurzzeitverzögerungen in 0,1- μ s-Schritten
- Programmierung anwenderspezifischer Zeitformate
- Zuverlässige Verhinderung von Überläufen durch interne 96-Bit-Operation
- Einfache Timer in Millisekundenauflösung auf Basis von Windows-Mechanismen (keine Echtzeit)

NEU!

RealTime Tasking Module

Programmierung hochfrequenter Echtzeit-Timer-Routinen und prioritätsgesteuertes, präemptives Echtzeit-Multitasking

Allgemein

Das RealTime Tasking Module ergänzt das **Timer Module** und ermöglicht die Programmierung von Timern, die den Anwendercode auf verschiedene Weise signalisieren können. Der Kontext der Programmausführung lässt sich wählen. Signalisierbare Objekte können Events und Callbacks auf Anwendungs- oder Kernel-Ebene sowie Echtzeit-Tasks sein. Zudem wird ein prioritätsgesteuertes, präemptives Echtzeit-Multitasking-System zur gegenseitigen Priorisierung einzelner Echtzeitaufgaben bereitgestellt. Damit lässt sich detailliert bestimmen, welche Bearbeitung vor anderen Vorrang hat, weniger wichtige unterbrechen soll und welche sich selbst von wichtigeren Aufgaben unterbrechen lässt, was den Mitteln eines reinen Echtzeitbetriebssystems (RTOS) entspricht.

Features

Real Time

- Entwicklung hochfrequenter Echtzeit-Timer-Routinen
- Zyklische oder einmalige Timer programmierbar
- Einfache Debug- und Testmöglichkeit bei Callback auf Anwendungsebene
- Timer lassen sich gezielt starten, stoppen, abbrechen
- Watchdog-Mechanismen einfach umsetzbar
- Startzeitpunkt lässt sich in 0,1- μ s-Auflösung vorgeben (etwa zur Synchronisation mit anderen Abläufen)
- Timer-Frequenz bis zu 20 kHz und mehr
- Jitter im einstelligen Mikrosekundenbereich
- Timerperiode dynamisch justierbar

Multitasking

- Präemptives Echtzeitsystem
- Echtzeit-Tasks mit bis zu 255 Prioritätsstufen programmierbar
- Dynamische Anpassung der Prioritätsstufe möglich
- Mehrere Tasks mit der gleichen Prioritätsstufe als „Round-Robin“
- Inklusive Prioritätsvererbung zur Vermeidung der Prioritätsinversion
- Echtzeit-Semaphore zur Synchronisation zwischen Echtzeit-Tasks
- Echtzeit-Events zur Signalisierung von Tasks von außerhalb
- Tasks lassen sich suspendieren, fortsetzen, triggern, vorzeitig beenden („exit“) und abbrechen („kill“)
- Tasks lassen sich verzögern (in 0,1- μ s-Auflösung)

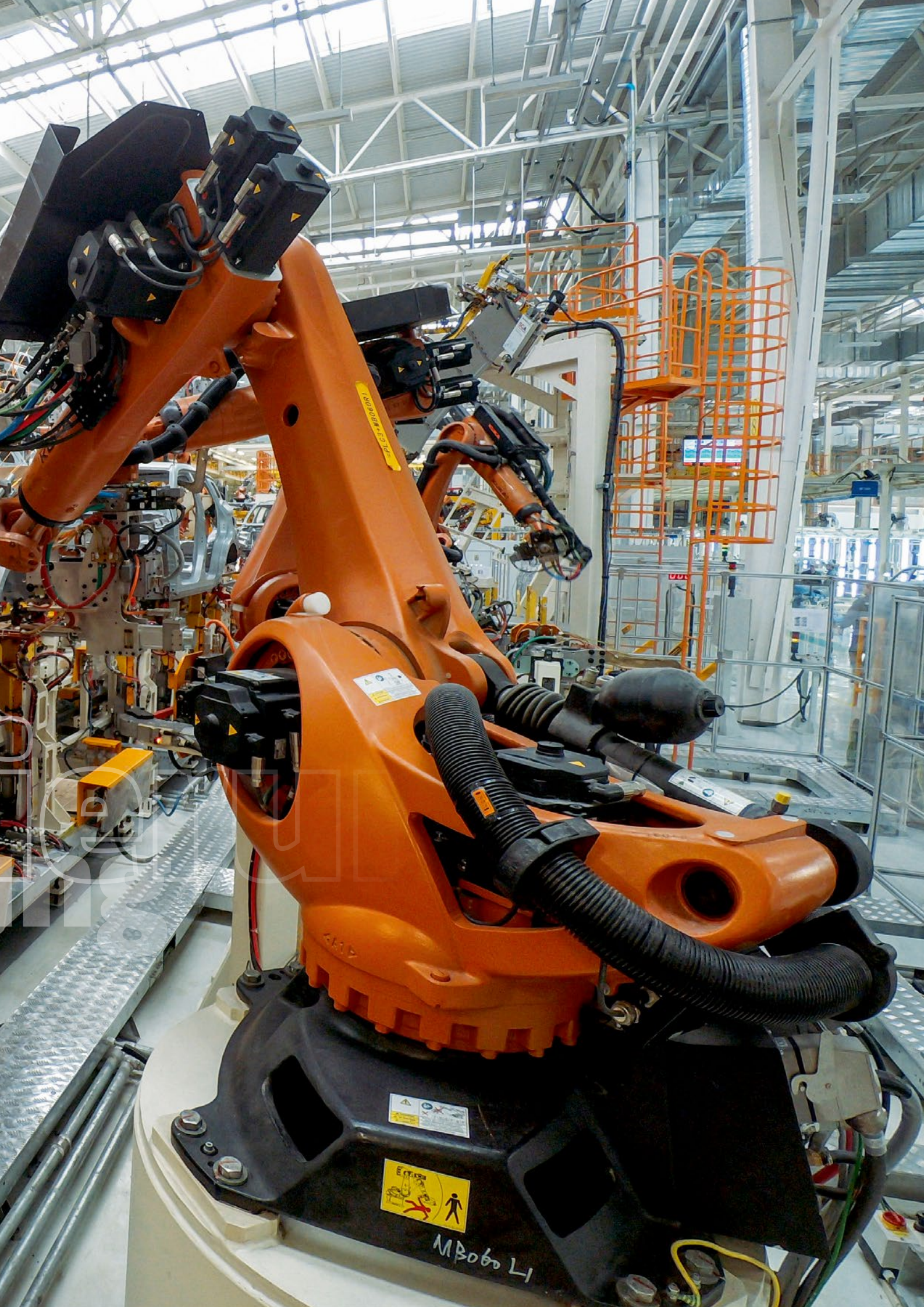
- Betrieb von CPU-Kernen ausschließlich im Echtzeitmodus zur Vermeidung von Windows-Beeinflussung
- sehr kurze Taskwechselzeiten
- Speedloop-Modus für hochgenaue zyklische Ausführung

Erfordert **Timer Module**

PTP Module	Echtzeit-Synchronisation mit Precision Time Protocol
Allgemein	<p>Mit dem PTP Module lassen sich mehrere Netzwerkteilnehmer in einem verteilten System präzise synchronisieren. Das Kithara Echtzeitsystem sorgt dafür, dass genaue Zeitstempel mit Abweichungen im Sub-Mikrosekundenbereich erzeugt werden, um so die Zeitgeber aller Teilnehmer genau abzugleichen. Die API ermöglicht die Verwendung des BMCA (Best Master Clock Algorithm), der den bevorzugten Zeitgeber nach konfigurierbaren Parametern ermittelt. Alternativ können Master und Slaves manuell gesetzt werden.</p>
Features	<ul style="list-style-type: none"> ■ Präzise Synchronisation mehrerer Netzwerkteilnehmer ■ Genaue Zeitstempel mit Abweichungen im Sub-Mikrosekundenbereich ■ Konfigurierbarer Best Master Clock Algorithm (BMCA) ■ Manuelle Einstellung von Master und Slaves ■ Benötigt Network Module
Erweiterungen	<p>NMEA Extension Echtzeit-Kommunikation und -synchronisation mit NMEA 0183.</p> <p>gPTP Extension Die gPTP Extension erweitert das PTP Module um die Unterstützung für das „generalized Precision Time Protocol“ (definiert in IEEE 802.1AS), einem vereinfachten PTP-Profil. Die gPTP Extension ermöglicht das hochgenaue Synchronisieren aller Zeitgeber in einem verteilten System, beschränkt sich jedoch auf einen übersichtlichen Satz an Funktionen und ist damit für ein breiteres Anwendungsspektrum geeignet. So müssen Zeitgeber beispielsweise nicht mehr einzeln konfiguriert werden, sondern sind vereinheitlicht.</p> <ul style="list-style-type: none"> ■ Präzise Synchronisation mehrerer Netzwerkteilnehmer ■ Vereinfachtes PTP-Profil ■ Alle Zeitgeber vereinheitlicht ■ Einfachere Konfiguration des Best Master Clock Algorithm



Automatisierung



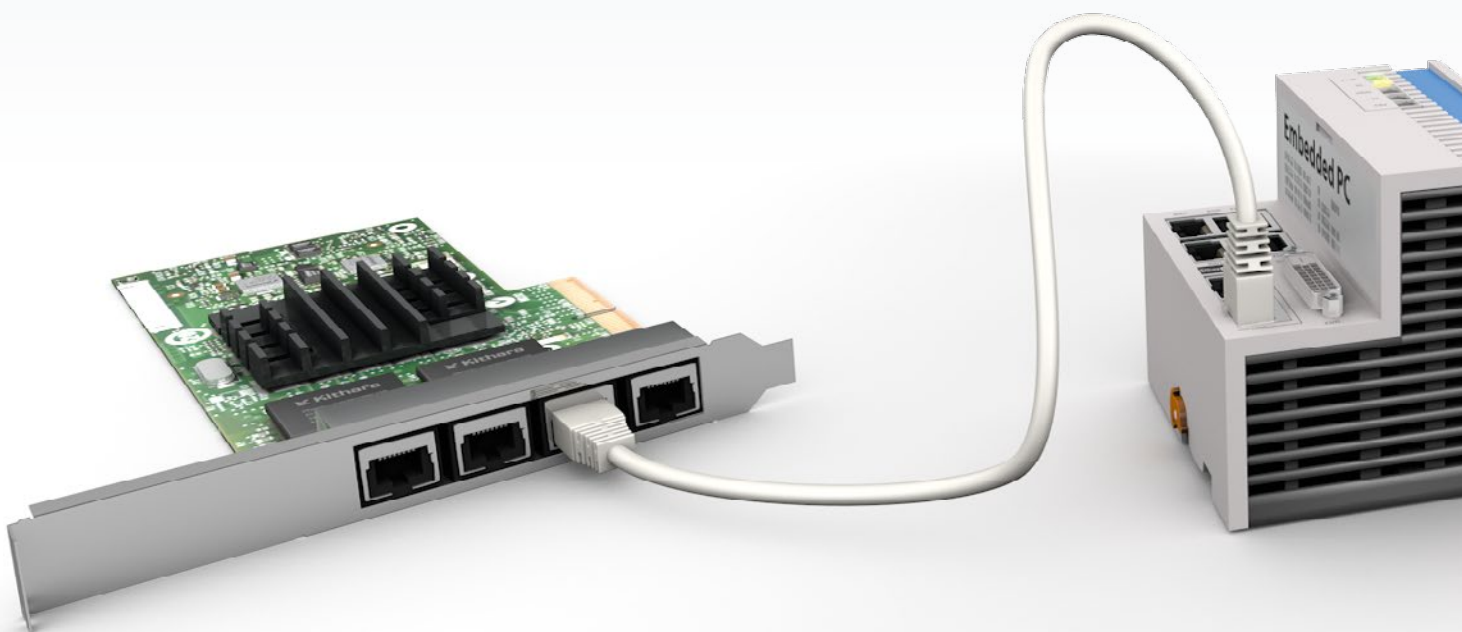
Softwarelösungen für PC-gestützte Automatisierung

In der Automatisierungstechnik sind standardisierte Protokolle inzwischen unverzichtbar. Bei den bekanntesten Feldbussen hat sich vor allem die offene und schnelle Industrial-Ethernet-Variante EtherCAT durchgesetzt und rasanten Zulauf erhalten. Hierfür stellt Kithara RealTime Suite auch einen leistungsfähigen EtherCAT-Master bereit.

Dessen „harte“ Echtzeiteigenschaften ermöglichen Anlagenautomatisierung mit Zykluszeiten von nur 50 μ s oder weniger. Der Master stellt außerdem alle benötigten Mechanismen von der automatischen Ermittlung der angeschlossenen Topologie über die Unterstützung von Slaves mit Distributed Clocks (DC) bis hin zu Besonderheiten wie Hotplug-Fähigkeit oder Kabelredundanz zur Verfügung.

Zur Implementierung sicherer Anwendungen kann auch Safety-over-EtherCAT (FSoE) integriert werden. In diesem Fall sorgt der EtherCAT-Master für den Austausch der Datenpakete zwischen den Safety-Ein- und -Ausgängen sowie der Safety-Logik-Baugruppe. Damit lassen sich Anwendungen bis SIL3 realisieren, eine ansonsten erforderliche gesonderte Verkabelung, etwa für eine Notfall-Deaktivierung, kann entfallen.

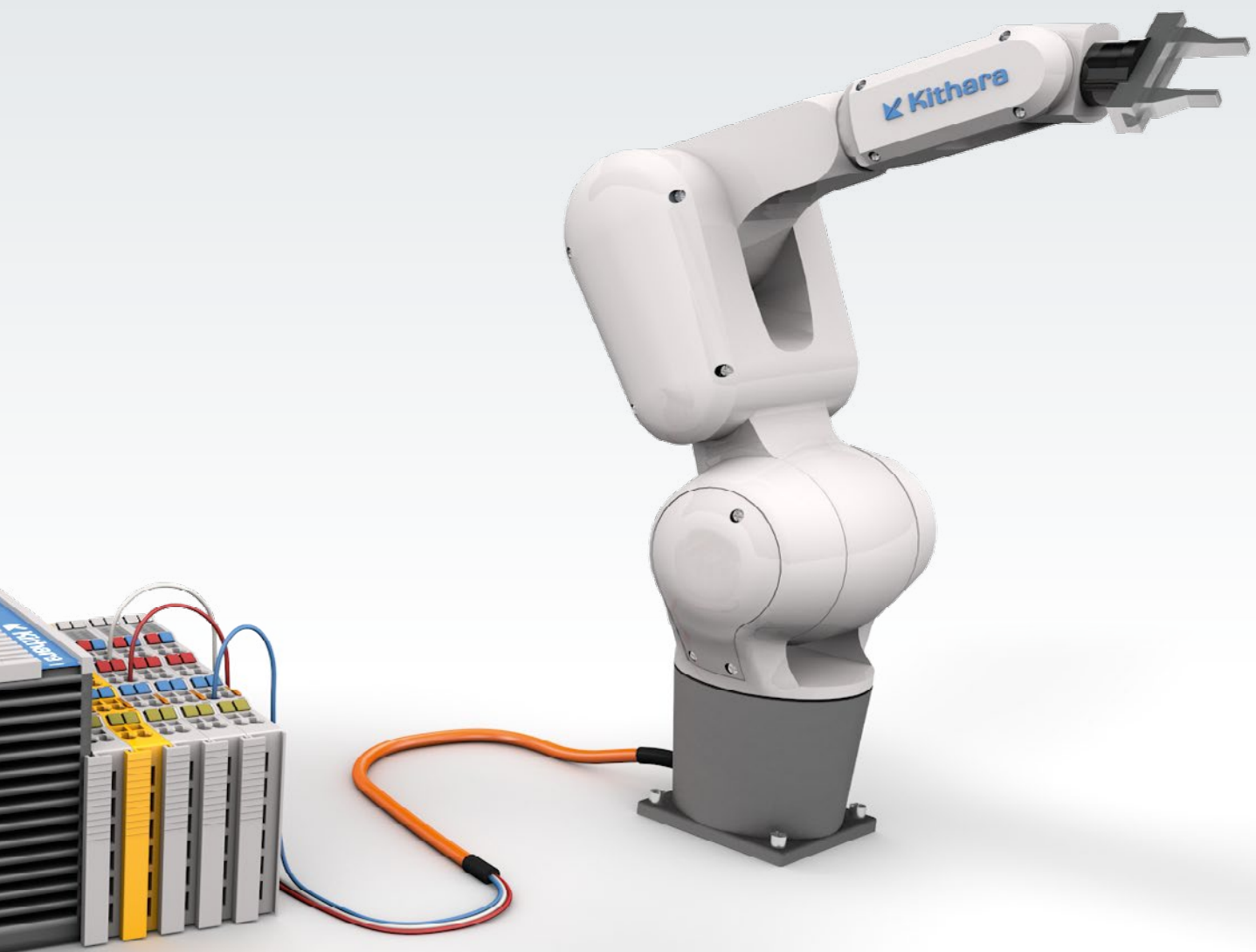
Des Weiteren ist die Implementierung von PCs als EtherCAT-Slaves möglich, um flexible Schnittstellen für skalierbare EtherCAT-Netzwerke zu schaffen. Über spezielle PCIe-Karten können PCs so auch in übergeordnete EtherCAT-Topologien eingebettet und als Slaves betrieben werden. EtherCAT-PC-Slaves profitieren dabei von der grafischen Oberfläche und den Eingabemöglichkeiten handelsüblicher PCs und sind auf vielfältige Aufgaben spezialisierbar, beispielsweise zum Auslagern oder Skalieren von Rechenleistung.



Der ebenfalls verfügbare CANopen-Master lässt sich sowohl an den unterstützten CAN-Schnittstellen betreiben als auch eingebettet in ein höheres EtherCAT-Netzwerk integrieren.

Zur Einbettung von EtherCAT-Topologien in höhere Ebenen oder sogar Firmennetze ist das EtherCAT Automation Protocol (EAP) verwendbar. Geräte und ganze Produktionssegmente, gleicher und auch verschiedener Ebenen, können so Daten mit bis zu 100 GBit/s austauschen. Das erlaubt einerseits

Master-Master-Kommunikation, andererseits aber auch die Anbindung an übergeordnete Produktionssysteme (MES, ERP) und Abteilungen (Logistik, Vertrieb etc.). Diese Komplettnetzwerk automatisierter Anlagen macht das EAP zur Basis für die Umsetzung cyberphysischer Systeme.



Module



EtherCAT Module

EtherCAT Master in Echtzeit

Allgemein

Kithara EtherCAT Master basiert intern auf dem prioritätsgesteuerten, preemptiven Echtzeit-Multitasking-System. Unter Ausnutzung der Hardware-Parallelität bei mehreren CPU-Kernen können die verschiedenen Echtzeitaufgaben verteilt werden. Dadurch bietet sich eine hohe Skalierbarkeit der Echtzeitausführung bis hin zu extremer Echtzeit auf exklusiv benutzten CPU-Kernen im Dedicated Mode.

Features

- Eigenständiger EtherCAT Master in Echtzeit
- Automatische Erkennung der EtherCAT-Topologie mit XML-Dateien oder SII-Protokoll
- Prozessdaten-Kommunikation (zyklischer I/O-Datenaustausch) in Echtzeit
- Kurze Zykluszeiten: bis zu 50 Mikrosekunden oder weniger
- Hot connect und Kabelredundanz
- Netzwerkkarten von RealTek und Intel verwendbar
- I/O-Klemmen, Servoumrichter etc. aller Hersteller werden unterstützt (zum Beispiel Beckhoff)
- Modularer Aufbau des EtherCAT Masters: Grundpaket + Optionen
- Prozessdaten-Kommunikation (PDO)
- Servicedaten-Kommunikation (SDO)
- Mailbox-Kommunikation
- Benötigt **Network Module**

Erweiterungen

EtherCAT PC Slave Device Extension

Mit der EtherCAT PC Slave Device Extension kann ein handelsüblicher PC wie ein EtherCAT-Slave verwendet und in EtherCAT-Netzwerke eingebunden werden. Vor dem Einsatz von PCIe-Slave-Karten besaß nur die Master-Seite die Möglichkeit zur komplexen PC-basierten Kommunikation mit dem EtherCAT-Netzwerk. Durch die Implementierung von Rechnern als Slaves können anspruchsvolle Automatisierungsprozesse zielgerichteter entworfen, implementiert und angepasst werden.



- Gemeinsames API mit EtherCAT Master und EtherCAT EAP
- Prozess- und Servicedatenkommunikation (PDO/SDO)
- Dateitransfer (FoE)
- Erzeugen von benutzerdefinierten PDO-Mappings
- SII-Daten (EEPROM) des Slave Device können beliebig initialisiert werden (zum Beispiel Vendor ID, Product ID, Revision)

Hardware-Unterstützung

Beckhoff – EtherCAT-Slave-Geräte

- FC1100, PCI EtherCAT Slave Card
- FC1121, PCIe EtherCAT Slave Card
- CX5000 CCAT (Beckhoff CX50xx, CX20xx, CX51xx Embedded PCs)

ESD – EtherCAT-Slave-Karte

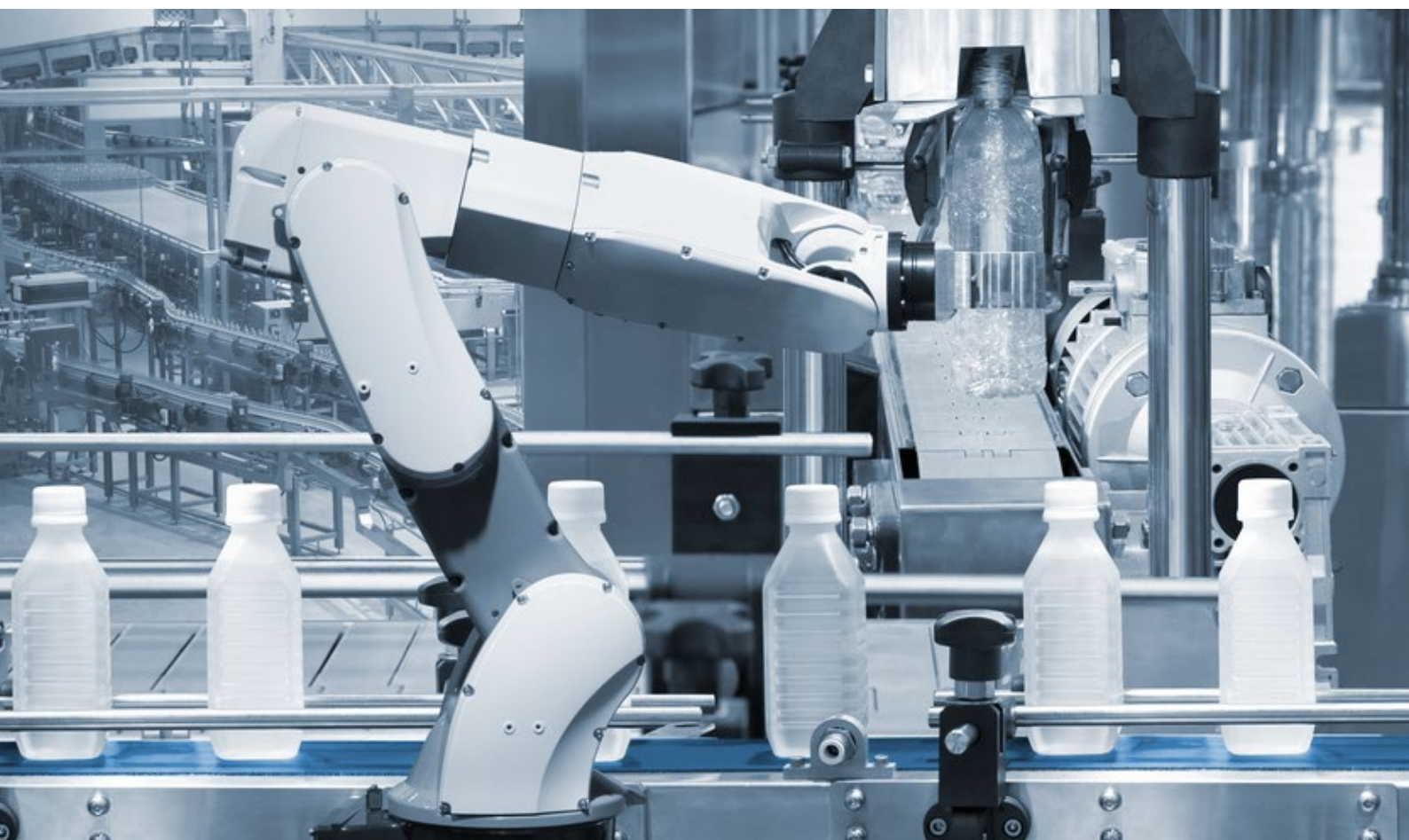
- ECS-PCIe 1100, PCIe EtherCAT Slave Card

Distributed Clocks (DC)

Distributed Clocks wird für die softwarebasierte Zeitsynchronisation zwischen allen Busteilnehmern eingesetzt. Der Master ermittelt während der Initialisierung die Verzögerungen zu den einzelnen Slaves und stellt deren Uhren. Zur Driftkorrektur wird regelmäßig eine Nachricht durch die Ring-Topologie an alle Slaves gesendet, um diese über Langzeit genauestens zu synchronisieren.

Ethernet over EtherCAT (EoE)

Mit Ethernet over EtherCAT lassen sich beliebige Ethernet-Geräte innerhalb von EtherCAT via Switchports anschließen. Hierbei wird



Standard-Ethernet-Kommunikation über EtherCAT getunnelt, was dem Master ermöglicht, diese zu optimieren, ohne die Prozessdaten-Kommunikation zu beeinträchtigen. Einige Slaves bieten auch ein Webinterface, etwa für die Konfiguration, welches über EoE angesprochen werden kann.

File access over EtherCAT (FoE)

File access over EtherCAT erlaubt den einfachen Dateizugriff auf Netzwerkteilnehmer und kann beispielsweise für das Uploaden einheitlicher Firmware auf mehrere Geräte im EtherCAT-Netzwerk verwendet werden. Das Protokoll ist absichtlich simpel gehalten, um auch Bootloader zu unterstützen.

Servo Profile over EtherCAT (SoE)

Das Servodrive-Profil über EtherCAT stellt die Unterstützung für die Sercos-Schnittstelle bereit, mit der sich anspruchsvolle Motion-Control-Anwendungen umsetzen lassen. Das Sercos-Profil für Servoantriebe sowie dessen Mapping auf EtherCAT sind in IEC 61800-7 standardisiert.

Hot Connect (HC)

Hot Connect ermöglicht es, vor oder im laufenden Betrieb, vorkonfigurierte Segmente (Slave-Gruppen oder einzelne Slaves) aus dem Datenverkehr zu nehmen oder hinzuzufügen. Damit lässt sich die Topologie flexibel anpassen.

Cable Redundancy (CR)

Mit der Kabelredundanz-Funktionalität wird der Betrieb, auch bei ungewollten Kommunikationsunterbrechungen wie Kabelbruch, weitergeführt. Hierfür werden die Prozessdaten redundant über einen zweiten Ethernet-Port durch die Topologie gesendet.

Safety over EtherCAT (FSoE):

Durch das TÜV-zertifizierte Protokoll Safety-over-EtherCAT lässt sich die Funktionalität von EtherCAT mit Sicherheitsstufe SIL 3 (nach IEC 61508) realisieren. Das Sicherheitsprotokoll verursacht keine Einschränkungen bei Übertragungsgeschwindigkeit oder Zykluszeit, da EtherCAT als einkanaliges Kommunikationsmedium genutzt wird.

EtherCAT Autom. Protocol

Echtzeit-Vernetzung über EtherCAT von der Leitebene aus

Allgemein

Das EtherCAT Automation Protocol ermöglicht Echtzeitkommunikation zwischen allen Endgeräten innerhalb eines Netzwerks, wodurch ein komplett neuer Standard an Konnektivität bei Auto-

matisierungsanlagen erreicht wird. Dazu können beispielsweise alle PC-basierten Steuerungen, Maschinen, Prüfstände, Transportbänder, Roboter, Anlagen zur Qualitätssicherung oder MES (Manufacturing Execution System) gehören.

Features

- Übertragungsrate von beispielsweise 1000 MBit/s, 10 GBit/s oder 100 GBit/s
- Prozessdaten- und Servicedatenaustausch (PDO/SDO), Dateitransfer
- Erfordert [EtherCAT Module](#)

CANopen Module

CANopen Master in Echtzeit

Allgemein

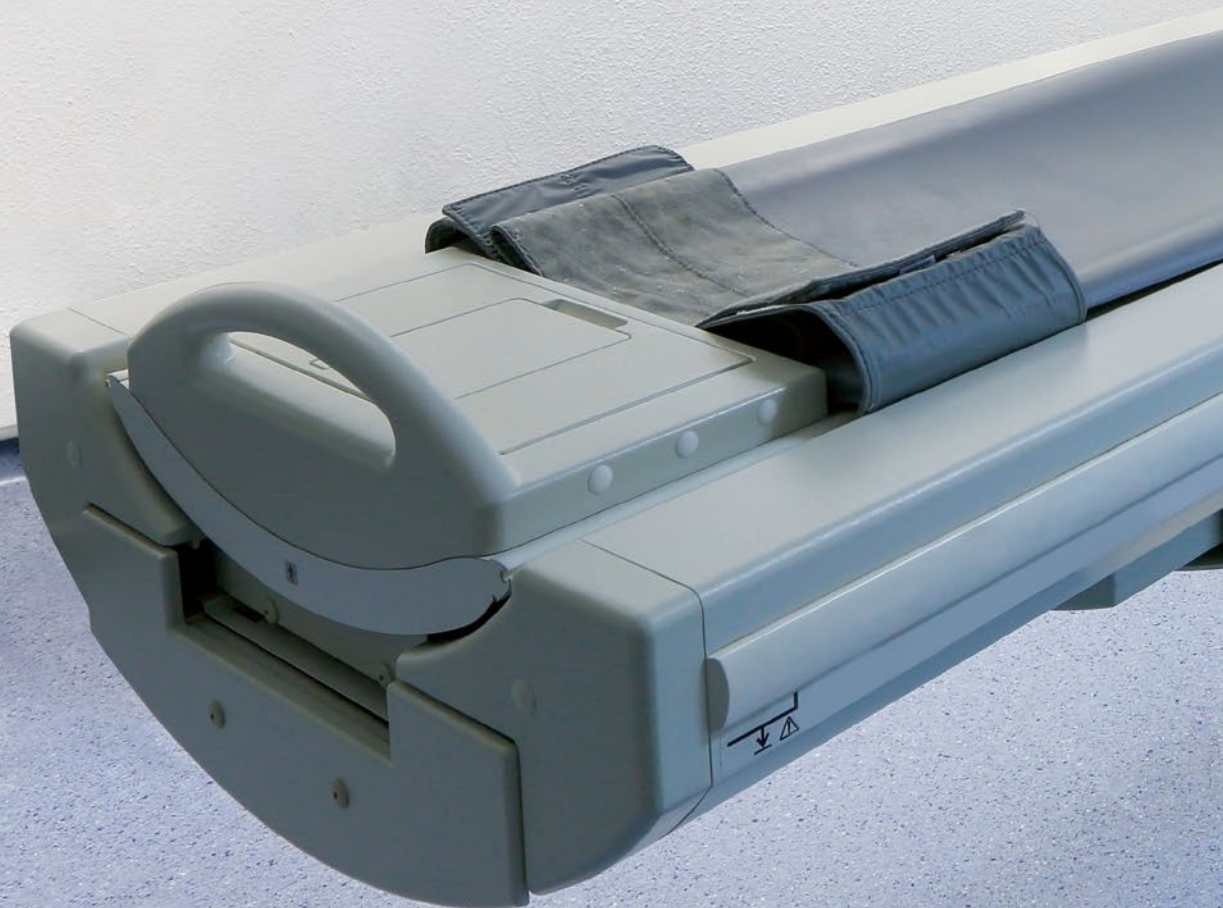
Das CANopen Module von Kithara RealTime Suite beinhaltet einen PC-basierten Master für das Automatisierungsprotokoll CANopen. Es basiert auf dem CAN Module und erfordert entsprechende Hardware.

Features

- Automatische Ermittlung der CANopen-Topologie
- Management der CANopen-Slave-Zustände
- Prozessdaten- und Servicedaten-Kommunikation (PDO+SDO)
- Mailbox-Kommunikation
- Erfordert [CAN Module](#)



Bildverarbeitung





Industrielle Bildverarbeitung in Echtzeit

Viele automatisierte Anlagen sind auf industrielle Bildverarbeitung angewiesen, sei es für Produktion, Verpackung, Überwachung oder Qualitätssicherung. Die entsprechenden Vision-Interface-Standards für Industriekameras sowie der Einsatz von Bildverarbeitungsbibliotheken benötigen jedoch garantierte geringe Reaktionszeiten, um verschiedenste Anforderungen zu erfüllen. Kithara RealTime Suite ist hierbei ein führender Wegbereiter auf dem Gebiet der Bildfassung und -verarbeitung in Echtzeit.

Die Unterstützung mit Automatisierungsprotokollen in einem geschlossenen Echtzeitzyklus bietet den entscheidenden Vorteil, das Ergebnis der algorithmenbasierten Bildanalyse sofort an den Prozess ausgeben zu können, ohne den Echtzeitkontext verlassen zu müssen. Dadurch können selbst Systeme in der Sensor-/Aktor-Ebene mittels verarbeiteter Bilddaten direkt angesteuert werden.

Für Ethernet-basierte Kamerasysteme wird ein echtzeitfähiger GigE-Vision-Treiber bereitgestellt, mit dem auf das Eintreffen des vollständigen Kamerabildes innerhalb weniger Mikrosekunden reagiert werden kann. Dabei wird auch Link-Aggregation mit paralleler Übertragung auf mehreren Kanälen unterstützt. Die Übertragungsraten können je nach verwendeter Hardware sogar mehr als ein Gigabyte pro Sekunde betragen. Die Vorteile von GigE Vision decken sich mit denen von Ethernet, also dem Einsatz günstiger, austauschbarer Standard-



Hardware sowie Datenraten bis 10 GBit/s und hohen Kabelreichweiten. Mit GigE Vision 2.0 wurde auch das Precision Time Protocol (IEEE 1588) eingeführt. Mehrere Kameras lassen sich damit in einen PTP-Modus versetzen und mikrosekundengenau in Echtzeit miteinander synchronisieren. Dafür wird ein Gerät als Master bestimmt, dessen Zeitgeber die Zeitstempel der anderen Geräte vorgibt. Über GPS lassen sich Kameras und weitere Geräte innerhalb eines Ethernet-Netzwerks sogar weltweit synchronisieren.

Ähnlich kostengünstig und flexibel ist die Verwendung des USB3-Vision-Standards. Auf Basis echtzeitfähiger Treiber für

USB 3.1 lassen sich auch USB3-Vision-Kameras mit Echtzeiteigenschaften und hohen Übertragungsraten nutzen.

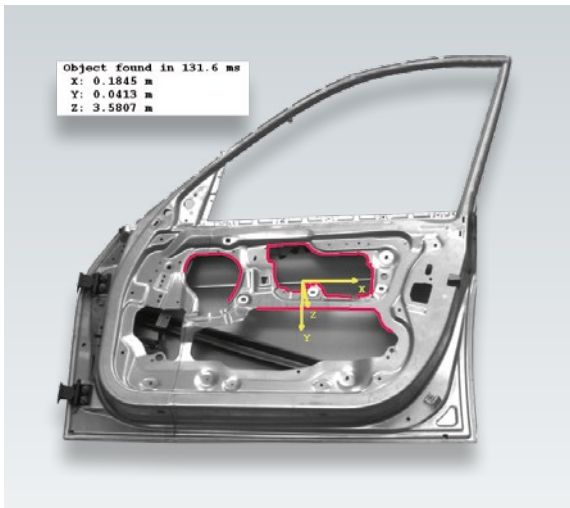
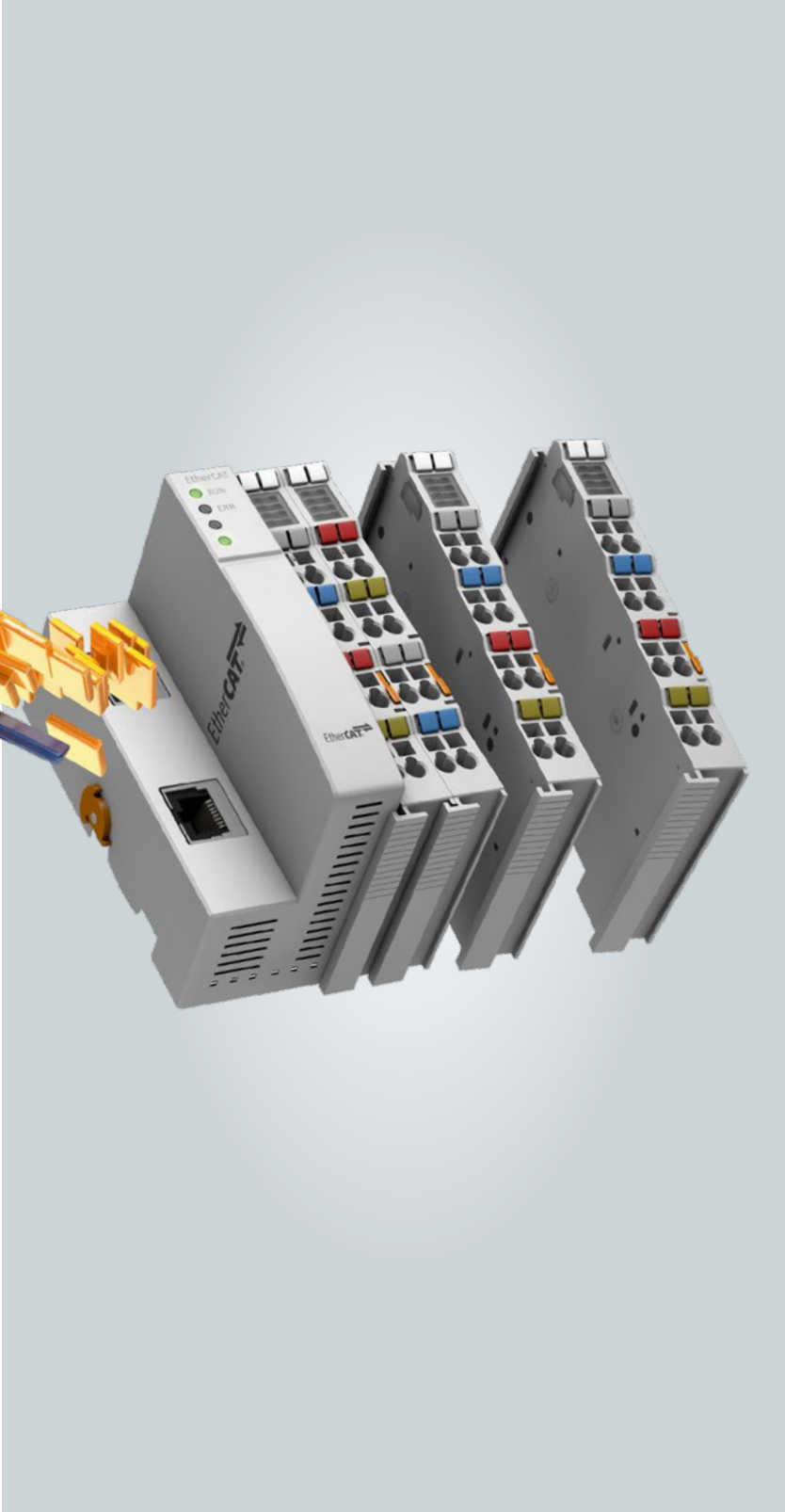
Die umfassende Programmierschnittstelle GenICam erlaubt Entwicklern die komfortable Integration beider Standards. Und über optionale Hardware wird sogar CameraLink unterstützt. Mit Hilfe eines externen Frame-Grabbers, welcher die CL-Steuerinformationen und -Datenpakete nach GigE Vision konvertiert, können alle CL-Konfigurationen bis einschließlich „Full“ über eine 10-Gigabit-Ethernet-Leitung in die Echtzeitumgebung integriert werden. Somit erhöht sich die bislang nur geringe Kabellänge auf Ethernet-Niveau.



Für die sofortige Weiterverarbeitung und Auswertung der Bilddaten sind eigene Algorithmen oder externe Bildverarbeitungsbibliotheken wie Halcon und OpenCV nutzbar. Dabei wird die jeweilige Bibliothek direkt in den Echtzeitkontext geladen. Bei Verwendung von Halcon können die in der Entwicklungsumgebung HDevelop entwickelten Prozeduren unmittelbar in Echtzeit ausgeführt werden.



Für GigE-Vision-Anwendungen mit besonders hohen Leistungsansprüchen kann der Framegrabber PGC-1000 von PLC2 Design in Echtzeit eingesetzt werden. Die PCIe-Karte übernimmt den kompletten Umwandlungsvorgang erfasst e r GigE-Vision-Daten, wodurch die CPU bei Bildverarbeitungsprozessen fast vollständig entlastet wird, um so eine leistungsfähigere Entwicklungsumgebung zu schaffen.



Bei Fertigung und Qualitätssicherung, vor allem auch im Automobilbereich, werden immer häufiger Kamerasysteme verwendet. Dabei kommt es besonders auf hochpräzise Erkennung und korrekte Interpretation der erfassten Bilder an. Ohne „harte“ Eigenschaft bei der Bildverarbeitung wäre dies nicht umsetzbar. Kithara schafft die hierfür notwendigen technischen Voraussetzungen.

Module



Camera Module

Bilderfassung mit GigE-Vision- und USB3-Vision-Kameras in Echtzeit

Allgemein

Das Camera Module dient der Erfassung von Bilddaten von industriellen Kameras nach dem GigE-Vision- oder dem USB3-Vision-Standard in Echtzeit. Es stellt alle Funktionen zur Erkennung, Verwaltung und zur Erzeugung von Echtzeit-Streams bereit.

Große Pufferspeicher in Verbindung mit den Echtzeit-Netzwerk- oder -USB3-Treibern verhindern zuverlässig einen Verlust von Datenpaketen. Die Reaktion auf eingetroffene Kamerabilder erfolgt im Kontext der Echtzeitumgebung innerhalb weniger Mikrosekunden. Dabei kann ein Echtzeit-Task aktiviert werden, um eine sofortige Steuerungsreaktion auszulösen.

Mit dem Camera Module ist dank der zugrunde liegenden Echtzeit-Treiber ein maximaler Datendurchsatz je nach Schnittstelle und Systemkonstellation bis zum theoretischen Maximum erreichbar. Zur Verarbeitung der Bilddaten siehe [Halcon Extension](#) und [OpenCV Extension](#).

Die Ergebnisse der Bildauswertung können noch im Echtzeitkontext verzögerungsfrei an den Prozess gegeben werden, zum Beispiel über EtherCAT oder CANopen. Dazu lassen sich Automatisierungslösungen im Rahmen von Kithara RealTime Suite mit verschiedenen anderen Themen kombinieren.

Features

Funktionen

- Bilderfassung von GigE-Vision- oder USB3-Vision-Kameras in Echtzeit
- Abfrage- oder ereignisgesteuert
- Mehrere Kameras gleichzeitig, Hot-Plug-fähig, Konfiguration nach GenICam 2.0
- Jede standardkonforme Kamera wird unterstützt

Steuerung

- Automatische Handhabung der Verbindungen
- Voller Zugriff auf die Kamerakonfiguration
- Schreiben und Lesen des Kameraspeichers
- Setzen des Custom Heartbeat Timeout
- Start und Stop der Erfassung von Bildern durch die Software
- Einstellung des Erfassungsmodus durch die Software
- GenICam-XML-Konfigurationsdatei wird automatisch heruntergeladen
- Zugriff auf GenICam-Funktionen zum Konfigurieren der Kamera
- Fehlerhandhabung

Streams

- Ein oder mehrere Streams pro Kamera sind möglich
- Erfassung von Einzelbildern und Bildfolgen in Echtzeit
- Speicheroptimierter Bildempfang, kein unnötiges Kopieren
- Einstellung der eigenen Buffer-Größen und -Anzahl
- Signalisierung entgangener oder unvollständiger Bilder

GenICam Funktionen

- Nummeriert alle Kategorien
- Einstellung und Abfrage aller zugänglichen Kamerafunktionen wie Bildformat, Pixelformat, Bildrate, Belichtungszeit und mehr

Das Camera Module benötigt für die Echtzeit-Schnittstelle weitere Module, zum Beispiel das **Network Module** für GigE Vision und/oder das **USB RealTime Module** für USB3 Vision.



Erweiterungen

Camera Acceleration Extension

Die Camera Acceleration Extension stellt leistungsfähige Echtzeittreiber für die GigE-Vision-Framegrabber-Beschleunigerkarte PGC-1000 des Unternehmens PLC2 bereit. Eingebettet in Kithara RealTime Suite kann die PGC-1000 innerhalb von Machine-Vision-Applikationen die CPU bei Bilderfassungsprozessen fast vollständig entlasten. So ist die Karte beispielsweise in der Lage, gleichzeitig bis zu vier GigE-Vision-Kamera-Streams mit je 10 GBit/s im Echtzeitkontext zu erfassen. Auch die Echtzeitsynchronisation mehrerer Kameras über das PTP Module von Kithara ist möglich.

- Echtzeitunterstützung für PGC-1000
- Fast vollständige Entlastung der CPU bei GigE-Vision-Bilderfassung
- Verbindungsgeschwindigkeit bis zu 4×10 GBit/s bzw. 64×1 GBit/s
- Synchronisation mehrerer Kameras über PTP Module
- Erfordert [Camera Module](#) und [Network Module](#)
- Erfordert PGC-1000 von PLC2 Design GmbH

Vision Module

Echtzeit-Bildverarbeitung mit Halcon oder OpenCV

Allgemein

Bei Halcon und OpenCV handelt es sich um Programmbibliotheken mit Algorithmen für Bildverarbeitung und maschinelles Sehen, die in das Kithara-Echtzeitsystem integriert werden. Sie ermöglichen eine sofortige Steuerungsreaktion auf verarbeitete Bilder und deren Einbindung in Automatisierungssysteme mit EtherCAT oder CANopen.

Features

- Verwendung der Bildverarbeitungsbibliotheken Halcon oder OpenCV zur Bildverarbeitung in Echtzeit
- Die Halcon-Lizenz ist gesondert bei der Firma MVTec zu erwerben
- Unterstützt Halcon-Versionen 10 bis 13, 18.11, 20.11 sowie OpenCV 3.0, 3.4 und 4.1
- Beinhaltet Halcon Extension und OpenCV Extension
- Erfordert [RealTime Tasking Module](#) und [Library Module](#)







Kommunikation

Hardwarezugriffe und Kommunikation in Echtzeit

Verschiedene Module von Kithara RealTime Suite dienen der Ankopplung an PC-Hardware. Über Zugriffe auf die I/O-Register des PCs, den physischen Speicher und die Behandlung von Interrupts lassen sich Treiber für eigene Hardware-Baugruppen oder Steckkarten erstellen. Für viele der gängigen Kommunikations-Schnittstellen stehen zudem bereits vorgefertigte, leistungsfähige Echtzeit-treiber zur Verfügung.

Gerade im industriellen Bereich nimmt die Bedeutung von Ethernet-Anwendungen mit hohen zeitkritischen Anforderungen zu. Die hierfür speziell entwickelten Treiber unterstützen alle gängigen Ethernet-Netzwerk-Controller von Intel und RealTek mit Datenraten bis zu 100 GBit/s. Sämtliche Funktionen und Vorteile von Ethernet, wie etwa der Einsatz flexibler und kostengünstiger Standard-Hardware, sind wie gewohnt vollständig nutzbar, werden jedoch um Echtzeiteigenschaften erweitert. So lassen sich auch alle Hardware-Besonderheiten der verwendeten Controller,

wie Jumbo-Frames oder verbesserte Datenfluss-Steuerung durch Interrupt-Optimierung, effektiv einsetzen.

Die Echtzeitfähigkeit wird durch eine direkte Ansteuerung der Netzwerk-Controller erreicht. Damit steht die gesamte Bandbreite der Verbindung zur Verfügung, die Reaktion auf eintreffende Datenpakete erfolgt unmittelbar. Um eine leistungsfähige Netzwerkkommunikation zu garantieren, werden alle störenden Windows-Mechanismen umgangen, die Übertragungsgeschwindigkeiten und unmittelbare Reaktionen negativ beeinflussen könnten. Für die Umsetzung der Socket-Kommunikation wird eine spezielle Treiberschicht für Datagramm- (UDP) oder Stream-orientierte (TCP) Kommunikationsanwendungen bereitgestellt.

Für schnelle Kommunikation über USB bis Version 3.1 sorgt ein speziell programmierter, einfach anwendbarer Zugang zum Extensible Host Controller Interface (xHCI). USB ist dank kostengünstiger Implementierung mit



standardisierter Hardware sowie guten Übertragungsraten einer der am weitesten verbreiteten Kommunikationsstandards. Durch den mit USB 3.0 eingeführten SuperSpeed-Modus, welcher noch höhere Datenraten sowie Voll-Duplex-Kommunikation erlaubt, wird USB zunehmend auch in anspruchsvolleren industriellen Bereichen gewinnbringend angewandt.

Auch Thunderbolt lässt sich in Echtzeit verwenden und ermöglicht so eine unmittelbare Kommunikation der darüber verbundenen Komponenten bei geringstmöglicher Verzögerung. Dabei erlaubt die Thunderbolt-Implementierung auch die Verwendung entsprechender Hardware-Erweiterungen in Echtzeit, beispielsweise zum Anschließen von PCIe-Steckkarten an Laptops. So lässt sich eine verbesserte Mobilität von softwaregestützten Echtzeitlösungen, etwa beim Testen oder Messen vor Ort an der Maschine, erzielen.

Serielle UART- sowie COM-Schnittstellen können ebenfalls angesprochen werden. Unterstützt werden die gängigen Standards RS-

232, RS-422 und RS-485 mit Baudraten bis zu 115.200 bzw. 921.600 Bit/s. Dabei sind UART-Funktionen auch aus Echtzeit-Tasks heraus und auf Dedicated CPUs nutzbar. Handshake- und Signal-Leitungen können direkt angesteuert werden.

Mit einer speziellen zusätzlichen Programmierschnittstelle können zudem Aufrufe von Betriebssystem-Funktionen für die Kommunikation mit Gerätetreibern auf der Kernel-Ebene behandelt werden. So lassen sich selbst „virtuelle“ COM-Schnittstellen erzeugen. Weiterhin wird eine Reihe moderner Hochleistungs-Multifunktionskarten von National Instruments unterstützt, welche sich über digitale und analoge Ein- und Ausgänge ansprechen lassen.

Musterbeispiele für die praktische Anwendung von zeitkritischer Hardware-Kommunikation sind unter anderem die industrielle Bilderfassung über Kameras (GigE-Vision, USB3-Vision) oder anspruchsvolle Automatisierungsaufgaben mit EtherCAT.



Module



[Network Module](#)
[USB RealTime Module](#)
[USB Driver Module](#)
[Serial/UART Module](#)
[Device Module](#)
[IoPort Module](#)
[Memory Module](#)
[Interrupt Module](#)
[MultiFunction Module](#)

Network Module

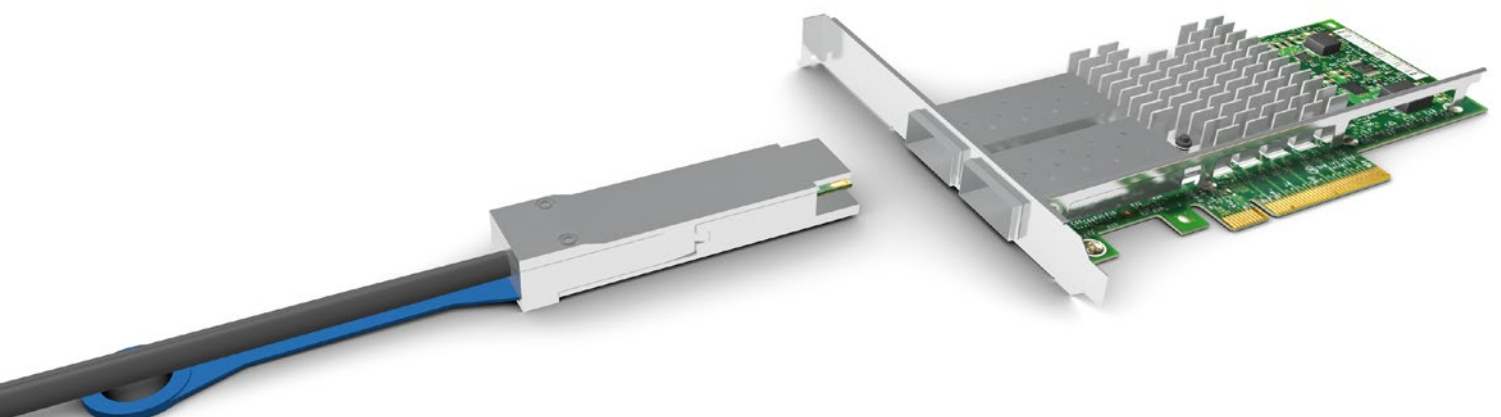
Native Echtzeit-Treiber für schnelle Ethernet-Kommunikation

Allgemein

Das Network Module stellt native Echtzeit-Ethernet-Treiber für die unterstützten Netzwerk-Controller bereit, sowohl für Kommunikation über Raw-Ethernet als auch UDP und TCP. Hierbei wird unter Umgehung sämtlicher behindernder Windows-Mechanismen mit hohen Übertragungsgeschwindigkeiten und unmittelbarer Reaktion für eine hohe Leistungsfähigkeit der Netzwerkkommunikation gesorgt. Unterstützt werden nahezu sämtliche Netzwerk-Controller von Intel und RealTek (siehe Hardware-Unterstützung).

Features

- Ethernet-Kommunikation bis 100 GBit/s in Echtzeit
- Verzögerungsfreies Senden und sofortige Reaktion beim Empfang
- Senden und Empfangen direkt aus dem Echtzeitkontext möglich
- Raw-Ethernet sowie verschiedene Ethernet-Protokolle, inklusive IP, UDP und TCP
- Ethernet-Kommunikation abfrage- oder ereignisgesteuert
- Jumbo-Frames bis 16128 Byte, je nach NIC
- IP- und MAC-Multicast, Broadcast, Promiscuous-Mode für Empfang aller Pakete
- Empfang von Datenpaketen entweder durch Callback-Funktion direkt im Interrupt-Kontext oder im Polling-Betrieb
- Priorisierung von zu sendenden Paketen in vier Stufen möglich
- Automatische Adressermittlung durch ARP-Unterstützung
- Funktionen für CRC-Berechnung und Byteorder-Wandlung enthalten
- Network Module Addons: Network Port Extension,
 - › Network PTM Extension, Network Data Rate Extension, Network BroadR-Reach Extension
- Erfordert ›RealTime Tasking Module



Zur Erreichung von Echtzeitfähigkeit wird der Standard-Windows-Treiber der Netzwerkkarte durch den Echtzeit-Treiber von Kithara RealTime Suite ersetzt. Unterstützt werden nahezu sämtliche derzeit verfügbaren Netzwerk-Controller von Intel und RealTek.

100 Gbit/s

Intel
■ E810

40 Gbit/s

Intel
■ XL710

25 Gbit/s

Intel
■ XXV710

10 Gbit/s

Intel
■ 82598, 82599
■ x540, x550
■ X710

1 Gbit/s

Intel
■ 82540, 82541, 82544, 82545, 82546, 82547
■ 82566, 82567
■ 82571, 82572, 82573, 82574
■ 82575, 82576
■ 82577, 82578, 82579
■ 82580, 82583, i350
■ i210, i211, i217, i219

RealTek

- 8100E, 8101E, 8102E, 8110S
- 8168B/8111B, 8168C/8111C, 8168CP/8111CP, 8168D/8111D+DL
- 8168DP/8111DP, 8168E/8111E, 8168EVL/8111EVL
- 8168F/8111F, 8168G/8111G, 8168H/8111H
- 8168EP/8111EP
- 8169, 8169S, 8169SB/8110SB, 8169SC/8110SC

ASIX – USB 3.0 Gigabit Ethernet Controller

- AX88179 (beispielsweise Digitus USB 3.0 Ethernet Adapter)

100 MBit/s

Intel
■ 8255x
■ 82562

RealTek

- 8139B, 8139C, 8139D

Beckhoff

- CX5000 CCAT(Beckhoff CX50xx, CX20xx, CX51xx Embedded PCs)

Erweiterungen

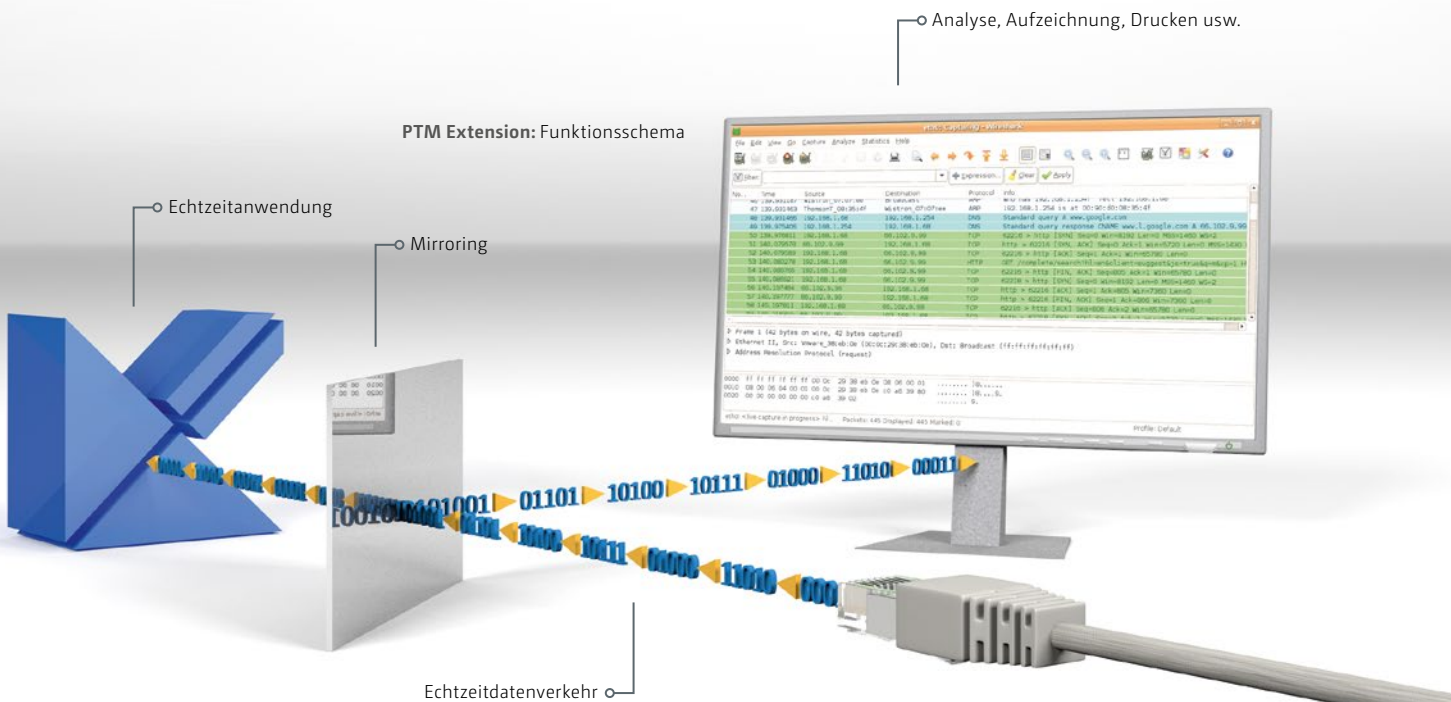
Network PTM Extension

Die Network PTM Extension ermöglicht die Generierung virtueller USB-Ethernet-Adapter über UDE (USB Device Emulation) für die Kommunikation zwischen Echtzeit und Windows via Port Passthrough oder Port Mirroring. Passthrough erlaubt es Windows, direkt mit der Hardware oder der Kithara-Echtzeitumgebung zu kommunizieren, während Mirroring den Echtzeit-Datenverkehr mitschneidet und Windows für Analysezwecke zur Verfügung stellt.

- Erstellung virtueller USB-Ethernet-Adapter über UDE
- Port Passthrough für direkte Kommunikation zwischen Windows und Hardware bzw. Echtzeit
- Port Mirroring für Mitschneiden des Echtzeitdatenverkehrs und Auslesen in Windows mit Wireshark
- Unterstützt alle Ethernet-basierten Technologien: Socket-Kommunikation, EtherCAT, GigE Vision, PTP und BroadR-Reach

Network BroadR-Reach Extension

BroadR-Reach (100BASE-T1) ist eine physische Ethernet-Übertragungsschicht speziell für die Automobilvernetzung.



USB RealTime Module

Echtzeit-Kommunikation mit USB-Geräten über direkten xHCI-Zugriff

Allgemein

Das USB RealTime Module von Kithara RealTime Suite garantiert, dass alle USB-basierten Schnittstellen und Geräte, wie etwa Kameras, mit schnellen Reaktionszeiten und größtmöglichem Durchsatz verwendet werden können. Dies wird durch einen speziell programmierten und einfach anwendbaren Zugang zum Extensible Host Controller Interface (xHCI) realisiert, welcher von Kithara RealTime Suite bereitgestellt wird. Grundprinzip dabei ist, den xHCI-Controller direkt aus dem Echtzeitkontext exklusiv anzusteuern.

Features

- Echtzeit-Kommunikation mit Geräten bis einschließlich USB 3.1 über xHCI-Zugang
- xHCI-Controller wird exklusiv genutzt
- Zugriff unmittelbar aus dem Echtzeit-Multitasking-Kontext
- Niedrigste Reaktionszeit
- Größtmöglicher Datendurchsatz erreichbar
- Zuverlässige Vermeidung von Datenausfällen durch frei wählbare Puffergrößen
- Low-, Full-, High- und Super-Speed
- Control-, Bulk-, Interrupt- und Isochron-Transfer
- Reaktion auf alle Plug-and-Play- und Power-Management-Ereignisse
- Erfordert [RealTime Tasking Module](#)

Hardware-Unterstützung

Im Zusammenhang mit dem USB RealTime Module können alle mit der xHCI-Spezifikation 1.1 kompatiblen USB 3.1 Hostcontroller eingesetzt werden.

xHCI-Controller folgender Hersteller sind derzeit mit Kithara RealTime Suite verwendbar: Intel, AMD, Renesas, Etron, NEC und VIA.

USB Driver Module

Ansprechen von USB-Geräten bis einschließlich USB 3.1 über Windows-Treiberstack

Allgemein

Das USB Driver Module von Kithara RealTime Suite erlaubt die Kommunikation mit USB-Geräten über einfach anwendbare Funktionen. Es basiert auf dem Windows-USB-Treiberstack und besitzt daher nur eingeschränkte Echtzeit-Eigenschaften. Für „harte“ Echtzeit empfehlen wir das [USB RealTime Module](#). Unterstützt werden alle Geräte bis einschließlich USB 3.1.

Features

- Entwicklung von USB-Gerätetreibern
- Zugriff auf eigene oder fremde USB-Geräte (bis einschließlich USB 3.1) von Anwendungs- oder Kernel-Ebene aus
- Low-, Full-, High-Speed
- Control-, Bulk-, Interrupt- und Isochron-Transfer
- Reaktion auf alle Plug-and-Play- und Power-Management-Ereignisse

Serial/UART Module

Einfach anwendbare Funktionen für schnelle Kommunikation

- Serielle Kommunikation in Echtzeit durch spezielle Hardware-Treiber auf UART-16550-kompatibler Hardware
- Signal- und Handshake-Leitungen direkt ansteuerbar
- Handler für alle Schnittstellen-Ereignisse in Echtzeit
- Für alle nicht echtzeitfähigen COM-Ports wird ein alternatives API mitgeliefert

Allgemein

Das Serial Module von Kithara RealTime Suite stellt einfach anwendbare Funktionen für die schnelle Kommunikation über die seriellen UART- sowie COM-Schnittstellen bereit. Die UART-Funktionen erfordern entsprechende, UART-16550-kompatible Hardware sowie einen vorliegenden Echtzeittreiber. Im Echtzeitbetrieb sind die UART-Funktionen auch aus Echtzeit-Tasks heraus und auf Dedicated-CPUs nutzbar.

Bei der COM-Kommunikation werden sämtliche Schnittstellen unterstützt, für die ein Windows-Treiber installiert ist. Da die COM-Funktionen so über Windows-Mechanismen laufen, können keine Echtzeiteigenschaften erreicht werden.

Features

- Serielle Kommunikation in Echtzeit auf der Kernel-Ebene durch speziellen Hardware-Treiber
- Handshake- und Signal-Leitungen direkt ansteuerbar
- Handler für alle Schnittstellen-Ereignisse können auch für Echtzeitkontext und Dedicated-CPUs angemeldet werden
- Für UART-Funktionen: UART-16550-kompatible Hardware erforderlich
- Für COM-Funktionen: serielle Kommunikation auf der Anwendungsebene erlaubt Programmierung aller im System bekannten COM-Schnittstellen

Zur Erreichung von Echtzeitfähigkeit für die gewünschte serielle COM-Schnittstelle müssen speziell entwickelte Echtzeit-Treiber von Kithara RealTime Suite verwendet werden. Diese unterstützen derzeit folgende serielle Schnittstellenkarten:

- Standard COM Schnittstelle (PNP0501)

PCIe

- StarTech PEX2S952, 2 × RS232
- Longshine LCS63210, 2 × RS232
- Delock 89220, 2 × RS232
- Delock 89236, 1 × RS232
- ADDI-DATA APcLe-7300, 1-fach seriell
- ADDI-DATA APcLe-7420, 2-fach seriell
- ADDI-DATA APcLe-7500, 4-fach seriell
- ADDI-DATA APcLe-7800, 8-fach seriell

ExpressCard

- I-Tec EXRS232, 1 × RS-232
- StarTech EC1S1P55254, 1-fach seriell, 1-fach parallel (Netmos 9912 Chipsatz)

PCI

- MOXA 4-port RS-232 C104H/PCI
- MOXA 8-port RS-232 C168H/PCI
- MOXA 2-port RS-422/485 CP132/PCI
- MOXA 4-port RS-422/485
- EXSYS 41052 serial card
- EXSYS 43092 serial card
- Oxford Serial Card Quad Port
- SUNIX 2-port RS-232 parallel
- SUNIX 2-port RS-232 PCI
- Meilhaus ME9000 8-port RS-485
- EXAR (XR17C152) 2-port UART PCI
- ADDI-DATA APCI-7300-3, 1-fach seriell
- ADDI-DATA APCI-7420-3, 2-fach seriell
- ADDI-DATA APCI-7500-3, 4-fach seriell
- ADDI-DATA APCI-7800-3, 8-fach seriell

CardBus

- Socket PCMCIA Serial Card
- Delock PCMCIA Serial Card

Device Module	Erzeugung virtueller Schnittstellen
Allgemein	<p>Das Device Module von Kithara RealTime Suite stellt dem Endanwender ein API zu Gerätetreibern über Betriebssystem-Funktionen bereit. Mit dem Device Module behandeln Sie Aufrufe dieser Funktionen auf der Kernel-Ebene und damit im Kontext des Gerätetreibers.</p> <p>Außerdem erlaubt das Device Module die Erzeugung von virtuellen Schnittstellen – in einer Sonderform sogar von virtuellen COM-Schnittstellen.</p>
Features	<ul style="list-style-type: none"> ■ Bereitstellung eines Windows-APIs für Gerätekommunikation (ReadFile, WriteFile, DeviceIoControl) ■ Beliebige Gerätenamen, zum Beispiel für virtuelle „COM-Ports“

Hardware-Zugriff

IoPort Module	Direkter Zugriff auf I/O-Ports
Allgemein	<p>Das IoPort Module von Kithara RealTime Suite ermöglicht den direkten Zugriff auf die I/O-Ports des PCs unmittelbar aus einer Windows-Applikation heraus. Mit Hilfe des IoPort Modules können Anwender die gewünschten I/O-Ports für den Direktzugriff freischalten.</p> <p>Der direkte Zugriff auf I/O-Ports bedeutet jedoch keineswegs eine Verringerung der Stabilität. Durch Wegfall der Entwicklung eines Kernel-Treibers kann sich der Programmierer voll auf seine Anwendung konzentrieren und gelangt nur über definierte Einsprungspunkte (des IoPort Modules) an die Hardware.</p>
Features	<ul style="list-style-type: none"> ■ Zugriff auf alle I/O-Register des PCs aus Anwendung oder DLL ■ Ermittlung von PCI-Konfigurationsdaten ■ Ermittlung von Ressourcen-Daten

Memory Module

Zugriff auf physischen Speicher

Allgemein

Das Memory Module von Kithara RealTime Suite ermöglicht einen direkten Zugriff auf den physischen Speicher.

Der physische Speicher wird in den Anwendungs-Adressraum eingeblendet, um somit den direkten Zugriff aus der Applikation heraus zu ermöglichen. Das Memory Module stellt zwei unterschiedliche Mechanismen bereit, um auf physischen Speicher zuzugreifen. Dabei wird unterschieden, ob sich der Speicher auf einer externen Hardware oder im PC-Hauptspeicher befindet.

Features

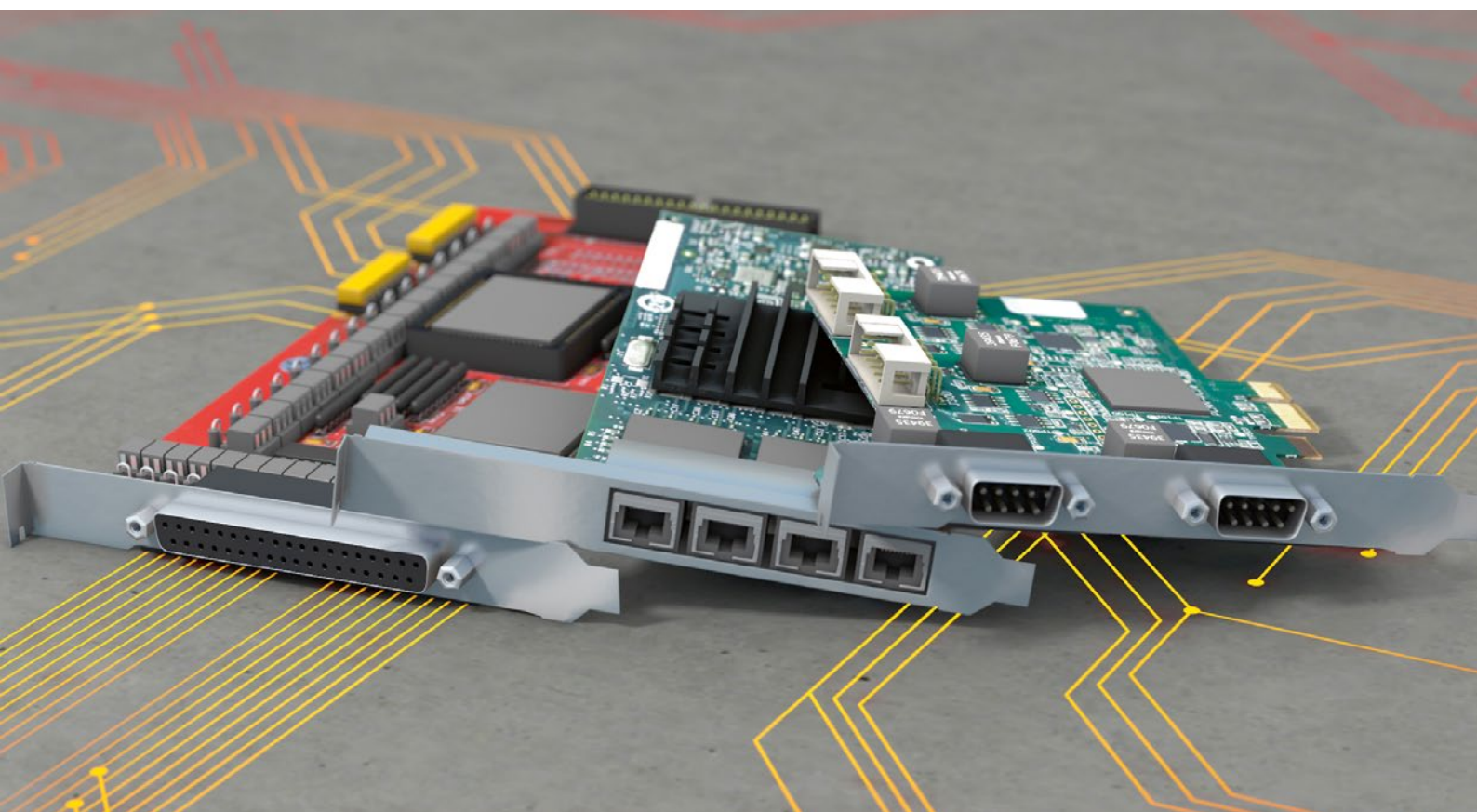
- Zugriff auf externen physischen Speicher (Dual-Port-RAM)
- Bereitstellung von PC-Speicher für externe Hardware (DMA-Speicher)

Interrupt Module

Steuerung externer Hardware

Allgemein

Zur Steuerung externer Hardware sind Interrupts unerlässlich. Mit ihnen teilen zum Beispiel PC-Erweiterungskarten das Erreichen bestimmter Zustände mit. Zur Erfassung dieser Ereignisse müssen Gerätetreiber und technische Applikationen einen entsprechenden Interrupt-Handler installieren.



Features	<ul style="list-style-type: none">■ Behandlung von Hardware-Interrupts auf der Anwendungs- oder Kernel-Ebene■ Nutzung der Betriebssystem-Mechanismen zur Interrupt-Behandlung■ Bei Vorliegen des RealTime Tasking Modules auch Echtzeit-Interrupts an PCI- oder PCIe-Hardware umsetzbar■ Benötigt IoPort Module
----------	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

MultiFunction Module	Zugriff auf Multifunktions-Erweiterungskarten in Echtzeit
-----------------------------	-----------------------------------------------------------

Allgemein	Unterstützung von Multifunktionskarten mit einem herstellerneutralen API, digital-I/O bit- oder wortweise, analog-I/O als Einzelwert, Kanalsequenz, limitierte Folge von Sequenzen oder Endlos-Modus mit Wechselpuffer-Interrupt.
-----------	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Features	<ul style="list-style-type: none">■ Unterstützung von Multifunktionskarten mit einem herstellerneutralen API■ Funktionen für A/D-, D/A- und Digital-I/O für Steuerung und Datenerfassung■ Digital-I/O bit- oder wortweise■ Analog-I/O als Einzelwert, Kanalsequenz, limitierte Folge von Sequenzen oder Endlos-Modus mit Wechselpuffer-Interrupt■ Echtzeit-Treiber für Karten verschiedener Hersteller■ Multifunktionskarten mit PCI- oder PCIe-Bus■ Erfordert RealTime Tasking Module
----------	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------



A blurred night street scene with bokeh lights and a road surface. The background is filled with out-of-focus lights in warm tones like orange, yellow, and red, suggesting city lights or traffic. In the foreground, a dark asphalt road with white dashed lane markings is visible, receding into the distance.

Automotive



Automotive



Die Automobiltechnik hat in den letzten Jahren eine rasante Entwicklung erfahren. Der Anteil von Software, zum Beispiel für Fahrzeugsicherheit, Fahrkomfort und Fahrerassistenzsysteme, steigt beständig an, und mit ihm auch die Anforderungen an Computertechnik und standardisierte Bussysteme zur Vernetzung der einzelnen Komponenten. Die erfolgreichsten Bussysteme sind CAN, CAN FD, FlexRay, BroadR-Reach und LIN, für die Kithara eine Reihe von Echtzeitfunktionen bereitstellt, um die Entwicklung solcher softwarebasierter Automobilsysteme zu unterstützen.

CAN (Controller Area Network) ist eines der erfolgreichsten Bussysteme in der Fahrzeugvernetzung, da es universell und flexibel einsetzbar ist. Die Erweiterung CAN FD (Flexible Data Rate) bietet zudem eine höhere Datenrate und mehr Informationen pro Nachricht.

FlexRay ist dank hoher Übertragungsraten über zwei Kanäle für noch datenintensivere Fahrfunktionen geeignet und wird neben CAN auch für sicherheitskritische Anwendungen eingesetzt.

Der auf UART basierende LIN-Bus wiederum erlaubt die Entwicklung besonders kostengünstiger Eindraht-Teilnetzwerke für einfachere Funktionen innerhalb von Automobil-Baugruppen und wird oft in übergeordnete Systeme wie CAN eingebunden.

Ethernet ist für viele Hersteller der nächste Schritt bei der zukünftigen Fahrzeugvernetzung. Die fortschrittliche Variante BroadR-Reach ist eine Ethernet-Übertragungsschicht, die speziell für diesen Bereich konzipiert wurde. Vorteilhaft sind die sehr hohe Übertragungsgeschwindigkeit, hohe Skalierbarkeit sowie Kosteneffizienz bei der Verkabelung und Programmierung.

Die Echtzeitfähigkeit dieser Busse ist vor allem bei Produktentwicklung und -integration, industrieller Fertigung sowie für Prüfstandentwickler in der Qualitätssicherung integrierter mechatronischer Systeme von Bedeutung, um diese PC-basiert unmittelbar zu steuern und zu diagnostizieren.

Kithara RealTime Suite stellt mit seinem modularen System verschiedene Mechanismen bereit, um für derartige Aufgabenstellungen eine perfekte Lösung zu bieten.

CAN/CAN FD

Das CAN Module ermöglicht Echtzeit für den Feldbus CAN (CAN-2.0B-Protokollkompatibel). Das API ist herstellerunabhängig und benötigt daher nur eine einmalige Einarbeitung, um jegliche unterstützte CAN-Hardware zu programmieren.

Die unmittelbare Reaktion auf eintreffende Nachrichten ermöglicht unter anderem, dass durch eine Callback-Funktion sofort auf Fehler-situationen reagiert werden kann. Selbst bei schnellen Baudraten und hohen Buslasten gehen keine Nachrichten verloren, da der Sende- und Empfangspuffer ausreichend groß angelegt werden kann. Ebenso ist es möglich, eigene Filterroutinen zu erstellen, die direkt zur Empfangszeit ausgeführt werden können.

Für Analyseaufgaben, bei denen sich das CAN-Interface völlig passiv verhalten muss, wird ein Listen-Only-Modus unterstützt. Außerdem kann bei Hardware mit SJA1000-CAN-Controllern das „Error Code Capture“-Register ausgelesen und somit weitere Informationen zur Fehleranalyse bereitgestellt werden.



FlexRay

Als Teil der Kithara-Echtzeiterweiterung stellt das FlexRay Module ein essenzielles Bindeglied zwischen Prüfsoftware und Prüfstand dar, mit dem sich zyklusgenaue Echtzeiteigenschaften für Hochpräzisions-Operationen umsetzen lassen. Mit dem Modul kann ein Windows-PC als vollwertiger FlexRay-Knoten verwendet werden. Dies beinhaltet die Verwendung als Leading- oder Following-Start-Node, also als aktiver sowie passiver Kommunikationspunkt innerhalb des Netzwerks. Beide FlexRay-Kanäle können unabhängig voneinander konfiguriert und die Baudrate flexibel angepasst werden (2,5/5/10 MBit/s).

BroadR-Reach

Mit Echtzeiteigenschaften für den BroadR-Reach-Standard (100BASE-T1, 1000BASE-T1) können Automobilentwickler physische Ethernet-Netzwerke in Fahrzeugen direkt ansteuern und testen. Durch garantierte Reaktionszeiten mit dem Kithara-Echtzeitsystem lassen sich Daten über die Ethernet-Übertragungsschicht präzise erfassen und in anspruchsvolle Testprozesse einbetten. So entsteht eine Brücke zwischen den leistungsfähigen Ethernet-treibern von Kithara RealTime Suite und der modernen Fahrzeugvernetzung.

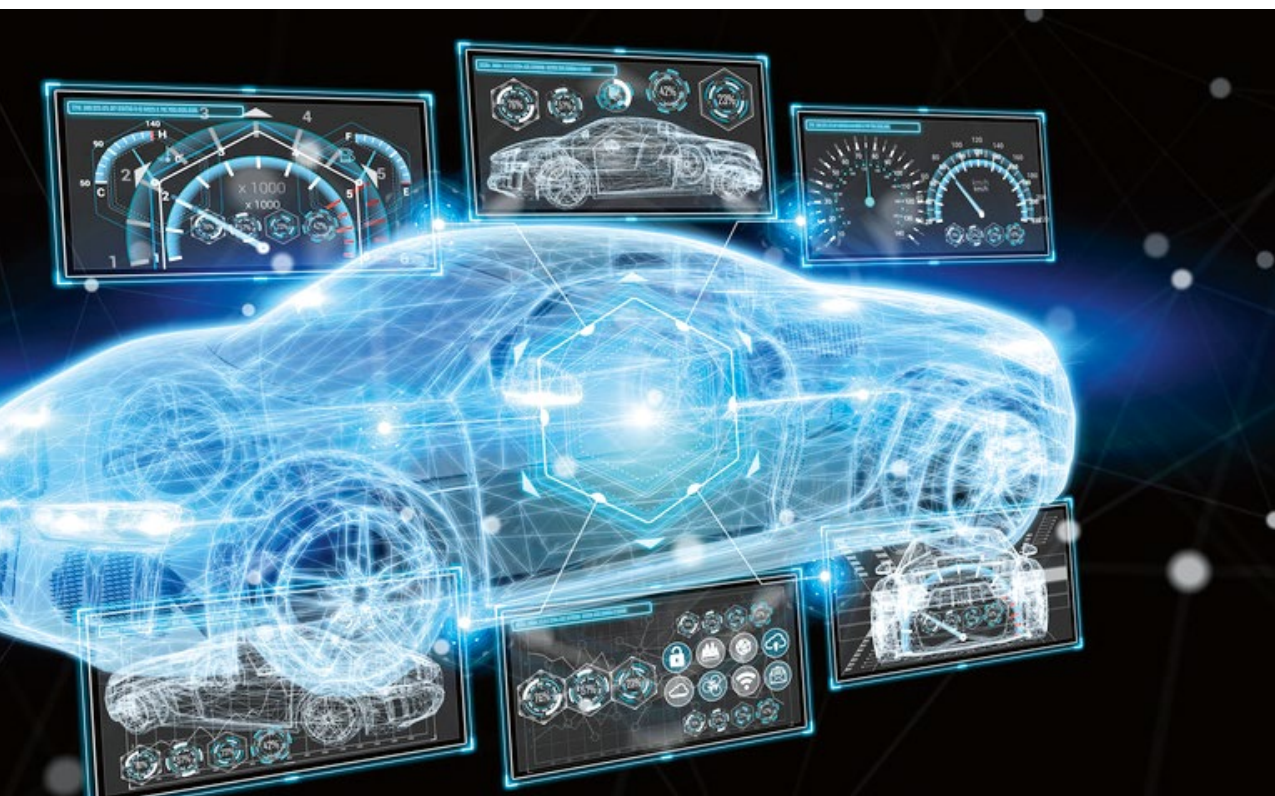
Viele der Kithara-Echtzeit-Funktionen kommen auch bei der Entwicklung neuer anspruchsvoller

voller Automobiltechnologien zum Einsatz. Sie finden beispielsweise Anwendung im Bereich Hardware-in-the-Loop, wo Simulationsdaten, dank Echtzeit, ohne Datenverlust präzise erfasst und wiedergegeben werden können.



LIN

Mit dem LIN Module von Kithara RealTime Suite ist es möglich, LIN in Echtzeit zu betreiben. Die Erstellung eines Master- oder Slave-Knotens wird über die UART-Schnittstelle eines handelsüblichen PCs realisiert. Anders als bei der Verwendung von Mikrocontrollern können existierende LIN-Netzwerke so einfacher protokolliert und Fehler effizienter behoben werden. Sendende Knoten überprüfen automatisch ihre übertragenen Daten, wodurch sich Kollisionen entdecken lassen.



CAN

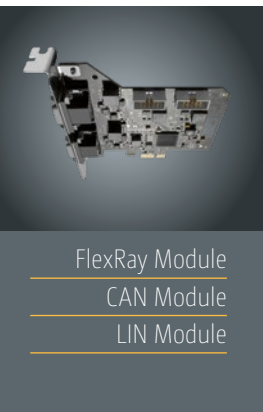
can^{FD}

lin
LOCAL INTERCONNECT NETWORK

FlexRay

BroadR_{xc}Reach

Module



FlexRay Module

CAN Module

LIN Module

FlexRay Module

FlexRay-Kommunikation in Echtzeit

Allgemein

Als Teil der Kithara-Echtzeiterweiterung stellt das FlexRay Module ein essenzielles Bindeglied zwischen Prüfsoftware und Prüfstand dar, mit dem sich zyklusgenaue Echtzeiteigenschaften für Hochpräzisions-Operationen umsetzen lassen.

Mit dem FlexRay Module kann ein Windows-PC als vollwertiger FlexRay-Knoten verwendet werden. Dies beinhaltet die Verwendung als Leading- oder Following Start Node, als aktiver sowie passiver Kommunikationspunkt innerhalb des Netzwerks. Beide FlexRay-Kanäle können unabhängig voneinander konfiguriert und die Baudrate flexibel angepasst werden (2,5/5/10 MBit/s).

Features

- Echtzeit-FlexRay-Kommunikation
- Unabhängig voneinander kombinierbare FlexRay-Kanäle
- Windows-PC als vollwertiger FlexRay-Knoten
- Verwendung als Leading- oder Following-Cold-Start-Node
- Baudrate flexibel anpassbar (2,5/5/10 MBit/s)
- Erfordert **RealTime Tasking Module**

Hardware-Unterstützung

Folgende FlexRay-Karten werden unterstützt:

- FlexCard PMC II (Star Cooperation)

CAN Module

CAN Kommunikation in Echtzeit

Allgemein

Das CAN Module ermöglicht Echtzeiteigenschaften für das serielle Bussystem CAN (Controller Area Network) bis einschließlich CAN 2.0B. Es erlaubt Anwendern innerhalb des Kithara-Echtzeitsystems, CAN-Messages verzögerungsfrei zu senden und zu empfangen. Benutzerdefinierte Filterroutinen lassen sich durch Callback-Funktionen auslösen. Auch die Signalisierung empfangener Nachrichten per Callback oder die Abfrage (Polling) von Messages sind möglich. Sende- und Empfangsroutinen können direkt aus dem Echtzeitkontext aufgerufen werden.

Zudem wird auch die CAN-Weiterentwicklung CAN FD (Flexible Data Rate) unterstützt, welche durch die Erweiterung der Nutzdatenlänge von 8 auf 64 Bytes eine bis zu 8-fach höhere Datenrate ermöglicht. Zusätzlich steigt durch Verbesserungen der Prüfwertbestimmung die potentielle Datensicherheit von Übertragungen, was die Einsatzmöglichkeiten für sicherheitskritische Anwendungen erhöht.

Für Analyseaufgaben, bei denen sich das CAN-Interface völlig passiv verhalten muss, gibt es einen „Listen-Only“-Modus. Des Weiteren kann bei Hardware mit SJA1000-CAN-Controllern das „Error Code Capture“-Register ausgelesen werden, sodass weitere Informationen zur Fehleranalyse bereitgestellt werden.

Features

- Echtzeit-Kommunikation über CAN bis 2.0B
- Optionale Unterstützung von CAN FD (Flexible Data Rates)
- Verzögerungsfreies Senden und sofortige Reaktion beim Empfang
- Verzögerungsfreies Senden und sofortige Reaktion beim Empfang mit Callback-Funktion oder asynchrone Nachrichtenabfrage (Polling)
- Erfordert **RealTime Tasking Module**

Hardware-Unterstützung

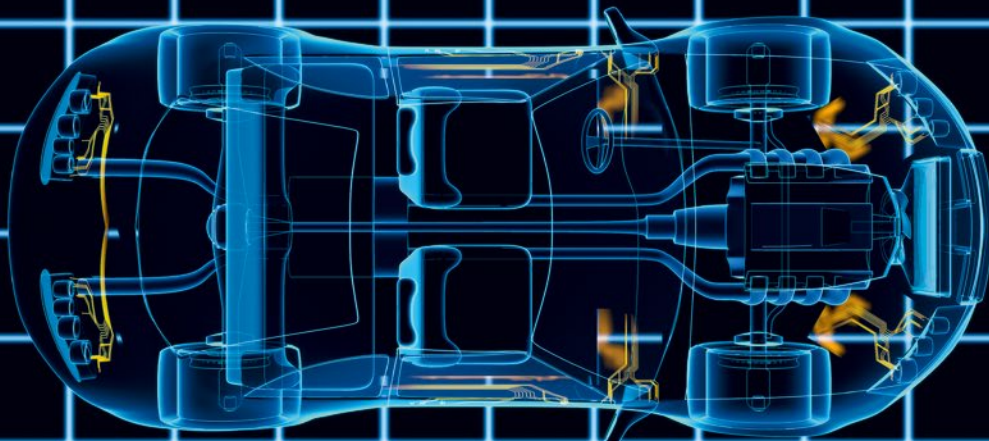
Das CAN Module unterstützt derzeit PCI-basierte Karten mit 1, 2 oder 4 Kanälen von Peak, Star Cooperation, EMS Dr. Wünsche, ESD, Ixxat und Kvaser (siehe Hardware-Kompatibilitätsliste). Gern unterstützen wir auch die von Ihnen verwendete Karte.

Peak System

- PCAN-PCI Express FD, Einkanal, Zweikanal, optoentkoppelt
- PCAN-PCI, Einkanal, Zweikanal, optoentkoppelt
- PCAN-PCI Express, Einkanal, Zweikanal, optoentkoppelt
- PCAN-miniPCI, Einkanal, Zweikanal, optoentkoppelt
- PCAN-cPCI, Zweikanal, Vierkanal, optoentkoppelt
- PCAN-USB, optoentkoppelt und PCAN-USB Hub
- PCAN-USB Pro, Zweikanal, optoentkoppelt

EMS Dr. Thomas Wünsche

- CPC-PCI, CPC-PCIe, Ein-, Zwei- und Vierkanal und CPC-104P mit einem SJA1000



ESD – Electronic System Design CAN-PCI

- CAN-PCI /200/266, optoentkoppelt, Ein- und Zweikanal
- CAN-PCIe /200, optoentkoppelt, Ein- und Zweikanal
- CAN-PCI-104 /200, optoentkoppelt, Ein- und Zweikanal
- CPCI-CAN /200, optoentkoppelt, Ein- und Zweikanal
- PMC-CAN /266, optoentkoppelt, Ein- und Zweikanal

Ixxat

- PC-I 04/PCI, Zweikanal

Kvaser Advanced CAN Solutions

- PCICan HS, Einkanal, Zweikanal, Vierkanal
- PCICanx HS, Einkanal, Zweikanal, Vierkanal
- PCIEcan HS, Einkanal, Zweikanal

Erweiterungen

CAN FD Extension

Echtzeit-Kommunikation über CAN FD (Flexible Data Rate)

LIN Module

Echtzeitkommunikation über LIN (Local Interconnect Network)

Allgemein

Das LIN Module von Kithara RealTime Suite ermöglicht LIN-Kommunikation in Echtzeit. Mit der Unterstützung der UART-Schnittstelle durch das **Serial Module** sowie einem geeigneten Pegelkonverter lassen sich Master- und Slaveknoten für LIN-Netzwerke erstellen. Anders als bei der Verwendung von Mikrocontrollern können existierende LIN-Netzwerke so einfacher protokolliert und Fehler effizienter behoben werden. Sendende Knoten überprüfen automatisch ihre übertragenen Daten, wodurch sich Kollisionen entdecken lassen.

Features

- LIN-Datenaustausch in Echtzeit
- Einfache Implementierbarkeit eines Master- oder Slave-Knotens
- Bus-Kollisionserkennung
- Automatische Prüfsummenvalidierung (LIN Version 1.x und 2.x)
- Datenraten bis zu 20 KBit/s
- Garantierte Latenzzeiten
- Empfang von LIN-Nachrichten ist durch Callbacks oder Polling möglich
- Zur Anpassung an die elektrischen Parameter des LIN-Busses wird ein LIN-Transceiver benötigt
- Erfordert **Serial Module** und **RealTime Tasking Module**

Hardware-Unterstützung

- UART (zur Verwendung von üblichen RS232-Schnittstellen ist ein Pegelkonverter erforderlich)
- Peak PCAN-USB Pro
- Weitere auf Anfrage

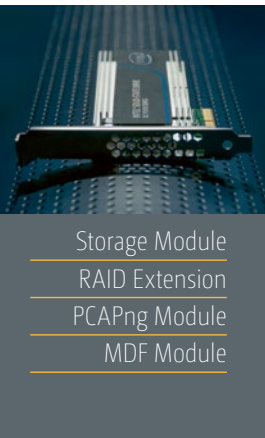


Unterstützung für PCAN-USB-FD-Adapter

Kithara RealTime Suite unterstützt auch den PCAN-USB-FD-Adapter von Peak Systems. Er ermöglicht die Anbindung an CAN-FD- und LIN-Netzwerke mit bis zu zwei Anschlüssen über die USB-Schnittstelle eines Computers.

Echtzeiteigenschaften für Schnittstellen wie CAN, CAN FD sowie FlexRay werden häufig bei Entwicklung und Anwendung von Automobil-Testsystemen eingesetzt. Hierbei können einzelne Komponenten oder ganze Hardware-systeme mittels extrem schneller und garantierter Reaktionszeiten unter Realbedingungen getestet und ausgewertet werden.

Echtzeitdatenspeicherung



In vielen Industriebereichen gewinnt Big Data zunehmend an Bedeutung. Zahlreiche Anwendungen profitieren bereits von den größer werdenden Datenmengen, und einige Lösungen lassen sich durch diese überhaupt erst umsetzen. Besonders Bereiche wie Mess- oder Prüftechnik können solche Daten effektiv nutzen, um hochpräzise Ergebnisse und Auswertungen zu erzielen.

Neben der Erfassung von Daten, beispielsweise von Industriekameras, ist auch die Speicherung ein kritisches Element für deren effiziente Verwendung. Dafür müssen Datensätze nicht nur verlustfrei bei hohen Durchsatzraten auf Speichermedien geschrieben werden, sie müssen sich oft auch genauso schnell wieder abrufen lassen. Aufbauend auf dem Storage Module befasst sich dieses Entwicklungsfeld von Kithara speziell mit der Speicherung von Daten in Echtzeit, um deren gewinnbringenden Einsatz in anspruchsvollen Projekten zu gewährleisten.

Das Storage Module erlaubt Echtzeit-Datenspeicherung auf SSDs mit NVMe-Schnittstelle. Auf diese Weise lassen sich Daten bei einer Durchsatzrate von mehreren Gigabyte pro Sekunde selbst im Dauerbetrieb sowohl schreiben als auch lesen. Mit UDF wird zudem ein leistungsfähiges Dateisystem bereitgestellt, wodurch auch Packet-Writing optimal eingesetzt werden kann.

Über RAID (Redundant Array of Independent Disks) lässt sich die ohnehin schon hohe Durchsatzrate sogar noch erhöhen, indem mehrere SSDs zusammengeschaltet werden. So kann die Datenrate um die Anzahl der geschalteten SSDs vervielfacht werden (RAID 0), mit Geschwindigkeiten im zweistelligen GByte/s-Bereich. Schreib- und Lesezugriffe finden dabei innerhalb von Kithara RealTime Suite weiterhin im Echtzeitkontext statt.

Um Datensätze auch strukturell effizient zu speichern, wird eine Echtzeitanbindung zu MDF (Measurement Data Format) bereitgestellt. Durch die Unterstützung von MDF 4.1 innerhalb der Echtzeitumgebung von Kithara lassen sich große Mengen an komplexen hierarchisch-strukturierten Messdaten unmittelbar speichern sowie abrufen. Dank der Echtzeitfähigkeit stehen erfasste Rohdaten damit lückenlos als MDF-Dateien für die weitere Anwendung zur Verfügung. Der De-Facto-Standard für Messaufgaben, unter anderem in der Automobilindustrie, besticht vor allem durch die schnelle und platzsparende Speicherung erfasster Daten. Das Format beinhaltet zudem leistungsfähige Funktionen für das effiziente Sortieren und Komprimieren von Datensätzen.

Des Weiteren kann das von Wireshark unterstützte Dateiformat PCAPng eingesetzt werden. In Verbindung mit Echtzeiteigenschaften lassen sich Datenpakete innerhalb von Netzwerken unmittelbar erfassen und auswerten. Dadurch steht mit Wireshark auch ein kostenloses und weit verbreitetes, komfortables Werkzeug zur späteren Analyse der gespeicherten Messdaten zur Verfügung.

Module

Storage Module

Echtzeit-Datenspeicherung auf NVMe-SSDs

Allgemein

Das Storage Module ermöglicht Echtzeit-Datenspeicherung über SSD-Laufwerke mit den jeweils aktuell angebotenen Kapazitäten. So lassen sich über die NVMe-Schnittstelle große Datenmengen mit Geschwindigkeiten von mehreren Gigabyte pro Sekunde schreiben beziehungsweise lesen. Als Dateisystem wird UDF bereitgestellt, wodurch packet-writing ermöglicht wird.



Features	<ul style="list-style-type: none"> ■ Echtzeit-Datenspeicherung über NVMe-SSD ■ Highspeed Lesen und Schreiben mit mehreren GByte/s ■ UDF-Dateisystem ■ Benötigt RealTime Tasking Module
Hardware-Unterstützung	<p>Im Zusammenhang mit dem Storage Module können alle mit der NVMe-Spezifikation-1.0-kompatiblen Speichermedien auf Anfrage verwendet werden. Folgende Geräte wurden getestet:</p> <ul style="list-style-type: none"> ■ Intel Solid-State Drive P3700/P3600/P3520/750 Series ■ Intel Solid-State Drive DC P3520 Series ■ Intel Solid-State Drive 600p Series ■ Samsung 950 SSD ■ HGST Ultrastar SN200 Series ■ VirtualBox Solid State Disk
Erweiterungen	<p>Storage RAID Extension Echtzeit-Datenspeicherung mit mehreren NVMe-SSDs im RAID-0-Verbund, multiplizierte Speicherkapazität und Durchsatzraten.</p>
RAID Extension	<p>Zusammenschaltung mehrerer SSDs mit RAID 0</p>
Allgemein	<p>Die RAID Extension erweitert das Storage Module um die Funktionalität zur Zusammenschaltung mehrerer SSDs sowie deren Echtzeitanbindung. Dadurch lässt sich die Durchsatzrate um die Anzahl der zusammengeschalteten SSDs vervielfachen.</p>
Features	<ul style="list-style-type: none"> ■ Echtzeitanbindung mehrerer zusammengeschalteter SSDs ■ Vervielfachung des Datendurchsatzes sowie der Speicherkapazität um die Anzahl der geschalteten SSDs ■ Datendurchsatz mehrere Gigabyte pro Sekunde bis in den zweistelligen Bereich ■ Unterstützung von RAID 0 ■ Stripe-Tiefe frei konfigurierbar ■ RAID-Disk über Pass-Through-Mechanismus bei Windows einblenden ■ Mit oder ohne externen RAID-Controller (beispielsweise High-Point SSD7101A-1) verwendbar ■ Erfordert Storage Module

PCAPng Module

Echtzeit-Datenspeicherung im PCAPng-Format

Allgemein

Das PCAPng Module (PCAP Next Generation) erlaubt die Echtzeit-Datenspeicherung von komplexen hierarchisch-strukturierten Messdaten im PCAPng-Format. PCAPng ist ein von Wireshark unterstütztes Dateiformat zum Erfassen von Datenpaketen innerhalb von Netzwerken. Wireshark kann deshalb zur späteren Analyse der gespeicherten Messdaten verwendet werden.

Features

- Echtzeit-Datenspeicherung im PCAPng-Format
- Wireshark zum Auslesen von gespeicherten Datensätzen, beispielsweise Mess- oder auch Bilddaten
- Benötigt [Storage Module](#)

MDF Module

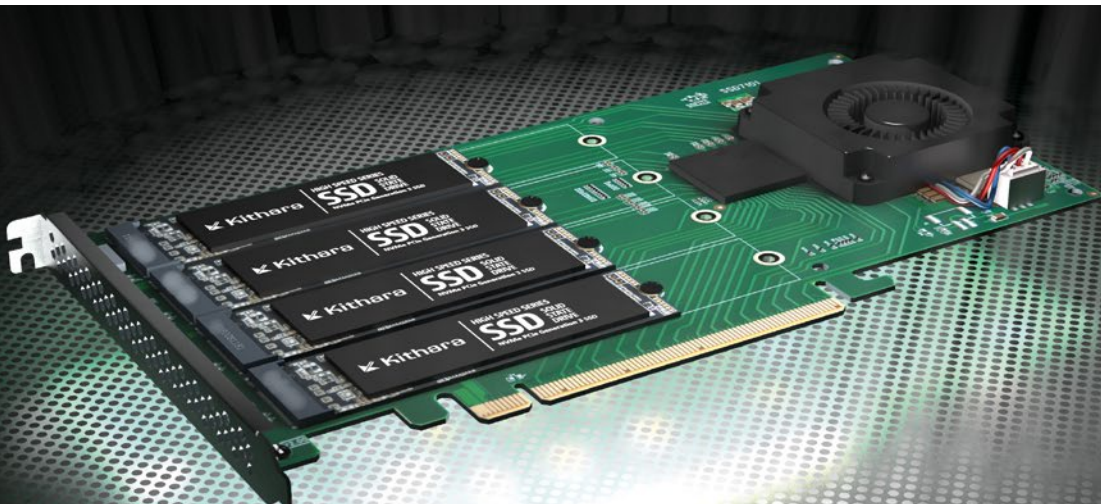
Echtzeit-Datenspeicherung in MDF

Allgemein

Das MDF Module erlaubt die Echtzeit-Datenspeicherung von komplexen hierarchisch-strukturierten Messdaten im Measurement Data Format. MDF 4.1 ist ein von ASAM bereitgestelltes, binäres Dateiformat zur späteren Auswertung oder Langzeitspeicherung erfasster und berechneter Datensätze. Die Größe der zu speichernden Dateien richtet sich nach der verwendeten Hardware (also auch im Terabyte-Bereich möglich).

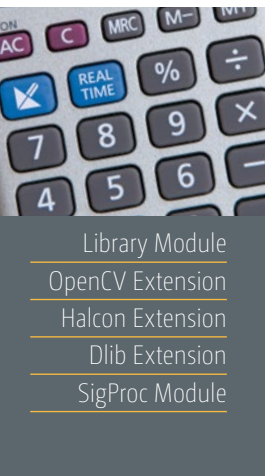
Features

- Echtzeit-Datenspeicherung in MDF
- Unterstützung von MDF 4.1, abwärtskompatibel zu früheren Versionen
- Schreiben nahezu beliebig großer Dateien (2^{64} Bytes)
- Speichern von Rohbotschaften der Busse CAN, LIN, FlexRay und Automotive Ethernet
- Benötigt [Storage Module](#)



Kithara-Echtzeit-Bibliotheken

Kithara empfiehlt, generell die echtzeitfähige Laufzeitbibliothek zu nutzen.



Library Module

Mathematische/trigonometrische und String-/Memory-Funktionen in Echtzeit

Allgemein

Für die Ausführung von Anwendungscode im Echtzeitkontext sind oftmals auch Funktionen der üblichen Runtime-Bibliothek erforderlich, die jedoch möglicherweise nicht für die Echtzeitausführung geeignet sind. Die Runtime Library stellt daher für Echtzeit angepasste Funktionen der C-Standardbibliothek bereit. Dazu gehören mathematische/trigonometrische sowie String-/Memory-Funktionen in Echtzeit.

Features

- Spezielle echtzeittaugliche Implementierung von Funktionen der C-Standardbibliothek
- Mehr als 20 mathematische/trigonometrische Funktionen (unter anderem sin, cos, tan, arcus- und hyperbolicus-Funktionen, exp, log, pow, sqrt, floor, ceil)
- Mehr als 20 String-/Memory-Funktionen (unter anderem memcpy, memset, strlen, strcmp, strcpy, strcat, strtok)
- Unterstützung von Anwendungs- und Kernel-Ebene
- Optional: Bildverarbeitung – Echtzeit mit Halcon oder OpenCV
- Optional: Machine Learning – Echtzeit mit Dlib



Erweiterungen

Dlib Extension

Dlib ist eine Open-Source-Programmbibliothek mit Algorithmen und Werkzeugen für die praktische Anwendung von Machine-Learning-Aufgaben. Die Bibliothek enthält verschiedene Softwarekomponenten, unter anderem für den Umgang mit linearer Algebra, maschinellem Lernen mit künstlichen neuronalen Netzen und Deep Learning. Durch die Dlib Extension lassen sich diese Funktionen im Echtzeitkontext von Kithara RealTime Suite nutzen.

- Dlib in Echtzeit
- Algorithmen und Tools für Machine Learning
- Funktionen für Datenstrukturen, lineare Algebra, maschinelles Lernen mit künstlichen neuronalen Netzen und Deep Learning
- Freie Programmbibliothek

OpenCV Extension

Echtzeit-Bildverarbeitung mit OpenCV

Allgemein

OpenCV ist freie Programmbibliothek mit Algorithmen für Bildverarbeitung und maschinelles Sehen. Sie verfügt über Algorithmen, unter anderem für Gesichtserkennung, 3D-Funktionalität, Haar-Klassifikatoren, verschiedene schnelle Filter, sowie Funktionen zur Kamerakalibrierung. Die OpenCV Extension ermöglicht die Anbindung der Programmbibliothek an das Kithara Echtzeitsystem. Dies ermöglicht eine sofortige Steuerungsreaktion auf verarbeitete Bilder sowie deren Einbindung in Automatisierungssysteme mit EtherCAT oder CANopen.



```

71
72 left := -1
73 right := 1
74 up := 2
75
76 for i := 0 to 256 by 1
77   read_image (Image, 'object_' + i$.04')
78
79   for j := 0 to KnownObj - 1 by 1
80     find_calib_descriptor_model (Image, Objects[j], [], [], [], [], 0.3, 2, CameraConf, 'num_points', Pose, Score)
81
82     if (|Score| > 0)
83       get_descriptor_model_points (Objects[j], 'search', 0, Row, Col)
84       gen_cross_contour_xld (Cross1, Row, Col, 6, 0.785398)
85
86       X := ROIY[j * KnownObj:(j + 1) * KnownObj - 1]
87       Y := ROIY[j * KnownObj:(j + 1) * KnownObj - 1]
88       pose_to_hom_mat3d (Pose, Mat3D)
89
90       affine_trans_point_3d (Mat3D, X, Y, [0,0,0,0], TansfX, TansfY, TansfZ)
91       project_3d_point (TansfX, TansfY, TansfZ, CameraConf, RowTransf, ColumnTransf)
92
93       gen_contour_polygon_xld (Contour, RowTransf, ColumnTransf)
94       close_contours_xld (Contour, Contour)
95
96       if ((Pose[5] > 45 and Pose[5] < 135))
97         flipObject(left)
98       elseif (Pose[5] > 225 and Pose[5] < 315)
99         flipObject(right)
100      elseif (Pose[5] > 135 and Pose[5] < 225)
101        flipObject(up)
102      else
103        * No action needed
104      endif

```

Features

- Echtzeit-Bildverarbeitung mit OpenCV
- Algorithmen unter anderem für Gesichtserkennung, 3D-Funktionalität, Haar-Klassifikatoren, Filter und Funktionen zur Kamerakalibrierung
- Automatische Parallelisierung
- Unmittelbare Kontrollreaktion auf verarbeitete Bilddaten zum Beispiel über EtherCAT oder CANopen
- Unterstützung von OpenCV 3.0, 3.4 und 4.1
- Erfordert **Library Module**

Halcon Extension

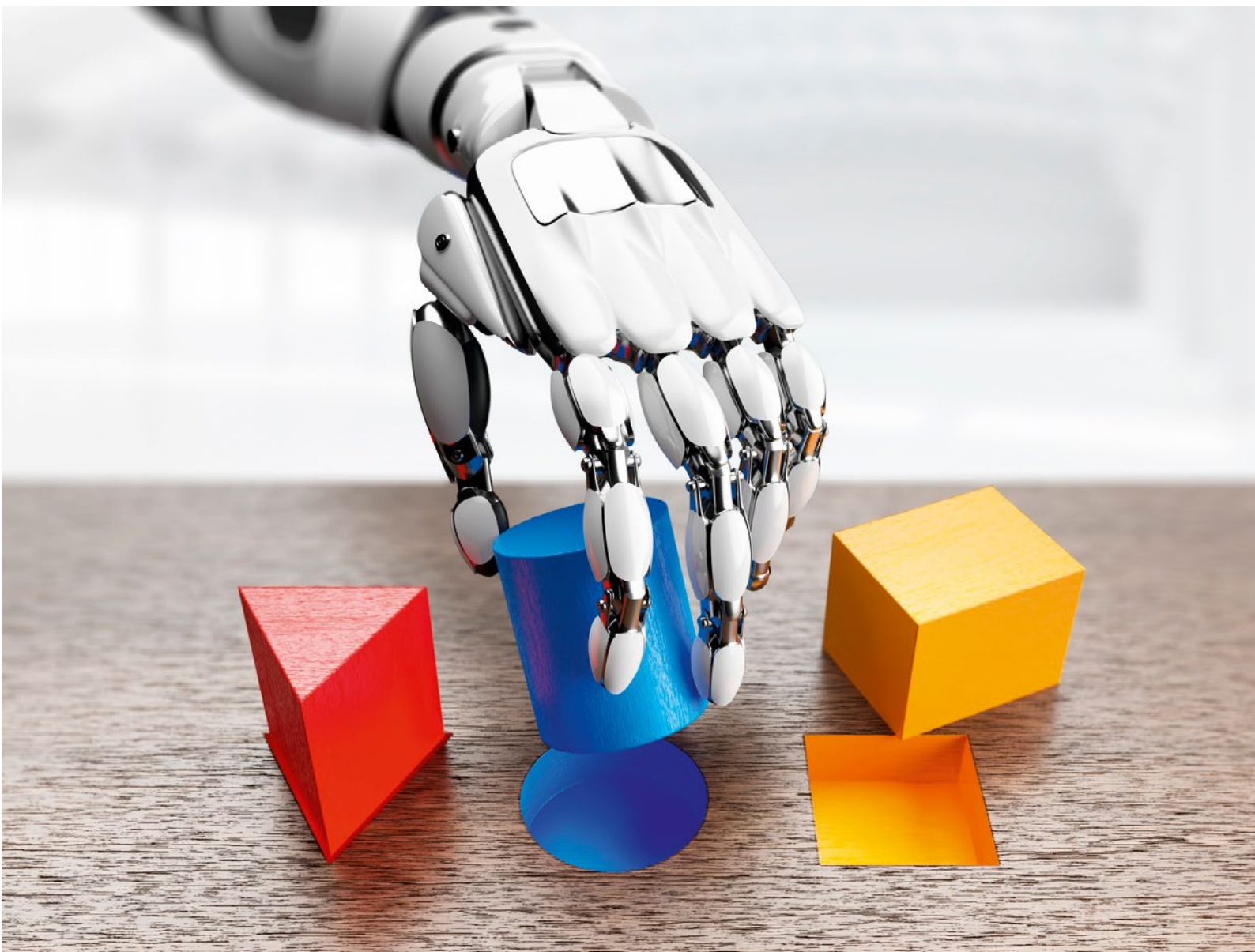
Echtzeit-Bildverarbeitung mit Halcon

Allgemein

Halcon des Münchener Unternehmens MVTec ist eine Programm-bibliothek mit Algorithmen für Bildverarbeitung und maschinelles Sehen. Sie verfügt über die eigene integrierte Entwicklungsumgebung HDevelop und über 2000 Operatoren. Die Halcon Extension ermöglicht die Anbindung der Programm-bibliothek an das Kithara Echtzeitsystem. Dies ermöglicht eine sofortige Steuerungsreaktion auf verarbeitete Bilder sowie deren Einbindung in Automatisierungssysteme mit EtherCAT oder CANopen.

Features

- Echtzeit-Bildverarbeitung mit Halcon
- Über 2000 verschiedene Operatoren
- Integrierte Entwicklungsumgebung HDevelop
- Automatische Parallelisierung
- Direktes Ausführen der HDevEngine aus dem Echtzeitkontext
- Unmittelbare Kontrollreaktion auf verarbeitete Bilddaten zum Beispiel über EtherCAT oder CANopen
- Unterstützung von Halcon Version 10 bis 13, 18.11 sowie 20.11
- Die Halcon-Lizenz ist gesondert bei der Firma MVTec zu erwerben
- Erfordert **Library Module**



SigProc Module

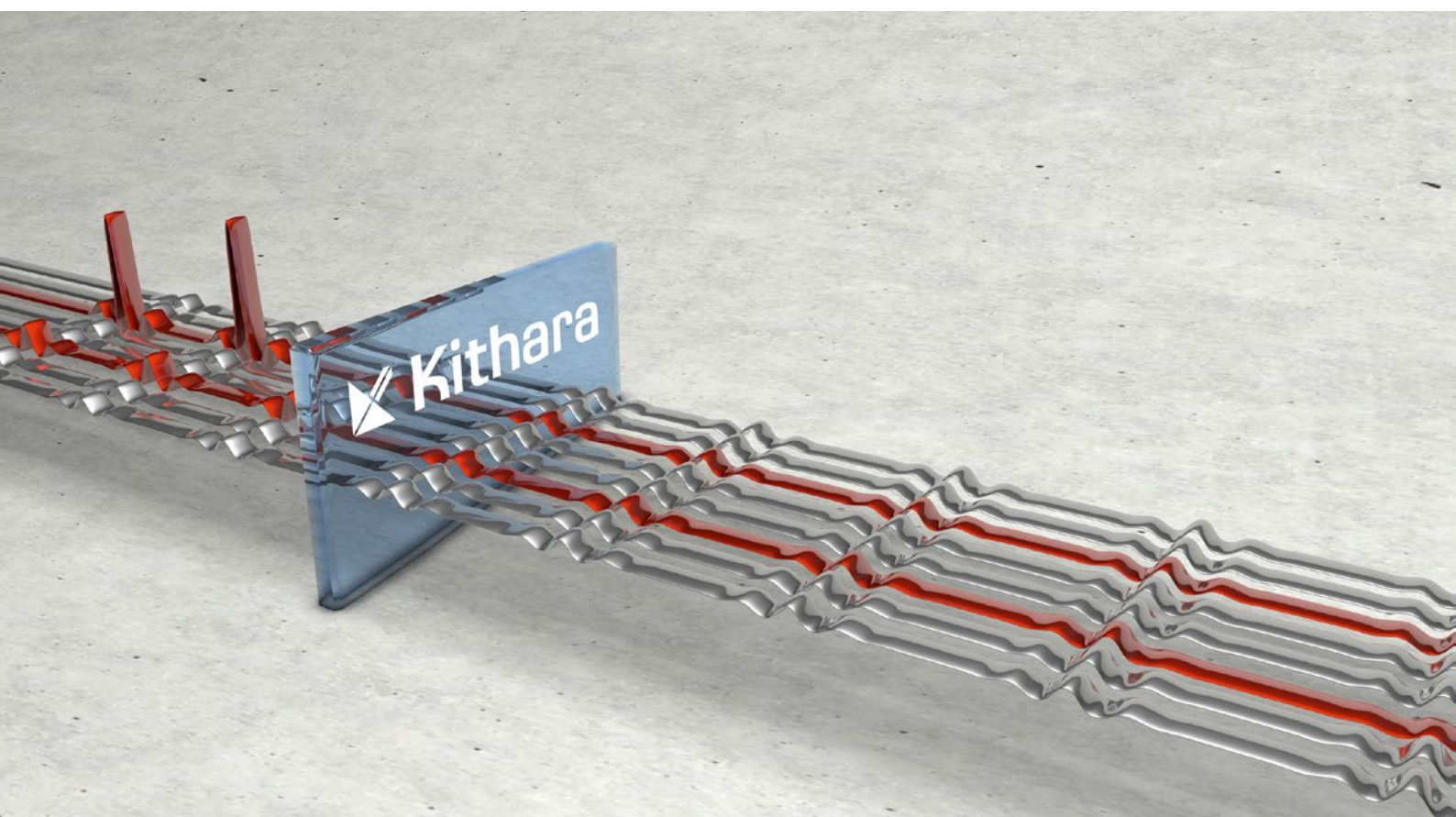
Digitale Signalverarbeitung in Echtzeit

Allgemein

Signalverarbeitung und Prozessregelung in Echtzeit werden oftmals für ein weites Spektrum von Ingenieur Anwendungen benötigt. Deshalb bietet das SigProc Module einen PID-Regler und eine Auswahl von digitalen FIR/IRR-Filtern mit verschiedenen Fensterfunktionen wie Hamming, Hanning und Rechteck sowie Transferfunktionspolynome wie Chebyshev I, Chebyshev II und Butterworth.

Features

- Digitale Signalverarbeitung in Echtzeit
- Mehr als 20 verschiedene Filter (Tiefpass, Hochpass, Bandpass, Bandsperre)
- 3 verschiedene Fensterfunktionen (Hamming, Hanning, Rechteck)
- 3 verschiedene Transferfunktionspolynome (Butterworth, Chebyshev I und II)
- Benutzerdefinierte, digitale FIR/IRR-Filter
- Prozessregelung in Echtzeit mit PID-Algorithmus



Kithara online



Alle Informationen rund um Kithara RealTime Suite sowie technische Unterstützung und die aktuellsten Neuigkeiten finden Sie im Internet unter [› kithara.com](https://www.kithara.com).

Entwicklungstools

Kithara Performance Analyzer

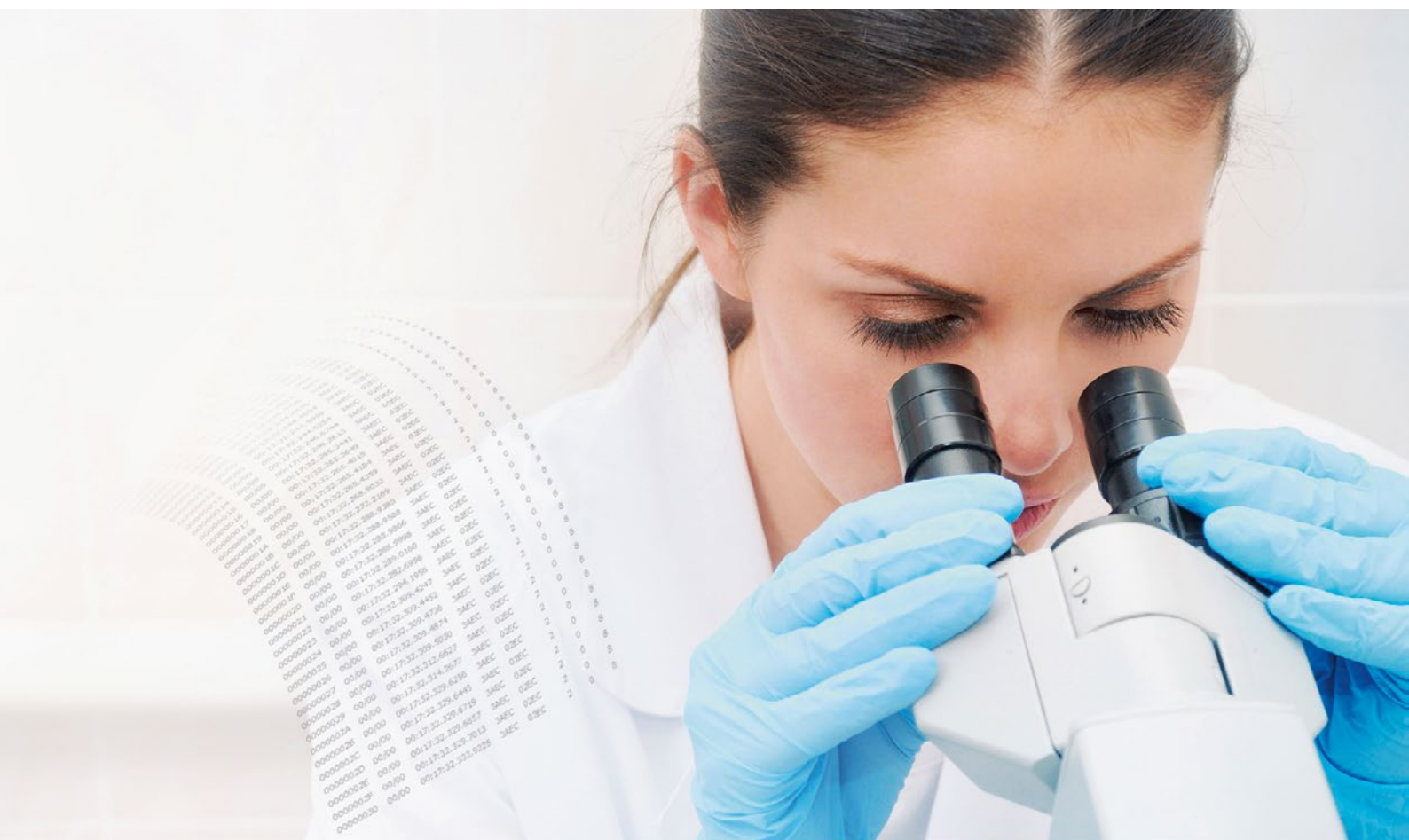
Systemressourcen-Visualisierung zur Diagnose der Rechnerleistung

Bei der Entwicklung von Echtzeitanwendungen werden spezifische Leistungsanforderungen an Entwicklungs- und Zielplattform gestellt. Mit Kithara Performance Analyzer lassen sich Echtzeitleistung und Systemintegrität in Verbindung mit den Funktionen von Kithara RealTime Suite visualisieren und effizient überwachen. Der Windows-Task-Manager bietet für die Programmentwicklung einen nur eingeschränkten Einblick in die Systemleistung und hat zudem keinen Zugriff auf die für Echtzeitfunktionen relevanten Werte. Kithara Performance Analyzer kann neben Systemressourcen wie Speicher- und CPU-Auslastung auch detaillierte Informationen zum Datendurchsatz aller verwendeten Schnittstellen abfangen, die innerhalb von Kithara RealTime Suite unterstützt werden. Dadurch lassen sich verschiedene Werte durchgehend erfassen und zeitparallel visualisieren, wodurch Wechselwirkungen und Problemquellen identifiziert werden können.

Kurzbeschreibung

Kithara Performance Analyzer ist ein entwicklungsnahe Visualisierungswerkzeug zur gleichzeitigen Darstellung und Überwachung verschiedener relevanter Systemleistungsressourcen. Es ist damit ein wichtiges Hilfsmittel zum Ermitteln der Echtzeitleistung eines Systems. Das interne Echtzeitsystem wird angewiesen, alle nötigen Informationen über Systemressourcen und Schnittstellen an Kithara Performance Analyzer weiterzugeben und in Echtzeit grafisch darzustellen. Der Benutzer entscheidet dabei, welche Werte dargestellt und in welcher Form sie angezeigt werden.

Datenübertragung und grafische Darstellung der Informationen führen zu keiner Leistungsbeeinträchtigung des Systems. Selbst bei einer großen Anzahl von wiedergegebenen Werten liegt die CPU-Belastung bei gerade mal ein bis zwei Prozent.



Eigenschaften von Kithara Performance Analyzer

Folgende Werte lassen sich mit Kithara Performance Analyzer auslesen und grafisch darstellen:

- Speicherauslastung für alle Speicherkategorien wie interner Echtzeitspeicher und Anwendungsspeicher
- CPU-Auslastung inklusive Task und Interrupt (einzelne CPU-Kerne separat darstellbar)
- Datendurchsatz (send & receive) aller von Kithara unterstützten Schnittstellen, inklusive Gigabit-Ethernet, USB 3/xHCI
- Auf den Schnittstellen basierende Protokolle: EtherCAT, CANopen, LIN, GigE Vision, USB3 Vision
- Jitter

Funktionen zur Visualisierung

- Zeitparallele Darstellung multipler Werte
- Flexible Zeitauflösung (15 Sekunden bis 2 Minuten)
- Lineare oder logarithmische Darstellung
- Graphische Oberfläche frei anpassbar – Visualisierungsfarben, Diagrammanordnung, Achsensgröße, Bezeichnungszuordnung und mehr



Kithara Kernel Tracer

Multi-Source Echtzeit-Message-Tracer und -Logger

Mit Kithara Kernel Tracer können Sie die internen Abläufe in Echtzeitanwendungen detailliert untersuchen. Während der Programmentwicklung werden leistungsfähige Werkzeuge, etwa zur Fehlersuche benötigt. Hierzu findet bisher meist ein Debugger Verwendung. Bei Multithread-Programmierung ist dieser jedoch oftmals nur eingeschränkt nutzbar, da er den Ablauf beeinflusst. Bei nebenläufigen Prozessen, mehreren Threads und vor allem bei Computern mit mehreren gleichzeitig arbeitenden CPUs (Multiprozessor- und Mehrkern-Systeme, Hyper-Threading) hängt das Ergebnis des Programmablaufs oftmals davon ab, ob die einzelnen Programmteile zeitlich parallel ausgeführt werden. Bei Verwendung eines Debugger ist dies nicht mehr sichergestellt:

- Der Programmablauf gerät bei Verwendung eines Debuggers durcheinander.
- Ein Programm verhält sich nicht wie erwartet, bei Einsatz des Debuggers funktioniert jedoch alles.
- Der Debugger ist nicht in der Lage, im Echtzeitkontext zu arbeiten.

Die Lösung ist ein Tracer-Werkzeug, mit dem der tatsächliche Ablauf der parallelen Programmteile protokolliert und anschließend ausgewertet werden kann.

Kurzbeschreibung

Kithara Kernel Tracer ist ein Multi-Source-Echtzeit-Message-Logger. Er dient der Fehlersuche, der Qualitätskontrolle und allgemein zum Aufspüren von Zusammenhängen, die ansonsten kaum zu ermitteln sind. Kithara Kernel Tracer sammelt garantiert alle Trace-Messages, die systemweit von verschiedenen Quellen stammen können, in der Reihenfolge ihres Auftretens ein, wobei diese normalerweise auf unterschiedlichen Seiten (Tabs) erscheinenden Messages auf einer Seite zusammengeführt werden können (merged).

Kithara Kernel Tracer ist in der Lage, mehrere hunderttausend Messages pro Sekunde zu verarbeiten. Um in dieser gewaltigen Datenmenge die entscheidenden Informationen zu finden, stellt das Werkzeug verschiedene Mechanismen bereit, um Messages zu filtern oder zu suchen. Es erlaubt eindrucksvolle Einsichten in die tatsächlichen Abläufe im Inneren eines PCs, zum Beispiel auf der Kernel-Ebene. Neben der Darstellung sämtlicher Aufrufe von Kithara-Funktionen mit all ihren aktuellen Funktionsargumenten können Sie auch selbst Messages erzeugen, die dann von Kithara Kernel Tracer mit vielen hilfreichen Zusatzinformationen dargestellt werden.

Eigenschaften von Kithara Kernel Tracer im Detail

- Multi-Source: Unterstützung beliebig vieler Threads und Programme aus Anwendungs- und Kernel-Ebene (Echtzeit)
- Multi-Core: Zeitlich korrekte Zuordnung der Messages bei mehreren CPUs (SMP, Hyper-Thread, Multi-Core etc.)
- Hohe Verarbeitungsgeschwindigkeit (mehrere hunderttausend Messages pro Sekunde möglich, sofern Leistungsreserve vorhanden)
- Alle Funktionsaufrufe von Kithara RealTime Suite werden mit Funktionsargumenten dargestellt
- Anwenderspezifische Messages können mit den Tools von Kithara RealTime Suite generiert werden (printf-ähnlich mit C/C++)
- Messages werden mit vielen Details dargestellt, inklusive der Systemzeit in 0,1-Mikrosekunden-Auflösung
- Messages lassen sich filtern und zusammenführen
- Erlaubt das Speichern und Laden der empfangenen Nachrichten zur Offline-Analyse
- Filterung erlaubt: dauerhaftes Löschen, vorübergehendes Ausblenden, Fett-, Kursiv- und farbige Darstellung von Messages
- Farbige Darstellung erlaubt Hervorhebung von Fehlermeldungen und Warnungen
- Filterung der Messages nur von einer Quelle oder global
- Weitere Kontextdetails: logische CPU, Thread-/Prozess-ID, Echtzeit-Task-ID, laufende(r) Message-Nummer und -typ, Thread- oder Task-Priorität, Inhalt von Datenblöcken (bei ausgewählten Messages)
- Suchfunktion für Messages nach Text, Zeitpunkt und allen anderen Feldern
- Message-Masken verhindern unnötige Generierung nicht benötigter Messages
- Messages werden auch über BSOD hinweg gespeichert (Crash-Dump wird automatisch gescannt)
- Unterschiedliche Einfärbung verschiedener Prozesse, Threads und Tasks zum besseren Auffinden
- Messages lassen sich nachträglich ändern (Text, Vorder-/Hintergrundfarbe) zur Dokumentation
- Programmierbare Filter zu- und abschaltbar
- Verschiedene Datenspalten ein- und ausblendbar
- USB-, UART-, CAN- und Netzwerk-Geräte lassen sich speziell beobachten
- Memory-Viewer zum Ansehen von physischem Speicherinhalt
- PCI-Viewer zum Analysieren von PCI-Geräten
- Mitgeliefertes SDK ermöglicht Programmierung eigener Trace-Quellen

Anwenderspezifische Meldungen

Sie können eigene Meldungen an den Tracer übergeben, beispielsweise um Variableninhalte darzustellen oder das Eintreten besonderer Programmzustände zu signalisieren. In den Tools von Kithara RealTime Suite stehen dafür die Funktionen `KS_logMessage` und `KS_vprintK` zur Verfügung. `KS_logMessage` kann von allen unterstützten Sprachen genutzt werden, `KS_vprintK` verwendet eine variable Argumentliste und ist daher von C/C++ aus verwendbar.

Hinweis

Die Funktionalität von Kithara RealTime Suite in Verbindung mit spezifischen Komponenten ist, aufgrund verschiedenster Hardware-Variationen, nur unter Vorbehalt gewährleistet. Für Fragen bezüglich der Kompatibilität mit einzelnen Hardware-Konfigurationen kontaktieren Sie uns bitte. Eine Liste mit von Kithara unterstützter Hardware erhalten Sie gern auf Anfrage. Alle Angaben beziehen sich stets auf die aktuelle Version unserer Software.

Notizen



Vertrieb

Für alle Fragen rund um den Vertrieb wenden Sie sich bitte per E-Mail an info@kithara.com oder telefonisch an +49 30 2789673-0.

Support

Den **technischen Support** erreichen Sie per E-Mail an support@kithara.com.

Kithara RealTime Suite

Jetzt kostenlos testen!



Anwendungen in
32/64 Bit
auf einheitlicher Codebasis

Auf los geht's los
Quick Start
Sofort verwendbare Programmgerüste

Voll funktionsfähige Testversion
90 Tage
kostenlos testen

Funktionsmodule beliebig
kombinierbar
Nur wenige Abhängigkeiten

Installation jederzeit
erweiterbar
um zusätzliche Module

Dank Sonderkonditionen
bezahlbar
Günstige Runtime-Lizenzen, je nach Bedarf



Referenzen

Folgende Unternehmen und Institutionen gehören zu unseren Kunden:



Einige unserer Partner:



Bei weiteren Fragen zu Eigenschaften, unterstützter Hardware, Betriebssystemen oder Programmiersprachen besuchen Sie bitte unsere Website unter [kithara.com](https://www.kithara.com). Wir empfehlen den Download einer Test- beziehungsweise Evaluierungsversion. Bitte kontaktieren Sie uns bei allen Fragen zu Ihrem Projekt!



QR Code | Kithara im Internet: