

## sysWORXX: Ihre Profi-Vorteile auf einen Blick



### Yocto Linux für industrielle Umgebungen

- Unterstützung langer Wartungszyklen, reproduzierbar und sicher
- vollständige Kontrolle über Updates und Sicherheits-Patches
- Unabhängigkeit von Upstream-Änderungen (im Gegensatz zu Debian)
- nutzerbasiertes Rechte-Management, auch ohne Root-Zugang nutzbar
- sichere Over-the-Air Updates dank RAUC
- optimiert für industrielle Anwendungen im Bereich Embedded Systeme



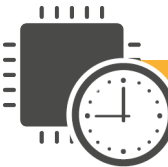
### Cyber Resilience Act (CRA) ready

- Minimierung der Angriffsfläche durch vollständige Systemkontrolle
- Voraussetzung für Zertifizierbarkeit (TÜV, VSD, ...)



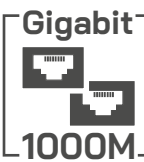
### Serielle Konsole für netzwerkunabhängige Inbetriebnahme

- einfacher Anschluss über USB
- Serielle Konsole auch ohne gültige IP-Adresse nutzbar
- Fehlkonfigurationen der ETH-Schnittstelle führt nicht zum Zugriffsverlust auf das Gerät



### On-board RTC

- gültige Systemzeit schon beim Booten
- autarke Systemzeit unabhängig von einer Internetverbindung



### ETH: 2x 1000 Mbits/s

- 2 separate Interfaces mit jeweils eigener MAC-Adresse für Nutzung in getrennten Netzen
- geeignet für industrielle Gigabit-Netzwerke



### Freie Wahl des Boot-Mediums



**SD-Karte**

- einfaches Backup vor Konfigurations- oder Softwareänderungen
- gefahrloses Testen von Konfigurationseinstellungen und Softwarekomponenten



**eMMC**

- Bring-up Skript zur einfachen Übernahme von SD-Card Images in den eMMC

- robuster, langlebiger Massenspeicher für 24/7 Dauerbetrieb
- keine mechanischen Kontaktprobleme

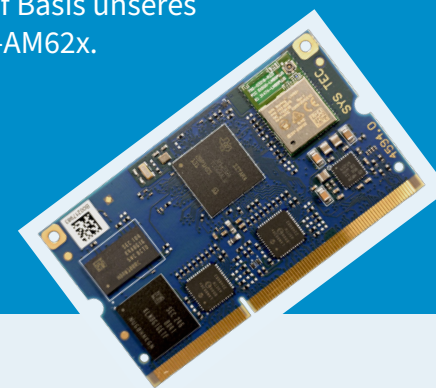
## SOFTWARE LIEFERUMFANG

	Basislieferumfang	Option
Linux OS	✓	
C/C++	✓	
C# (.NET Core)	✓	
Java (OpenJDK)	✓	
Python	✓	
Node-RED	✓	
Docker	✓	
CODESYS (IEC-61131)	✓	Lizenz abhängig von genutzten Programm-Ressourcen
CANopen (unabhängig von CODESYS)		
Source Code CANopen Device	-	SO-877
Source Code CANopen Manager	-	SO-1063
OPC-UA (unabhängig von CODESYS)	-	3rd-Party
Meshnet Software	-	3rd-Party
qBee Device Management	-	3rd-Party
MQTT	[✓] (in CODESYS und Node-RED integriert)	Open-Source Libraries frei verfügbar
Modbus (unabhängig von CODESYS)	-	Open-Source Libraries frei verfügbar

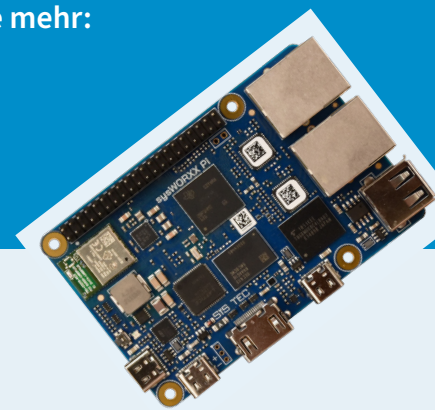


### Individuelle Lösung gesucht?

Effizient, verlässlich und vielseitig - der **sysWORXX ECUcore-AM62x** ist das leistungsstarke Herzstück unserer CTR-Steuerungen. Sie benötigen etwas Einzigartiges? Wir entwickeln und fertigen Ihre spezifische, auf Sie abgestimmte Elektroniklösung auf Basis unseres sysWORXX ECUcore-AM62x. **Erfahren Sie mehr:**



Ready, Set, Code: Das **sysWORXX Pi-AM62x**, ein Single Board Computer (SBC), ist die leistungsstarke Lösung für Ihre vorgelagerte Softwareentwicklung. Es bietet Ihnen die Möglichkeit, das Zusammenspiel zwischen Hard- und Software zu testen sowie das geplante Systemkonzept vorab zu validieren. **Erfahren Sie mehr:**



**Wir sind für Sie da – Kontaktieren Sie uns!**

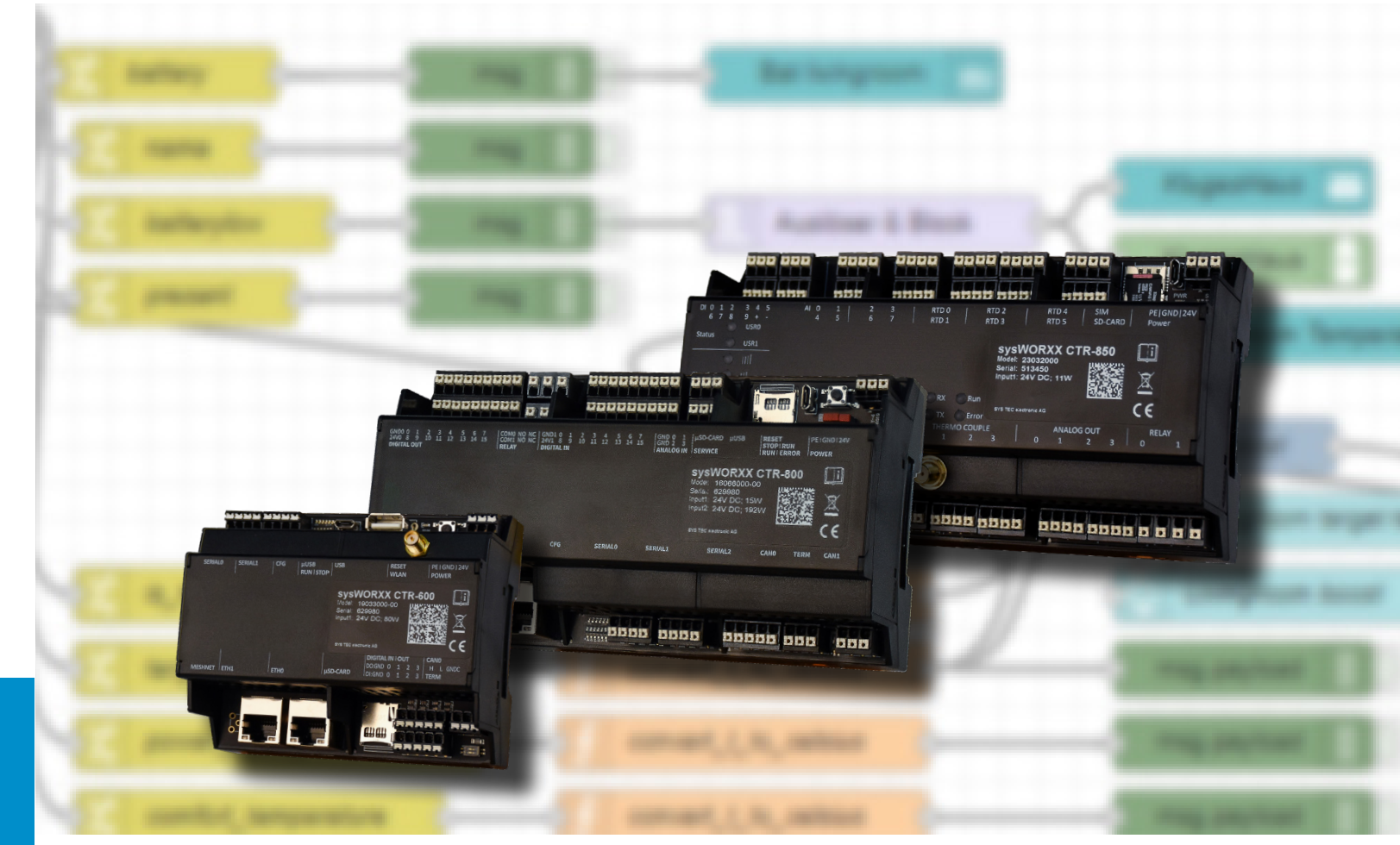
Unsere freundlichen Mitarbeiter helfen Ihnen gerne:



sales@systec-electronic.com | 03765 - 38600 - 2110 | www.systec-electronic.com



**SYS TEC**  
ELECTRONIC



## UNSER PRODUKT-PORTFOLIO FÜR SMARTE AUTOMATION

Mit unserer Controller Familie verknüpfen wir Welten: Von Sensor zu Cloud, Maschine zu Maschine und Maschine zu Mensch. Setzen Sie Ihrer Automatisierung keine Grenzen und entdecken Sie unsere Lösungen...

### sysWORXX CTR-850



#### SMARTE KOMPAKT-STEUERUNG FÜR CONDITION MONITORING

- frei programmierbare Linux Kompakt-Steuerung für Automatisierung und Digitalisierung
- spezifische Ein/Ausgänge zur Nutzung im Bereich Condition Monitoring
- erweiterbar mit dem sysWORXX SSM-100 für Körperschallanalyse
- ortsunabhängige Internet-Anbindung über integriertes LTE Modem
- programmierbar als SPS in IEC 61131-3 in der Programmierumgebung CODESYS
- einfache Applikationserstellung in Low-Code Umgebung Node-RED
- Unterstützung gängiger High-Level Programmiersprachen von C/C++ über Java bis Python

Mehr Details?  
<https://www.systec-electronic.com/produkte/industrial-internet-of-things/sysworxx-ctr-850>



### sysWORXX CTR-810



#### KOMPAKT-STEUERUNG FÜR EFFIZIENTE VORKONFEKTIONIERUNG

- frei programmierbare Linux Kompakt-Steuerung für Automatisierung und Digitalisierung
- vollständig kompatibel zum sysWORXX CTR-800
- die abziehbaren Steckverbinder ermöglichen eine effiziente Vorkonfektionierung der Anschlusskabel (Kabelbaum) sowie einen einfachen und schnellen Gerätetausch im Feld
- Basistechnologie für Ihre Entwicklung
- Brand-Labeling möglich
- programmierbar als SPS in IEC 61131-3 in der Programmierumgebung CODESYS
- einfache Applikationserstellung in Low-Code Umgebung Node-RED
- Unterstützung gängiger High-Level Programmiersprachen von C/C++ über Java bis Python

Mehr Details?  
<https://www.systec-electronic.com/produkte/industrial-internet-of-things/sysworxx-ctr-810>



### sysWORXX CTR-800



#### SMARTE KOMPAKT-STEUERUNG FÜR ALLE ANFORDERUNGEN

- frei programmierbare Linux Kompakt-Steuerung für Automatisierung und Digitalisierung
- vielseitig nutzbare digitale und analoge Ein/Ausgänge
- einsetzbar als Feld-Bus Gateway, Edge Controller und Kompakt-Steuerung
- einfache Applikationserstellung in Low-Code Umgebung Node-RED
- programmierbar als SPS in IEC 61131-3 in der Programmierumgebung CODESYS
- Unterstützung gängiger High-Level Programmiersprachen von C/C++ über Java bis Python
- Basistechnologie für Ihre Entwicklung
- Brand-Labeling möglich

Mehr Details?  
<https://www.systec-electronic.com/produkte/industrial-internet-of-things/sysworxx-ctr-800>



### sysWORXX CTR-600



#### DIE SMARTE GATEWAY-LÖSUNG

- frei programmierbares Linux Gateway
- einsetzbar als Feld-Bus Gateway, Edge Controller und Kompakt-Steuerung
- einfache Applikationserstellung in Low-Code Umgebung Node-RED
- programmierbar als SPS in IEC 61131-3 in der Programmierumgebung CODESYS
- Unterstützung gängiger High-Level Programmiersprachen von C/C++ über Java bis Python
- Low Cost Steuerung (z.B. zum Auslesen von Smart-Metern über Modbus)
- Meshnet-Router und Meshnet-SinkNode
- unterstützen Nordic-SDK (nRF5)
- Basistechnologie für Ihre Entwicklung
- Brand-Labeling möglich

Mehr Details?  
<https://www.systec-electronic.com/produkte/industrial-internet-of-things/sysworxx-ctr-600>



## PRODUKTÜBERSICHT

	CTR-600	CTR-800	CTR-810	CTR-850
SoM - Core	TI Sitara AM623 Dual Core (2x Cortex®A53 @ 1,4 GHz), 1024 MiB RAM, 8 GiB eMMC			
Kundenspezifische Anpassungen	Single Core/Quad Core, RAM bis 8 GB, bis 64 GB eMMC			
Spannungsversorgung	24V DC	24V DC	24V DC	24V DC
Gehäuse	6 TE	9 TE	9 TE	9 TE
Backplane Anschluss für Erweiterungsmodule	Nein	Ja	Ja	Nein
Ethernet	2 x 1.000 Mbps	2 x 1.000 Mbps	2 x 1.000 Mbps	2 x 1.000 Mbps
WLAN	Integriert mit externer Antenne	Optional via USB-Stick	Optional via USB-Stick	Optional via USB-Stick
USB 2.0 Host	1	1	1	1
CAN Schnittstellen	1 x CAN 2.0B	2 x CAN 2.0B	2 x CAN 2.0B	-
Serielle Schnittstellen (konfigurierbar RS-232, RS485/Modbus)	2	3	3	1
Meshnet	Optional: Integriert mit externer Antenne	Optional: via RFG-2.4	Optional: via RFG-2.4	-
LTE Modem	-	-	-	Integriert mit externer Antenne
Digital Input (24V DC)	4	16	16	10
Digital Output (24V DC, 0,5A)	4	16	16	-
Analog Input (konfigurierbar 0...10V DC, 0...20 mA DC)	-	4	4	8
Analog Output (konfigurierbar 0...10V DC, 0...20 mA DC)	-	-	-	2/2
Temperaturfühler	-	-	-	-
PT 100	-	-	-	6
Thermoelement (Typ K)	-	-	-	4
Relais Output (max. 220VDC, 60W / 250VAC, 62.5VA)	-	2	2	2
Betriebssystem	Yocto Linux, Armbian			
Softwareunterstützung	C/C++, Rust, C# (.NET Core), Java (OpenJDK), Python, Node-RED, CODESYS (IEC-61131)			

## SOFTWARE-ÖKOSYSTEM

Von codebasiert bis grafisch: Maximale Individualität. Smart vernetzt. Flexibel deployen – isoliert und portabel.

**Node-RED Editor**

**Node-RED GUI**

**Structured Text (ST)**

```

CASE ProcStep OF
  1: MotorEnable := TRUE;
  2: MotorSpeed := 100;
  1: MotorSpeed := 500;
END_CASE;
IF (Error <> 0) THEN
  MotorEnable := FALSE;
END_IF;
        
```

**Function Block Diagram (FBD)**

**Sequential Function Chart (SFC)**

**Ladder Diagram (LDD)**

C#

python

Java

.NET

**Local In-/Out**

- Digital In
- Digital Out
- Counter
- PWM
- Analog In

**CANopen**

**Modbus**

**SERIAL INTERFACE**

**TC/IP**

**OPC UA**

**MQTT**

**docker**

App 1

App 2

App 3

Bins/Libs

Bins/Libs

Bins/Libs

**Docker Engine**

**Operating System**

**Infrastructure**