

**reis**

# ROBOTstar VII

INTUITIV – UNIVERSELL – FLEXIBEL



•PATENTIERT•



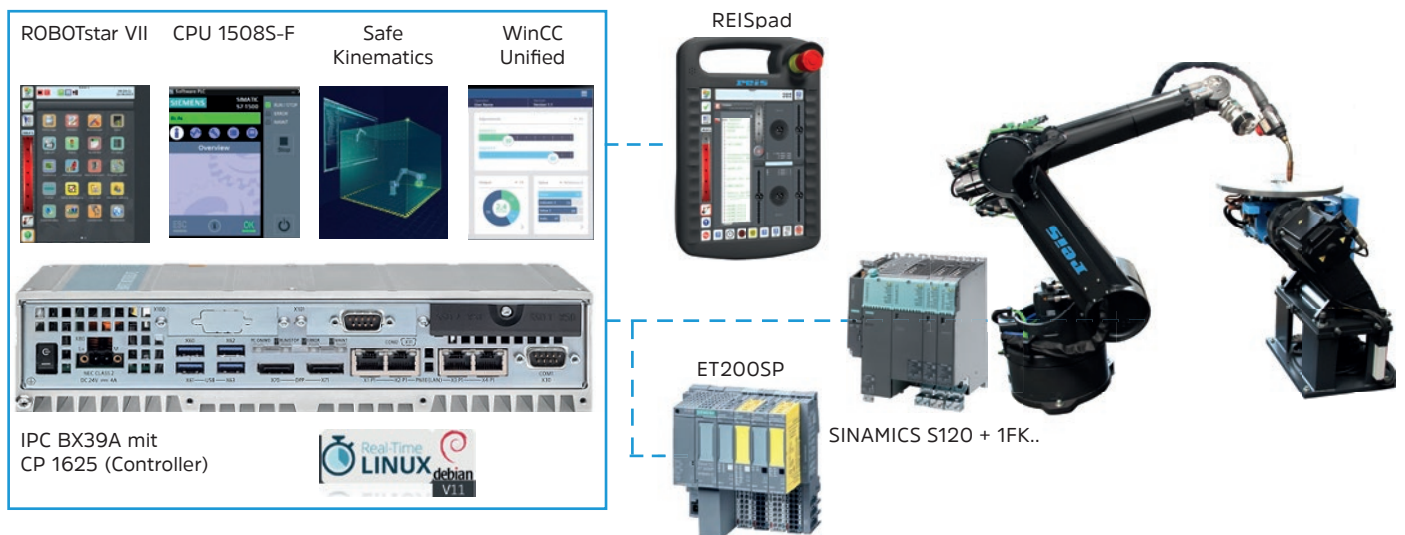
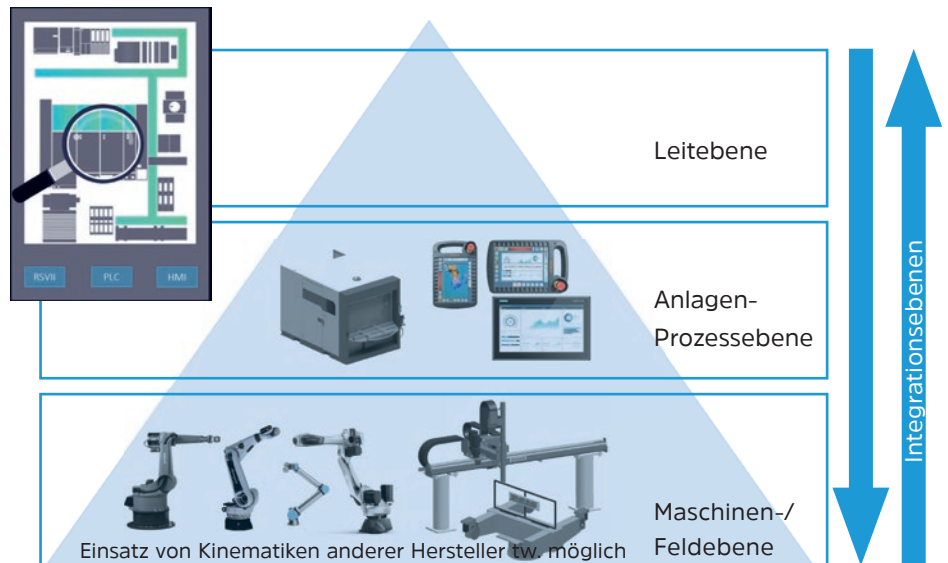
# ARCHITEKTUR

## ROBOTstar VII

Mit der neuen Eigenständigkeit hat sich [Reis Robotics](#) im Oktober 2023 entschieden, die vor 10 Jahren selbst entwickelte und weltbekannte Reis [ROBOTstar VI](#) Steuerung auf LINUX zu portieren. Die neue [ROBOTstar VII](#) steht somit als LINUX App zur Integration in eine gänzlich neue, innovative und zukunftsfähige Steuerungs-Plattform zur Verfügung. In Rekordzeit wurde die Entwicklung gemeinsam mit Siemens vorangetrieben.

### Vorteile

- ein integriertes Steuerungskonzept
- modernste Technik
- weniger Installationsaufwand
- integrierte SPS (Siemens)
- integriertes HMI (Siemens) möglich
- Rückspeisefähig / Gleichstromfähig
- Entwicklung und Service in unseren Händen



### DIE VORTEILE

- Robotersteuerung mit integrierten Siemens SIMATIC Controller
- Neueste und zukunftssichere Technologie
- Möglichkeit, auch Roboter und Automatisierungstechnologien von Drittherstellern anzusteuern und via Profinet IRT anzubinden



# SOFTWARE ARCHITEKTUR

## ROBOTstar VII

Erleben Sie die Revolution der Robotersteuerung mit unserer offenen und flexiblen Softwarearchitektur.

Unsere Lösung bietet eine einzigartige Kombination aus Flexibilität, Effizienz und Innovation, die sich an die Bedürfnisse von Entwicklern und normalen Anwendern gleichermaßen richtet. Profitieren Sie von einer offenen, skalierbaren Plattform, die schnelle und einfache Integration ermöglicht und dabei auf einem zukunftssicheren, innovativen Ansatz basiert.

### Ihre Vorteile auf einen Blick

- **Flexibler, effizienter Betrieb:** Passen Sie die Steuerung Ihrer Roboter präzise an Ihre spezifischen Anforderungen an und optimieren Sie Ihren Betrieb
- **Offene, skalierbare Lösung:** Erweitern und skalieren Sie Ihre Anwendungen ohne Einschränkungen, um mit den wachsenden Anforderungen Schritt zu halten
- **Schnelle, einfache Integration:** Integrieren Sie unsere Software nahtlos in bestehende Systeme und reduzieren Sie den Aufwand für die Implementierung
- **Innovativer, zukunftssicherer Ansatz:** Nutzen Sie modernste Tech-

nologien, um sich auf die Zukunft vorzubereiten und stets einen Schritt voraus zu sein

### Beispielhafte Lösungsansätze

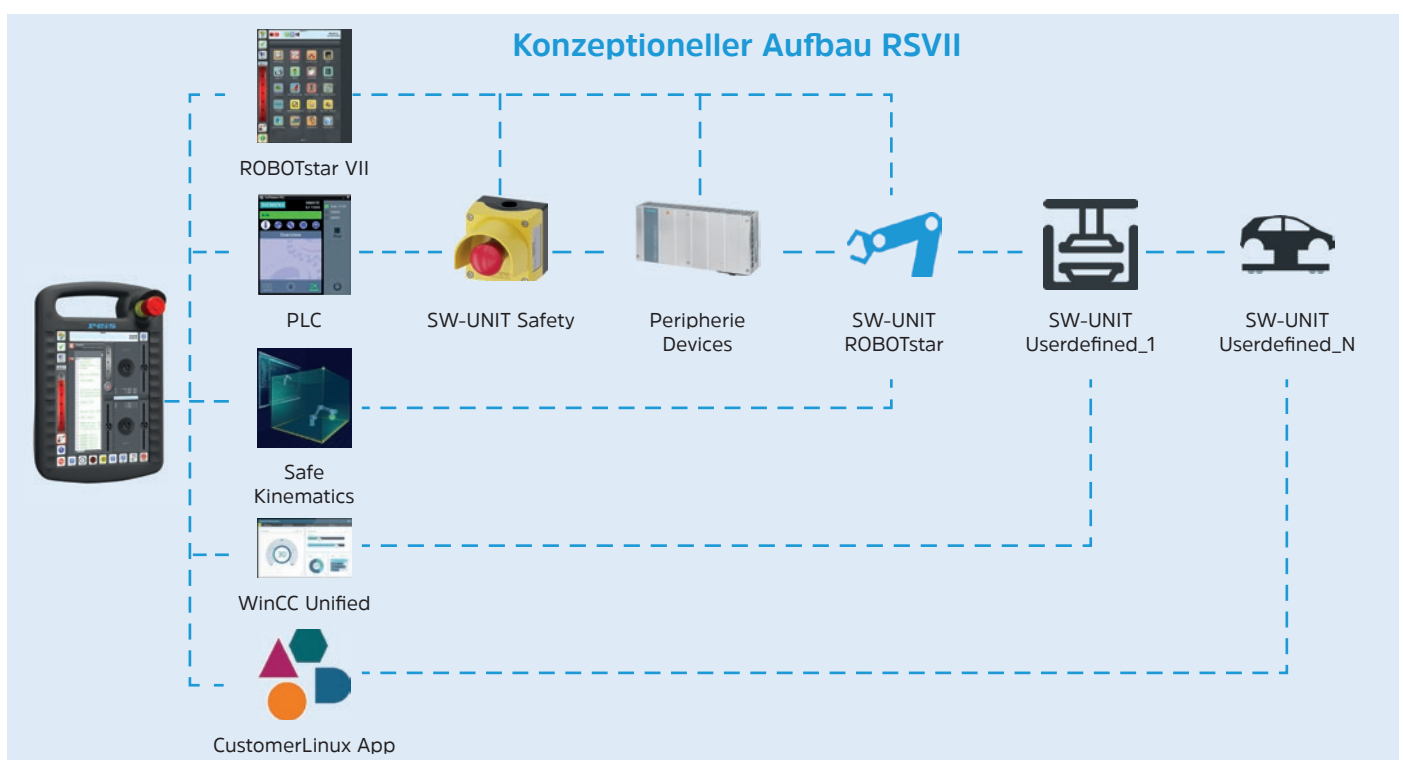
- **IT-OT-Integration:** Verbinden Sie Ihre IT- und OT-Systeme für eine nahtlose Datenübertragung und verbesserte Entscheidungsfindung
- **Produktionsanalytik:** Nutzen Sie detaillierte Analysen, um Ihre Produktionsprozesse zu überwachen und kontinuierlich zu optimieren
- **Effiziente Wartung:** Implementieren Sie präventive Wartungsstrategien, um Ausfallzeiten zu minimieren und die Lebensdauer Ihrer Maschinen zu verlängern

### Flexible Einsatzmöglichkeiten

Unsere Robotersteuerungslösung wird in naher Zukunft als individuell angepasste LINUX-Anwendung oder in der integrierten Siemens Edge-Umgebung einsetzbar sein. Dies bietet Ihnen die Freiheit, die für Ihre spezifischen Anforderungen am besten geeignete Implementierung zu wählen.

Jetzt die Zukunft gestalten - Entdecken Sie, wie unsere offene Softwarearchitektur Ihnen hilft, die Herausforderungen an eine moderne und zukunftsfähige Produktionssteuerung zu meistern.

**Sprechen Sie uns jederzeit an.**



# MAXIMALE FLEXIBILITÄT UND EFFIZIENZ

## mit unserem innovativen Schaltschrankaufbau

Unsere Schaltschranklösung ist speziell für die Bedürfnisse moderner Robotersysteme und Fertigungs-  
linien konzipiert. Mit einem großzügigen und flexiblen Aufbau bietet sie ausreichend Platz für Ihre  
Applikationen und sorgt gleichzeitig für eine kostengünstige und effiziente Implementierung.

### Ihre Vorteile auf einen Blick:

- **Großzügiger Schaltschrankaufbau:** Nutzen Sie den großzügigen Innenraum für die Integration Ihrer Applikationen und zusätzliche Komponenten, ohne Platzmangel
- **Platz für Applikationen:** Unser Schaltschrank bietet genügend Raum für sämtliche Applikationen und ermöglicht eine übersichtliche und ordentliche Anordnung aller Komponenten
- **Eine Schaltschranklösung für Zellen und Anlagen:** Egal, ob für einzelne Roboterzellen oder komplette Anlagen – unsere Schaltschranklösung passt sich Ihren Anforderungen perfekt an
- **Flexibler Schaltschrankaufbau:** Erleben Sie maximale Flexibilität bei der Konfiguration und Erweiterung des Schaltschranks, um Ihre spezifischen Bedürfnisse zu erfüllen
- **Kostengünstiger Ansatz:** Profitieren Sie von einem wirtschaftlichen Design, das nicht nur Ihre Investitionskosten senkt, sondern auch die Betriebskosten minimiert

Technische Daten	
Schnittstellen	RJ 45 Protokolle: Profinet/Profisafe IO- Controller Profinet/Profisafe IO-Device IP-Protokolle OPC UA (in der RSVII und in der Siemens SPS)
Anzahl Achsen	24 Roboterachsen Transformation für Zusatzachsen (z.B. Interpolation von 11 Achsen gleichzeitig und synchronisiert) 128 Servoachsen insgesamt steuerbar über die Hard- und Softwarearchitektur 100 Servoachsen steuerbar wenn 24 Roboterachsen definiert sind
Robotersicherheitsfunktionen	Ja, 12 interpolierende Achsen SLS, SZM, SLO, STO
SPS	Ja S7 1500 Programmiersprachen: AWL, KOP, FUP, SCL, CFC, Graph Integrierte SPS in der RSVII
Visualisierung	Ja ProVis WinCC unified (Optional)
ReisPAD	Ja ProVis WinCC unified (Optional)

## Effizienz und Flexibilität in einer Lösung vereint

Unsere Schaltschranklösung ergänzt perfekt unsere offene und flexible Softwarearchitektur der Robotersteuerung. Zusammen bilden sie ein unschlagbares Team, das Ihnen hilft, Ihre Produktionsprozesse zu optimieren und zukunftssicher zu gestalten.

## Jetzt entdecken und profitieren

Erfahren Sie mehr über unsere innovative Schaltschranklösung und wie sie Ihre Roboterzellen und Anlagen auf das nächste Level bringt.

### Soft- & Hardware

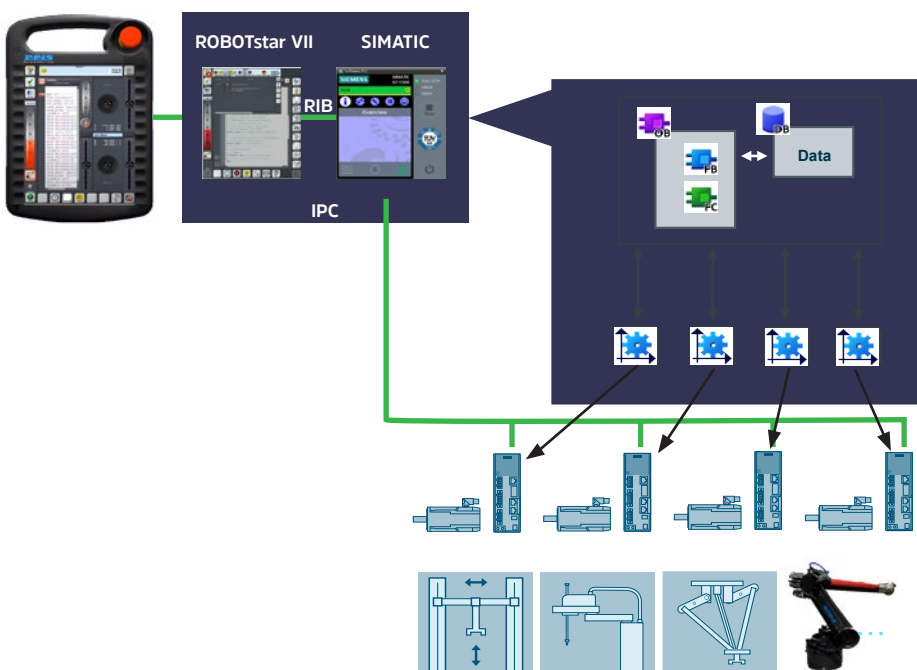
- Siemens IPC
- RSVII
- Siemens SPS

### Siemens-Schaltschrankaus-rüstung

- Regler
- I/O's
- Schaltgeräte
- Switche



## Motion Control Überblick



### ◀ Anwendung

Antriebsfreigabe, Schrittketten, MotionIn...

### ◀ Technologieobjekte

Einzelachs-bewegungen, Mechanikkonfiguration, Diagnose

### ◀ Antriebssystem

Antriebsregelung, Sollwert-generierung...

### ◀ Kinematik

Maschinenspezifisch

# SCHALTSCHRANK

## Aufbau & Möglichkeiten



### Applikationsbereich

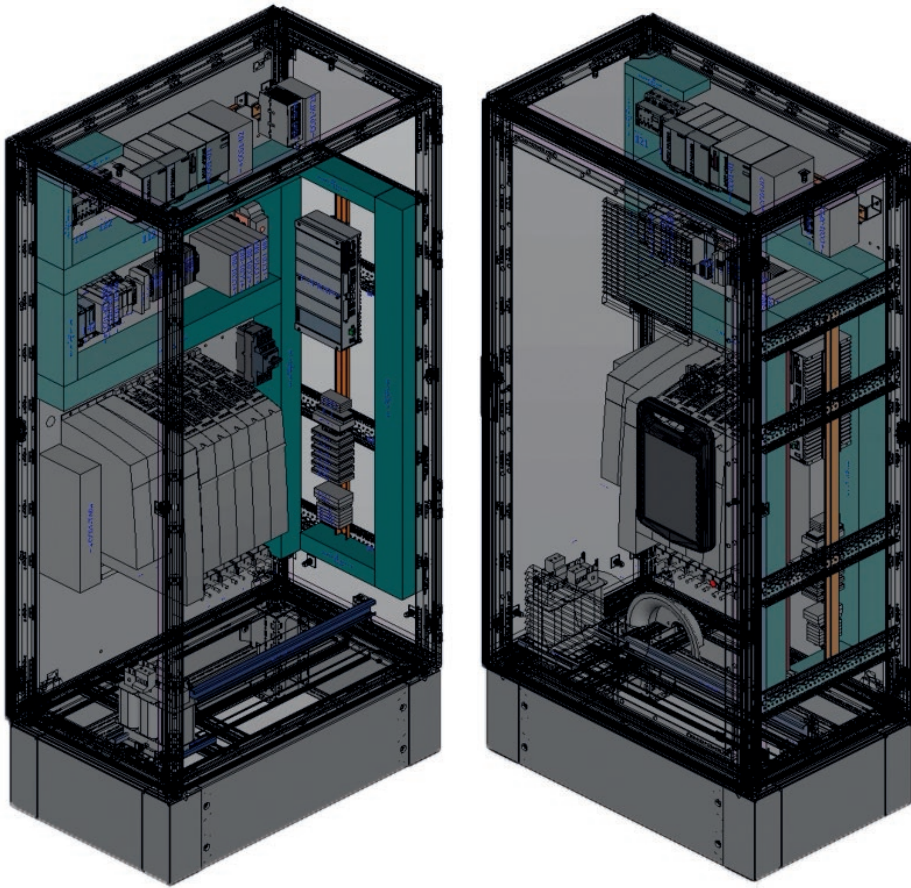
#### Standardschrank klein

Maße: 1400 \* 800 \* 500 mm  
(ohne Sockel)

Achsen: bis zu 12 Achsen

Hauptschalter kann je nach  
Applikation angepasst werden





## Technische Daten

**Elektrischer Anschluss:**  
3AC 380-480V 50/60Hz,  
optional mit Trafo

**Schutzart:** IP54

**Umgebungstemperatur:**  
0-40 °C, optional 0-50 °C

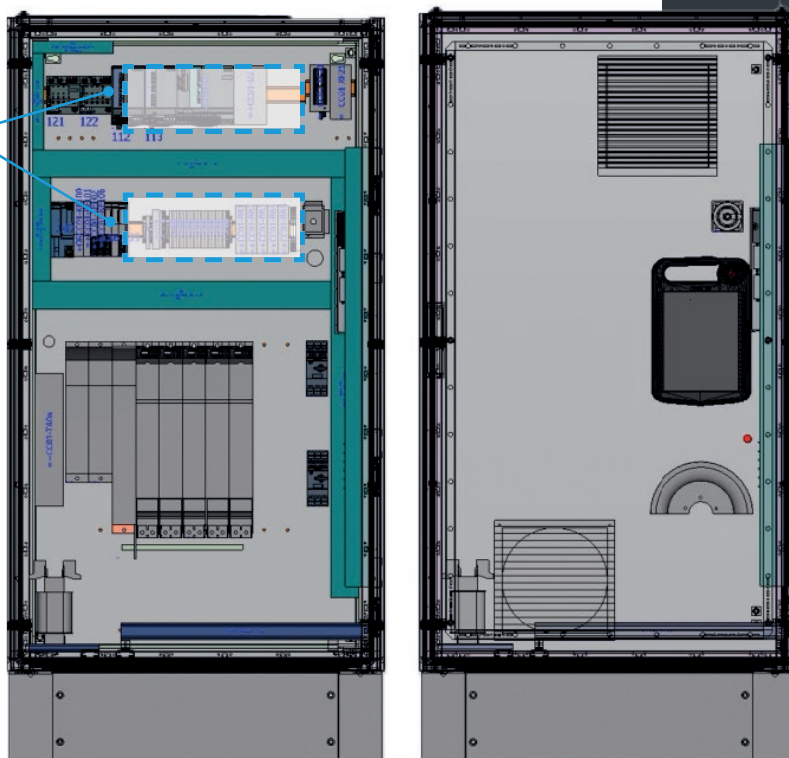
**Relative Luftfeuchtigkeit:**  
max. 85 %

**SIMATIC IPC BX-39A mit 240 GB**  
(SSD M.2) internen Speicher

**Abmessungen:**  
1400 x 800 x 500 mm (H x B x T)

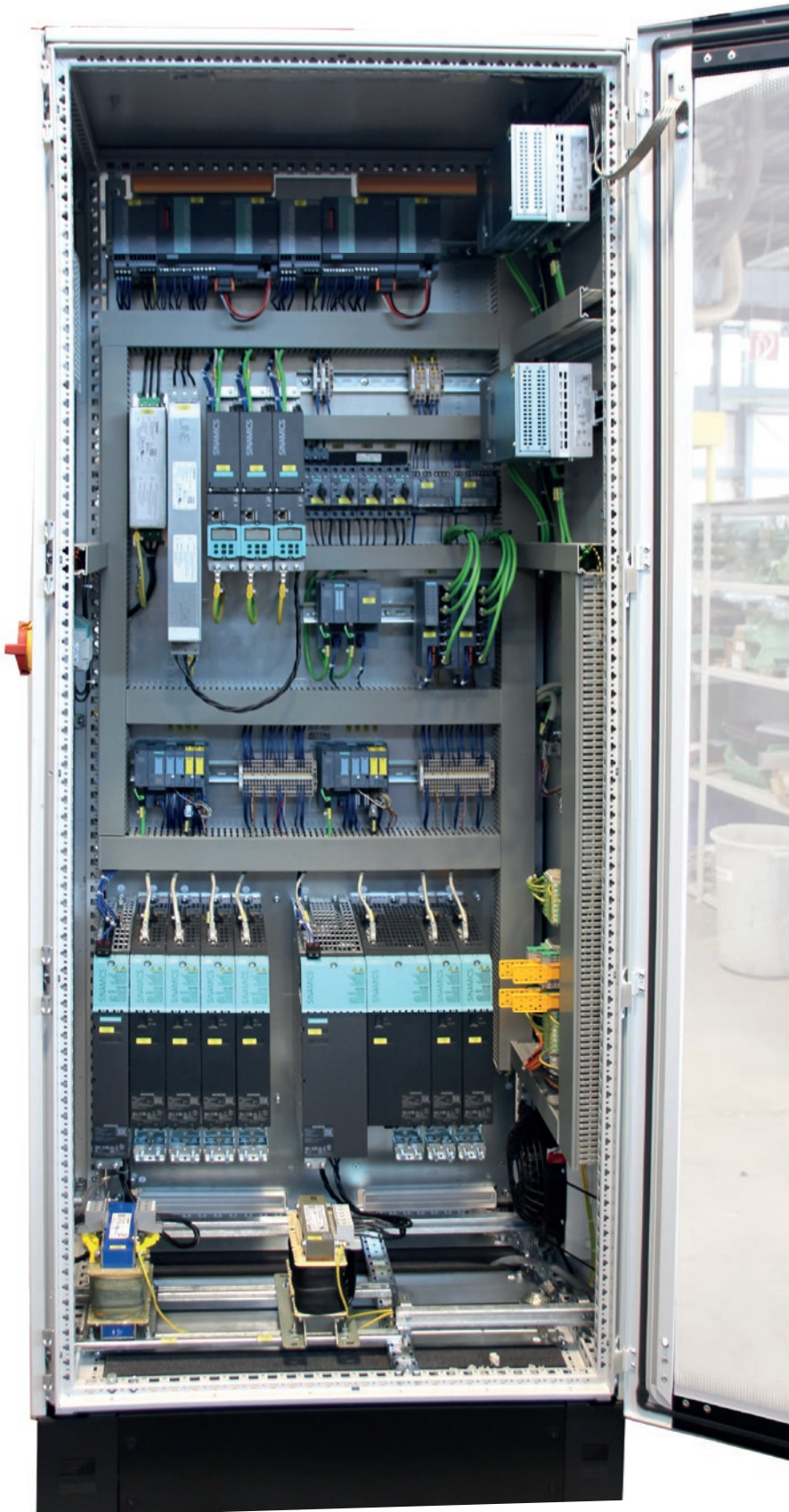
**Gewicht:** 210 kg

Applikationsbereich



# SCHALTSCHRANK

## Aufbau & Möglichkeiten



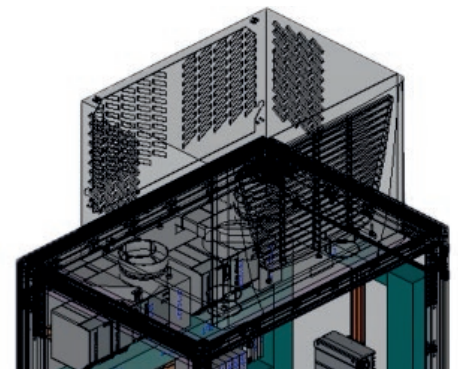
### Applikationsbereich

#### Standardschrank groß

Maße: 2000\*800\*600 mm  
(ohne Sockel)

Achsen: bis zu 24 Achsen oder  
2 Roboter a 6-9 Achsen

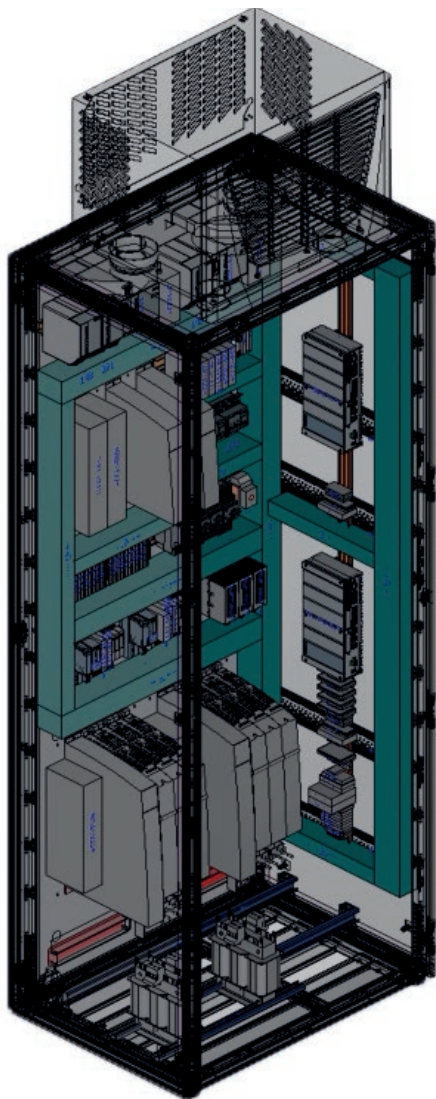
Hauptschalter kann je nach  
Applikation angepasst werden



### Optional

gibt es noch den Anbau eines Dach-  
kühlgerätes für die Erweiterung der  
Umgebungstemperatur als Standard-  
Option.





## Technische Daten

**Elektrischer Anschluss:**  
3AC 380-480V 50/60Hz,  
optional mit Trafo

**Schutzart:** IP54

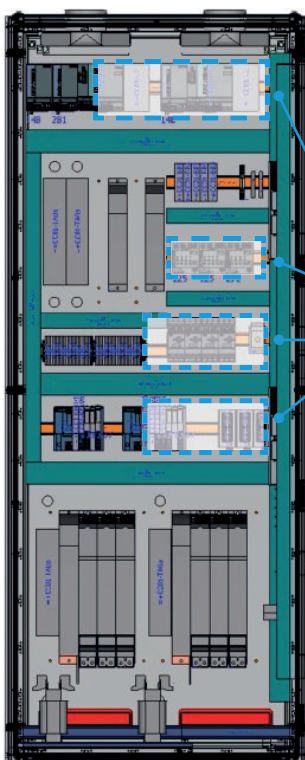
**Umgebungstemperatur:**  
0-40 °C, optional 0-50 °C

**Relative Luftfeuchtigkeit:**  
max. 85 %

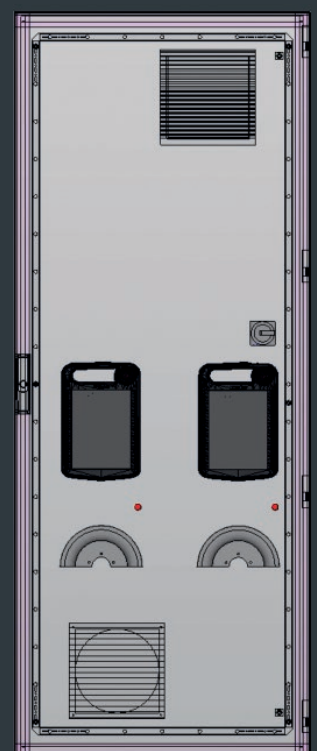
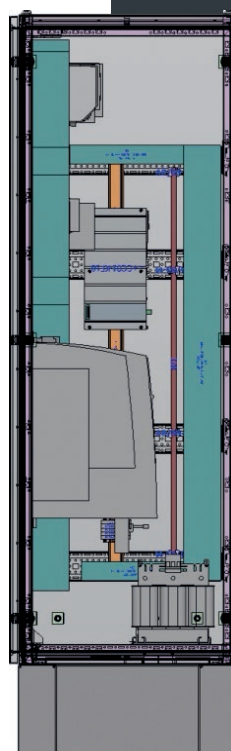
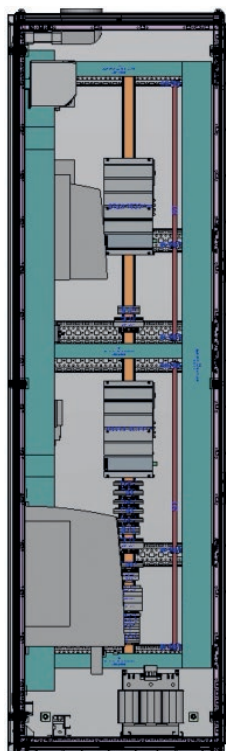
**SIMATIC IPC BX-39A mit 240 GB  
(SSD M.2) internen Speicher**

**Abmessungen:**  
2000 x 800 x 600 mm (H x B x T)

**Gewicht:** 300 kg



Applikationsbereich



# ROBOTstar VII

## ergonomisch, leicht und übersichtlich

Die neue **ROBOTstar VII** basiert auf der seit Jahren bewährten und erfolgreichen Steuerungssoftware und Hardware der **ROBOTstar VI** mit ihrer enormen Funktionsvielfalt. Somit sind nahezu alle Steuerungsfunktionen auch auf der neuen Steuerung verfügbar. Die Anwendungsprogramme der **ROBOTstar V** und **ROBOTstar VI** sind aufwärts kompatibel und können auf der **ROBOTstar VII** weitestgehend weiter verwendet werden.

Die augenfälligste Neuerung ist das neue Programmierhandgerät **reisPAD** mit einer grafischen Bedienoberfläche, die über einen Touchscreen bedient wird.

### Intuitives Bedienkonzept mit Visualisierungsbausteinen

Das **reisPAD** kommuniziert mit der Robotersteuerung über Ethernet und ist somit netzwerkfähig. Mit einem Bediengerät können verschiedene Robotersteuerungen in einem Netzwerkverbund angewählt und betrieben werden.

Für das intuitive Handverfahren des Roboters kann die bewährte Reis 6D-Maus eingesetzt werden. Dazu wird die 6D-Maus mittels einer Magnethalterung an den Roboter geheftet und über ein Kabel mit dem **reisPAD** verbunden. Der Roboter lässt sich so in allen 6 Freiheitsgraden bewegen.

Es ist ein USB-Anschluss vorhanden, an den eine PC-Tastatur, eine PC-Maus oder ein Wechselspeicher für die Datenarchivierung angeschlossen werden kann.

Das neue Bedienkonzept nutzt konsequent den kapazitiven Touchscreen für alle Bedien- und Eingabefunktionen. Die robuste Glasoberfläche ist auch für raue Betriebsumgebungen, z.B. in Gießerei, geeignet.

Die Anzahl der Hardwarekomponenten wurden auf ein absolutes Minimum reduziert. So wurden alle Bedienelemente konsequent in Software realisiert, mit Ausnahme der sicherheitsrelevanten Zustimmungsschalter und Not-Halt. Auch der Schlüsselschalter für die sicherheitsrelevante Betriebsartenwahl ist in Software realisiert. Dies ist möglich, da der in die Robotersteuerung integrierte Safety-Controller die ordnungsgemäße Funktionalität der Software gewährleistet.

### Vorteil der Softwarelösung gegenüber einer Folientastatur:

- durch eine Softwareupdate können neue Oberflächen erzeugt werden
- dadurch ist das System zukunfts-sicher

#### DIE VORTEILE

- **reisPAD** verfügt über ein Linux Betriebssystem
- **ProVis** (Programmierung / Visualisierung), grafische Benutzeroberfläche
- **ProVis** – Java-Anwendung – **reisPAD**-HMI betriebssystemunabhängig
- keine Vorinstallation der Systemsoftware Beim Verbindungsaufbau **reisPAD/ROBOTstar VII** wird der **ProVis**-Download aus der Steuerung eingeleitet
- tastenloses Design ermöglicht auf einfache Weise Oberflächenerweiterungen über Softwareupdates





C-Rahmen mit omnipräsenten Systemfunktionen: immer an der gleichen Stelle verfügbar

Möglichkeit auch im App-Bereich Softkeys **blind zu ertasten** über:

- 1 Ecken
- 2 Fingermulden
- 3 Noppen
- 4 Kante

App(lication)-Bereich:  
Inhalt wechselt kontextabhängig



Erfahren Sie mehr über  
unser [ROBOTstar VII-Steuerung](#)

# BEDIENPHILOSOPHIE

## und Visualisierungslösung

reisPAD ist mit einem Linux-Betriebssystem und ProVis (Programmierung / Visualisierung) ausgestattet. Die grafische Benutzeroberfläche ProVis ist eine Java-Anwendung. Das reisPAD-HMI ist unabhängig vom Betriebssystem. Kein Vorinstallieren der Software notwendig. Beim Aufbau der Verbindung zwischen dem reisPAD und der ROBOTstar VII startet der ProVis-Download vom Steuerelement. Tastenloses Design erlaubt es auf unkomplizierte Art und Weise, Software-Updates zur Oberflächenerweiterung anzuwenden.

Das Look and Feel und die Bedienung sind an SmartPhones angelehnt und kann somit als allgemein bekannt vorausgesetzt werden.

Der durch den Entfall der Folientastatur erreichte Platzgewinn geht zu Gunsten eines großen 10,4" Farbdisplays, auf dem nun reichlich Platz für die Anzeige von Informationen und Bedienelementen vorhanden ist. Die Bedienelemente werden als große, treffsichere Softkeys auf der Touchoberfläche angeboten. Die Anordnung der Bedienelemente kann für Rechts- und Linksbedienung umgeschaltet werden.

Die Bedienelemente sind so auf dem Display platziert, dass die wichtigsten Funktionen über haptische Marken am Gehäuserand (Ecken, Kanten, Fingermulden, Noppen) ertastet werden können. Dies kompensiert die prinzipiell fehlende Haptik eines Touchscreens.

Über diese haptischen Marker ist sogar eine Blindbedienung der wichtigsten Funktionen möglich, was in Tests unter Beweis gestellt wurde: komplette Bewegungssequenzen können, ohne auf das reisPAD zu schauen, blind erstellt, und in ein Programm eingefügt werden.

Als weitere Unterstützung bei der Touchscreeneingabe dient die Audioausgabe über den eingebauten Lautsprecher. Darüber kann ein akustisches Feedback gegeben werden, z.B. Tastenklick, Warnhinweise, oder Fehlermeldungen.

Die Bedienoberfläche ist internationalisierbar und wird in verschiedenen Landessprachen angeboten. Die Programmiersprache wird dabei in Form von verständlichen Klartext-Befehlen in der Landessprache dargestellt, z.B. „Greifer\_öffnen“.

Die Bedienerführung wird unterstützt durch Verwendung von Icons, die jeweils mit einem beschreibenden Kurztext unterlegt sind. Dies fördert die leichte Erlernbarkeit. Ein Anfänger orientiert sich zunächst an den Texten bis er sich Farbe und Form der Icons eingeprägt hat. Der versierte Anwender orientiert sich dann nur noch an der visuellen Form und Farbe der Icons. Diese Technik wurde angewendet bei APPs, Softkeys, Listen, etc.

Das Oberflächendesign verwendet fotorealistische Bedienelemente, was den Wiedererkennungswert erhöht und zudem zeitlos ist.

### Die Oberflächenstruktur hat einen strukturierten Aufbau:

Ein fest stehender Rahmen (C-Rahmen) enthält alle Bedienelemente, die ständig verfügbar sein müssen. Der Raum innerhalb des Rahmens und eine Seite des Displays sind für die unterschiedlichen Applikationen reserviert. Dadurch ist die Oberfläche sehr leicht und intuitiv erlernbar.

Die Bedienhierarchie ist möglichst flach gehalten: maximale Leveltiefe

von 1-2, sodass der Bediener immer den Kontext behält.

Alle Menüs wurden eliminiert und durch Listen mit Schnellzugriffsmöglichkeit ersetzt.

Alle für einen Arbeitsvorgang vorrangig erforderlichen Bedienelemente sind auf der Bedienoberfläche direkt verfügbar. Lediglich einige weniger oft benötigte Funktionen können aus Listen ausgewählt werden.

Beispiel: Roboterprogramm editieren und Roboter-Handverfahren und Ansteuerung von Peripherie (z.B. Greifer) ist gleichzeitig möglich.

Die Bedienoberfläche ist für Zweihandbedienung optimiert. Die Bedienelemente sind so angeordnet, dass sie mit beiden Daumen optimal erreicht und bedient werden können. Dadurch ist doppelt so schnelles Arbeiten möglich. Der Touchscreen ist multitouchfähig, sodass dort wo es unabdingbar ist, auch mit mehreren Fingern gleichzeitig bedient werden kann.

Eine Gestensteuerung, wie sie von SmartPhones her bekannt ist, erlaubt die Navigation im Programm, das Scrollen in Listen/Tabellen und das Bewegen des Roboters.

Der verfügbare Displayplatz ist optimal ausgenutzt, es sind keine ungenutzten Bereiche vorhanden. Das Bedienkonzept ist so ausgelegt, dass durch Minimierung der erforderlichen Bedienaktionen, eine Programmerstellung in kürzestmöglicher Zeit möglich ist.

Virtualisierung der Bedienelemente



Intuitive Bedienung smartphone



reisPAD



ROBOTstar V



ROBOTstar VII

So ist ein sehr rationelles Arbeiten möglich, bis hin zur Blindbedienung der wichtigsten Funktionen.

Als Beispiel sei hier angeführt, die omnipräsente Verfügbarkeit aller Funktionen die für das Erstellen von Programmen erforderlich sind, ohne dass dabei der Programmkontext verlassen werden muss:

- Editor zum Editieren und Navigieren im Programmtext
- Bedienelemente zum Bewegen des Roboters in allen 6 Freiheitsgraden
- Elemente zum Einlernen von Positionen
- Onlineansteuerung der Peripherie
- Start/Stop für das Testen der erstellten Programme
- Override-Regler für die Vorgabe der Roboterverfahrensgeschwindigkeit

Bei der Eingabe von Programmbefehlen wird der Programmierer mit Hilfe einer kontextsensitiven Syntaxvervollständigung geführt.

Sollte der Anwender dennoch Hilfe benötigen, so kann er am reisPAD eine Onlinedokumentation abrufen, die in jeder Steuerung hinterlegt ist. Bei Fehlermeldungen kann eine Hilfefunktion aufgerufen werden, die Aufschluss über die Fehlerursache und mögliche Abhilfemaßnahmen gibt. Die Hilfe-Informationen können vom Anwender selbst verändert und erweitert werden, so dass ein individuelles Expertensystem aufgebaut werden kann.

Die im Editor dargestellte Programmansicht kann per Gestiksteuerung größer oder kleiner gezoomt werden. Eine Funktion, wie sie bei Smartphones ebenfalls üblich ist.

Der Anwender kann eigene Makros erstellen und in Form von APPs in einer Liste hinterlegen. Diese APPs können jederzeit aufgerufen werden.

### RobOffice, die Steuerung fürs Büro:

Die Steuerungssoftware gibt es als virtuelle Robotersteuerung RobOffice, die unter Windows lauffähig ist. Die touchbasierte Bedienoberfläche des reisPAD ist als Teilfunktion in die grafische Bedienoberfläche ProVis eingebettet und somit auch in RobOffice verfügbar.

# HIGHLIGHTS

## der touchbasierten Bedienoberfläche

Bei der Entwicklung der touchbasierten Bedienoberfläche sind einige Neuheiten entstanden, die im Bereich der Industrierobotik bislang einzigartig sind und nachfolgend beschrieben werden:

### 1. balancedUse

Das Gehäuse ist symmetrisch aufgebaut, mit Ausnahme des Not-Aus-Tasters. Es enthält auf der Vorderseite links und rechts des Displays einen Displayrahmen mit haptischen Marken über deren Hilfe ausgewählte Positionen am Touchscreen ertastet werden können. Dadurch eine Blindbedienung des Touchscreens möglich ist (Bild 1a).



1a. Symmetrischer Gehäuseaufbau vorne



1b. Symmetrischer Gehäuseaufbau hinten

Auf der Rückseite befinden sich zwei symmetrisch platzierte Griffleisten, an denen das Gerät mit einer oder mit beiden Händen sicher gehalten werden kann (Bild 1b). In jeder Halteleiste ist jeweils ein Zustimmungsschalter

integriert, von denen wahlweise einer für die Verfahrensfreigabe des Roboters betätigt werden muss.

Durch diese symmetrische Anordnung wird einer Handermüdung vorgebeugt, da die Zustimmungsschalter wechselweise mit der linken oder rechten Hand betätigt werden können. Bei Handermüdung kann die jeweils andere Hand die Zustimmung übernehmen, ohne dass dadurch die Roboterbewegung unterbrochen wird.

### 2. blindToUse

Die wichtigsten virtuellen Bedienelemente wurden auf dem Touch-Display so platziert, dass sie blind bedienbar sind. Dabei dient der abgesetzte und besonders ausgeprägte Gehäuserahmen zur taktilen Orientierung auf dem Touchscreen (Bild 2).



2. Gehäuserahmen mit haptischen Marken am Gehäuserand: Kanten, Ecken, Fingermulden und Noppen

Als haptische Marken am Gehäuserand dienen Kanten, Ecken, Fingermulden und hervorstehende Noppen, die so ausgeformt sind, dass sie mit den Fingern zuverlässig ertastet werden können. Dadurch wird zum Einen die prinzipiell fehlende Haptik eines

Touchscreens kompensiert, zum Anderen kann der Bediener den visuellen Fokus auf dem Roboter und den Prozess richten ohne auf das Bediengerät sehen zu müssen, wodurch die Bediensicherheit erhöht wird.

Es gibt 4 verschiedene haptische Markertypen, die jeweils symmetrisch am linken und rechten Displayrahmen angeordnet sind:

**a) Kanten:** entlang dieser Kanten kann der Finger eine Verschiebewegung ausführen über die eine am Displayrand platziertes virtuelles Schieber-Element verstellt wird

**b) Ecken:** Die vier Ecken des Displayrahmens definieren eine eindeutige, exakte Position auf dem Touchdisplay. Auf diese ausgewählten Positionen können wichtige blind zu bedienende Softkeys platziert werden

**c) Noppen:** Die aus dem Gehäuserand hervorstehenden Noppen dienen für eine grobe Orientierung auf dem Displayrand. Sie sind am oberen und unteren Ende des Displayrahmens platziert und markieren so die obere und untere Displaybegrenzung. In der Mitte zwischen oberer und unterer Begrenzung ist eine weitere Noppen platziert, der die Mitte des Displays markiert. Somit kann die obere und untere Displayhälfte ertastet werden

**d) Fingermulden:** Die aus dem Gehäuserand eingelassenen Fingermulden dienen für eine feinere haptische Orientierung auf dem Displayrand, z.B. für das Erfühlen von Softkeypositionen, wenn diese unmittelbar neben dem Fingermulden angeordnet sind. Jeder Fingermulde kann ein Softkey eindeutig zugeordnet werden. Die Fingermulden sind halboffen ausgeführt

und in Richtung Display geöffnet, so dass ein Finger, wie in einer Rinne geführt, auf das Display gleiten und dort eine Softkeyfunktion auslösen kann

### 3. TouchControlled-HMI

Die Anzahl der Hardwarekomponenten wurden auf ein absolutes Minimum reduziert. Alle Bedienfunktionen wurden konsequent in Touch-Software realisiert, mit Ausnahme der energieunterbrechenden, sicherheitsrelevanten Schalter „Not-Aus“ und „Zustimmung“. Es sind keine weiteren elektrischen Komponenten wie Folientasten, Schalter oder Signallampen mehr erforderlich. Dadurch ist das System wartungsarm.

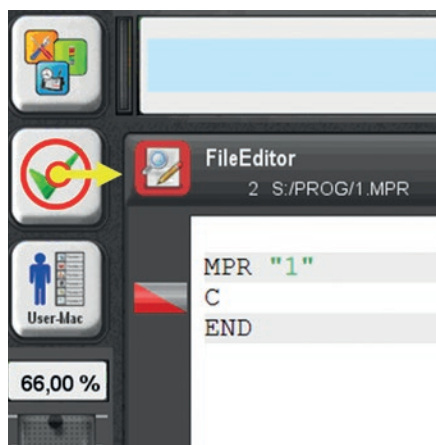
Der erreichte Platzgewinn geht zu Gunsten eines großen komfortablen Touchdisplays. Die auf dem Touchdisplay dargestellten virtuellen Bedienelemente und Anzeigen sind für die Industrieanwendung ausgelegt und sind kontrastreich und groß dargestellt, sodass eine zuverlässige Bedienung möglich ist (Bild 3).



3. touchbasierte Bedienoberfläche mit großen Bedienelementen

### 4. reliableTouch

Die Auslösung einer Touchfunktion erfordert eine manuelle Aktion des Bedieners auf dem Touchscreen. Um ein versehentliches Auslösen von Bedienelementen durch unbeabsichtigte Berührung zu verhindern, wird eine Funktion erst ausgelöst, wenn nach der Berührung des Touchdisplays eine spezielle „Kleingestik“ ausgeführt wird, z.B. Verziehen des Fingers in eine definierte Richtung (Bild 4).



4. Auslösen einer Funktion erfolgt nach Berührung und Ziehen des Fingers über eine Radiusgrenze hinweg

Die Intensität der erforderlichen Gestik kann stufenlos eingestellt werden: Sie reicht von einer einfachen Fingerberührung (allgemein übliche Bedienung von Touchscreens) bis hin zu einer speziellen Kleingeste. Durch die spezielle Ausprägung der Fingermulden im Gehäuserand kann der Finger (in Fortsetzung der Fingermulden) auf das Touchdisplay gleiten und dabei eine Funktion auslösen.

Merkt der Bediener, dass er eine ungewollte Funktionsauslösung eingeleitet hat, kann er die Funktionsauslösung durch Zurückziehen des Fingers in die Ursprungsposition unterdrücken.

### 5. smartTouch

Bedienelemente, die in den Ecken des Gehäuserandes platziert sind, werden mit einer besonderen Gestensteuerung verknüpft. Sie können jeweils entlang der Gehäusekanten L-förmig in zwei Richtungen, verschoben werden (Bild 5).



5. Funktionsauswahl durch L-förmige Geste entlang zweier Gehäusekanten

Jeder Bewegungsrichtung kann eine wählbare Funktion zugeordnet werden. So ist es z.B. möglich, beim Bewegen auf einem Schenkel eine Funktion A und beim Bewegen auf dem anderen Schenkel eine Funktion B zu aktivieren. Der Grad der Auslenkung wird ausgewertet, wobei es zwei Auswertungsmöglichkeiten gibt:

**a) Der Grad der Auslenkung** wird sofort als analoger Parameter (z.B. Geschwindigkeitsvorgabe) an die Funktion übergeben. Wird der Finger in ausgelenkter Stellung losgelassen, springt der Analogwert sofort auf Null. Wird der Finger gleitend wieder in die Ausgangsposition geführt, stellt sich der Parameter analog der Auslenkung wieder auf Null zurück. Diese Funktion kann verwendet werden um z.B. ein Bewegungsprogramm in positive oder negative Richtung zu starten und dabei jeweils analog in der Geschwindigkeit variieren

# HIGHLIGHTS

## der touchbasierten Bedienoberfläche

**b) Bei Überschreitung eines definierbaren Schwellwertes** wird eine schaltende Funktion ausgelöst. Die Aktivierung der Funktion erfolgt allerdings erst dann, wenn der Finger in ausgelenkter Stellung den Touchscreen verlässt. Wird der Finger jedoch ohne loszulassen auf dem Schenkel wieder in die Nulllage zurückgeführt, wird die Funktionsauslösung unterbunden

### 6. TouchMotion

Die Möglichkeit nach dem Berühren des Touchdisplays den Finger mehr oder weniger zu „ziehen“, wird genutzt um eine analoge Verfahrensvorgabe für den Roboter zu erzeugen. So kann der Roboter in 6 Freiheitsgraden (z.B. X, Y, Z, und Werkzeugorientierung A, B, C) feinfühlig gesteuert werden.

Mit der Auslenkung des Fingers ist es möglich eine **Positionsvorgabe** an den Roboter zu geben, ähnlich der Cursorsteuerung per Touchpad bei einem Notebook. Der Roboter lässt sich dabei gleichzeitig in zwei Koordinatenrichtungen, z.B. X und Y verfahren.

In einem weiteren Modus wird durch die Fingerauslenkung eine **Geschwindigkeitsvorgabe** für den Roboter erzeugt: je mehr der Finger ausgelenkt ist, umso schneller fährt der Roboter.

Nach dem Berühren der ausgewählten Verfahrfläche kann der Finger anschließend auch über die Feldbegrenzung über das gesamte Display gezogen werden und so Fahrvorgaben erzeugen. Nach dem Loslassen bleibt der Roboter sofort stehen. Danach muss für eine erneute Verfahrvorgabe das gewünschte Feld wieder getroffen werden. Die Sensibilität der Reaktion auf eine Fingerbewegung

kann über einen Regler, z.B. den Override, sowohl für Positionsvorgabe als auch für Geschwindigkeitsvorgabe stufenlos eingestellt werden. Die sensitive Fläche für das 2D-Verfahren ist in der Nähe des Displayrandes platziert, so dass sie mit einem deutlichen Abstand zum Displayrand mit dem Finger (z.B. mit dem Daumen) noch gut erreicht werden kann (*Bild 6*).



6. 6D Roboterverfahren mittels haptisch ertastbarer Bedienelemente

Um auch in eine dritte Koordinatenrichtung verfahren zu können, wird ein fingerbreites Feld direkt seitlich am Displayrand platziert, so dass dieses Feld mit dem Finger ertastet werden kann. Dieses Feld erzeugt eine eindimensionale Verfahrvorgabe, z.B. in Z-Richtung.

Durch die spezielle Anordnung kann der Bediener die zwei Verfahrfelder deutlich unterscheiden und blind erreichen: Das Feld direkt am Displayrand (Finger hat spürbaren Kontakt mit dem Gehäuserand) aktiviert die Verfahrvorgabe für die dritte Dimension. Das Feld, ca. einen Fingerbreit neben dem Displayrand, aktiviert die

gleichzeitige Verfahrvorgabe in zwei Dimensionen.

Ein Industrieroboter besitzt 6 Freiheitsgrade. Für die Einstellung der Werkzeugorientierung mit den drei Winkeln (A, B, C) wird das gleiche Verfahren wie oben beschrieben verwendet. Dazu wird das Display in zwei Zonen aufgeteilt. In einer (z.B. der oberen) Zone befinden sich die Felder für die Dimensionen 1-3 (z.B. X, Y, Z). In der anderen (z. B. der unteren) Zone befinden sich die Verfahrfelder für die Dimensionen 4-6 (z.B. A, B, C). Über haptische Marke können die beiden Zonen blind unterschieden werden.

Mit einem Multitouch-Display kann der Roboter so in allen 6 Freiheitsgraden simultan verfahren werden. Mit einem Singletouch-Display können die Verfahrfunktionen nur sequentiell verwendet werden.

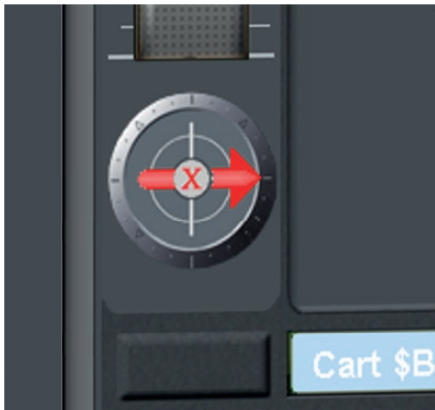
### 7. FrameCalibration

Idealerweise ist beim Verfahren des Roboters mittels TouchMotion-Funktion kollinear zum Koordinatensystem des Roboters ausgerichtet. In diesem Fall stimmt die Roboterbewegung mit der Fingerbewegung optimal überein.

Dreht sich jedoch der Bediener mit dem Bediengerät zur Seite weg, so ist diese Übereinstimmung nicht mehr gegeben. Die Roboterbewegungsrichtung stimmt dann nicht mehr mit der Fingerbewegungsrichtung überein.

Das Koordinatensystem des Displays muss in diesem Fall wieder zum Roboterkoordinatensystem rekaliбриert werden. Dazu gibt es auf dem Touchdisplay ein spezielles virtuelles Bedienelement mit einem Zeiger (*Bild 7*).





7. Kompass für schnelle Rekalibrierung per Geste

Dieses Bedienelement muss zunächst mit einem Finger angetippt werden und anschließend muss der Finger in die ausgewählte Richtung des Roboterkoordinatensystems, z.B. X-Richtung, gezogen werden. Nach dem Abheben des Fingers wird die Vektorrichtung zwischen dem ersten Berührungspunkt und dem Loslasspunkt berechnet. Mit Hilfe dieses Vektors und der ausgewählten Roboterkoordinatenachse wird eine Rotationsmatrix berechnet, über die fortan alle Fingerbewegungen transformiert werden, bevor sie als Verfahrensvorgabe an den Roboter gegeben werden. Die Rekalibrierung erfolgt somit schnellstmöglich mit einer einzigen Geste. Nach der Rekalibrierung sind beide Koordinatensysteme wieder kollinear zueinander eingestellt, quasi eingenordet. Zur besseren Überprüfbarkeit wird die Richtung des kalibrierten Touchkoordinatensystems grafisch auf dem Touchdisplay dargestellt.

Diese Kalibrierungsmethode kann auch für beliebige andere Koordinatensysteme (z.B. Userframes) verwendet werden.

## 8. SoftOverride

Die bewährte (früher als Handrad ausgeführte) Override-Funktion, wird nun über einen Slider am Touchscreen ausgeführt.

Dieser Slider ist auf dem Display am Gehäuserand platziert (Bild 8). Dadurch kann seine Position ertastet und durch Verschiebung des Fingers entlang des Displayrandes verstellt werden. Die blinde Verstellung kann durch haptische Marken am Displayrand zusätzlich unterstützt werden. Der Override kann bei einer Verschiebung zwischen zwei haptischen Marken um einen festen Betrag (z.B. 20%) verstellt werden.

Diese Funktion kann auch verwendet werden um andere analoge Größen z.B. Prozessparameter in Blindbedienung einzustellen.



8. SoftOverride mit ertastbaren Rastmarken (Fingermulden) am Gehäuserand

## 9. soft-ModeSelector

Der bislang übliche Schlüsselschalter für die Anwahl der Roboterbetriebsarten „Einrichten“, „Automatik“, Auto-

matik-Test“ wird durch eine Softwarefunktion ersetzt. Die Besonderheit liegt in der Datenverarbeitung in sicherer Technik. Ein Touchscreen ist prinzipiell ein einkanaliges und somit unsicheres Gerät. Unter Zuhilfenahme eines in die Robotersteuerung integrierten SafetyControllers wird jedoch eine sichere Funktionalität der Software gewährleistet.

### Funktionsbeschreibung

Auf dem Display werden von der Bedienoberfläche verschiedene Betriebsart-Optionen in Form von Softkeys zur Auswahl angeboten (Bild 9a).



9a. Anwahl einer Betriebsart

Über die Berührung eines dieser Softkeys wählt der Bediener eine neue Betriebsart „X“ an. Von der Bedienoberflächensoftware wird die neu ausgewählte Betriebsart als Kommando „Anforderung neue Betriebsart-X“ an den SafetyController gesendet. Der SafetyController entnimmt aus seinem Speicher ein dieser Betriebsart entsprechendes Icon und platziert es an eine zufällige Stelle in einem größeren Bild. Die Position des Icons im Bild ist nur dem SafetyController bekannt.

Dieses Bild wird nun als Bitmap zur Bedienoberfläche gesendet und dort zur Anzeige gebracht (Bild 9b).

# HIGHLIGHTS

## der touchbasierten Bedienoberfläche



9b. Bestätigung der Betriebsart

Durch einen Fingertip auf das dargestellte Icon muss der Bediener die vom SafetyController erkannte Betriebsart bestätigen. Die Berührungsposition auf dem Touchscreen wird in Form von Touchkoordinaten an den SafetyController zurückgesendet. Dieser vergleicht die Berührungsposition mit der Iconposition im Bild. Sind beide (innerhalb einer definierten Toleranz) gleich, wird der eingeleitete Betriebsartenwechsel ausgeführt. Andernfalls wird der Betriebsartenwechsel verworfen und die vorherige Betriebsart bleibt erhalten.

Durch dieses Verfahren entsteht ein **sicherer Wirkkreis zwischen Bediener und SafetyController**

- Bediener wählt eine Betriebsart an,
- SafetyController zeigt die erkannte Betriebsart an
- Bediener bestätigt dem SafetyController die Richtigkeit der angezeigten Betriebsart
- SafetyController stellt die neue Betriebsart ein.

### Anmerkungen:

- alle Icons sind im SafetyController in sicherer Technik gespeichert
- optional kann eine Anforderung zum Betriebsartenwechsel, auch über einen Hardware-Schlüsselschalter kommen
- das Einstecken/Abziehen des Schlüssels in den Betriebsartenwahlschalter wird durch ein Login/Logout-Verfahren mittels PIN nachgebildet

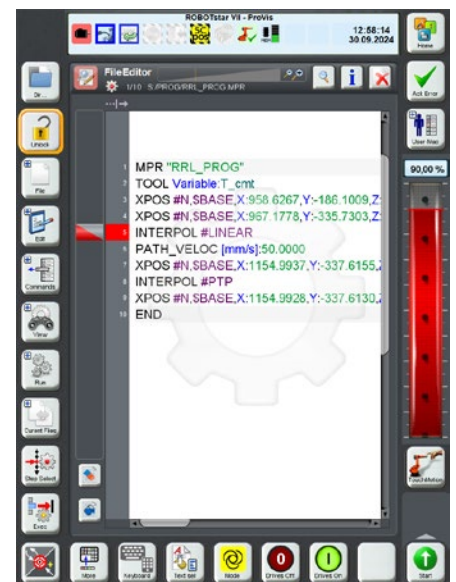
## 10. LeftRightMode

Durch das konsequente tastenlose Design der Bedienoberfläche und durch den symmetrischen Aufbau des Gehäuses mit seinen symmetrisch angeordneten haptischen Marken, kann der Anwender die Bedienoberfläche über eine einfache Setup-Funktion von Rechtshänder-Bedienung auf Linkshänder-Bedienung umschalten.

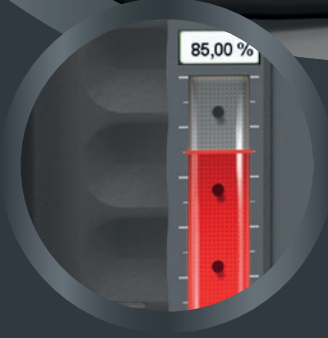
Dabei werden alle Positionen der Softkeys an der vertikalen Achse des Bediengerätes gespiegelt (Bild 10a und 10b). Diese Variabilität bleibt konventionellen Geräten mit mechanisch aufgebauten Tasten vorenthalten.



10a. Rechtshänder-Bedienung



10b. Linkshänder-Bedienung



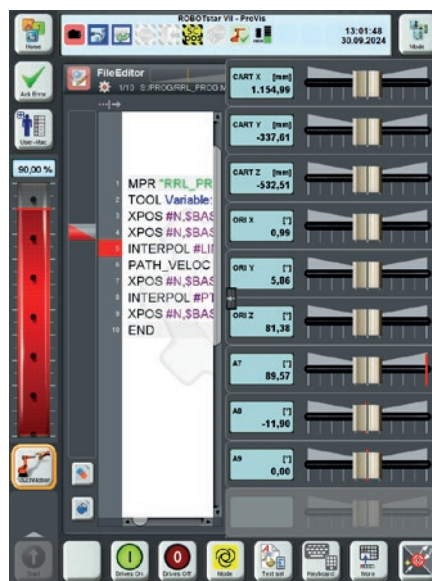
# ProVis

## Programmierung und Visualisierung

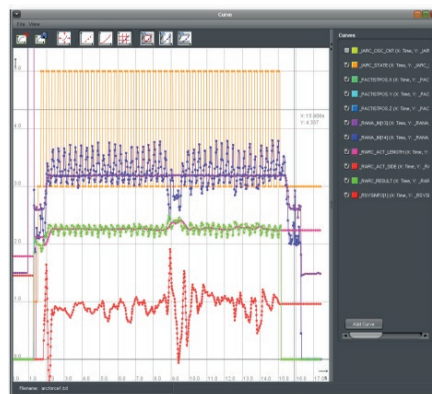
ProVis ist Bestandteil der ROBOTstar-Systemsoftware und kann auf unterschiedlichen Geräten ausgeführt werden, darunter Steuerung, reisPAD, Tablet und PC. Die Nutzung ist auch weltweit über das Internet möglich.

### ProVis vereint eine komplette Toolsammlung

- ist integrierter Bestandteil der ROBOTstar VII System Software
- reisPAD HMI (Apps)
- CAD-Daten-import/export (VRML,DXF,STL)
- Kollisions- und Erreichbarkeitsprüfung
- 3D-Online/Offline-Programmierung
- Online Teach-In
- automatische Bahngenerierung
- SCADA Konfigurator
- Diagnosefunktion/Oszilloskop ...
- WebBrowser
- automatische Datensicherungs-Funktion
- Virtual Reality/Augmented Reality
- **Keine zusätzlichen Lizenzkosten bei Kauf eines Roboters mit ROBOTstar VII**



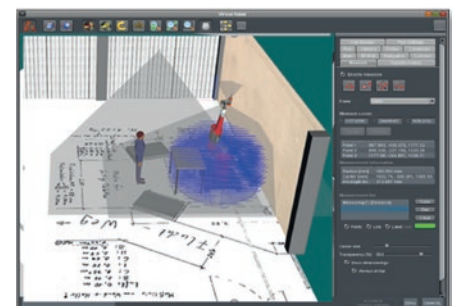
reisPAD HMI



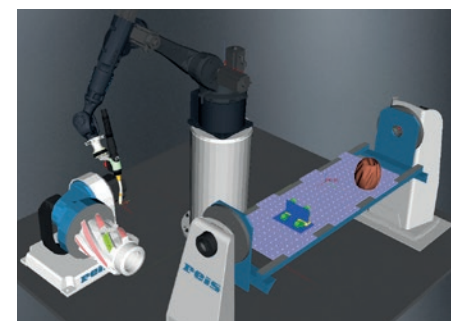
Diagnose Software



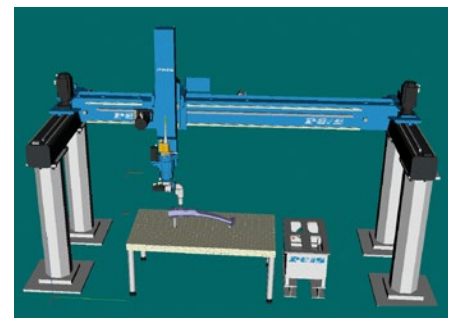
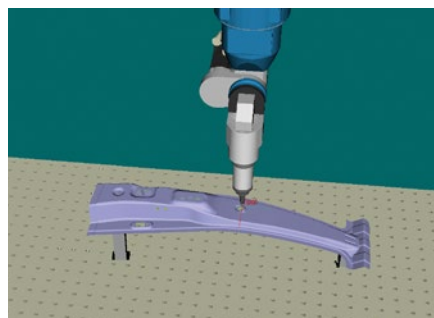
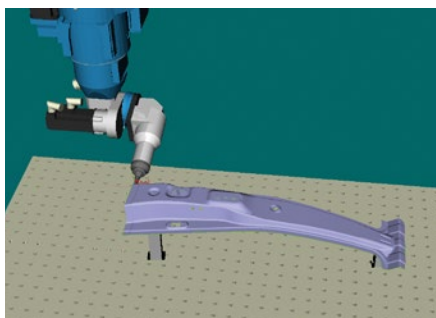
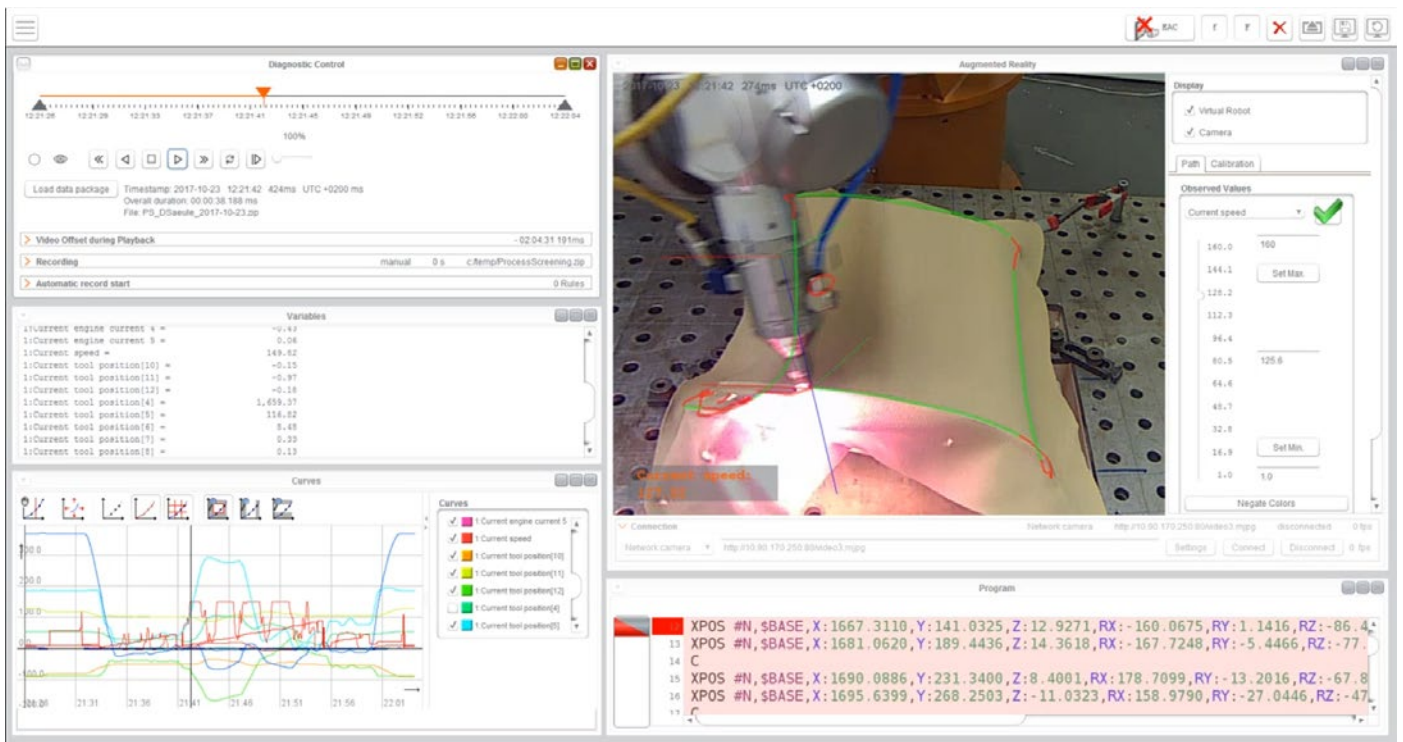
Integrierte SCADA-Funktionalität



3D-Konfigurator für Safety Bereiche



Virtual Reality  
3D-Visualisierung der Roboterzelle



Grafikgestützte Programmierung mit automatischer Bahnerstellung  
online am Roboter und offline auf dem PC

# SIMATIC SAFE KINEMATICS

## Eine Alternative zu Schutzzäunen

Industrieroboter übernehmen immer mehr Handlungsaufgaben. Häufig sind sie nur durch einen Schutzzaun gesichert, doch besserer Schutz kostet wertvollen Platz oder ist teuer. Die Prinzipien von SIMATIC Safe Kinematics beruhen auf den bahnbrechenden Erfindungen der Firma [Reis Robotics](#), den Pionieren der modernen Robotersicherheitstechnik. Diese Software-basierte Lösung kann die Bewegung im mehrdimensionalen Raum von vordefinierten Kinematiken mit bis zu 12 interpolierenden Achsen fehlersicher überwachen – und bietet damit eine sichere und platzsparende Alternative für die Industrie.

### Voll integrierte und zertifizierte Software-basierte Lösung

Um die Sicherheit zu gewährleisten und die Effizienz von Maschinen zu steigern, ist ein sicheres Zusammenspiel zwischen dem Maschinenbediener und der Maschine notwendig.

- SIL3 (IEC 61508),
- SILCL 3 (IEC 62061) und
- PL e/Kategorie 4 (ISO 13849-1)

### Sichere Geschwindigkeitsüberwachung

Mit der sicheren Geschwindigkeitsüberwachung überwachen Sie die kartesischen Geschwindigkeiten einzelner Punkte der Kinematik, z.B. am Tool Center Point oder an Gelenken.

### Sichere Zonenüberwachung

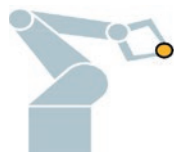
Mit der sicheren Zonenüberwachung überwachen Sie die Position der Kinematik im kartesischen Raum, z.B. um den Verfahrbereich der Kinematik einzuschränken oder um Bereiche zu überwachen, die vom Bedienpersonal betreten werden können.

### Sichere Orientierungsüberwachung

Mit der sicheren Orientierungsüberwachung überwachen Sie an einer anwenderdefinierten seriellen Kinematik die Orientierung des Flansches, z.B. darf ein Werkstück nur bearbeitet werden, wenn das Werkzeug senkrecht zum Fußboden steht.

#### SICHERE GESCHWINDIGKEITSÜBERWACHUNG (SLS)

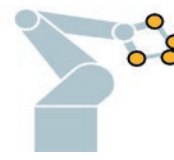
verschiedener Punkte der Kinematik



**SLS-TCP**  
Tool Center Point



**SLS-JOINT**  
Gelenkpositionen



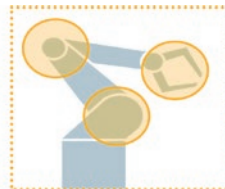
**SLS-TOOL**  
Werkzeugpunkte



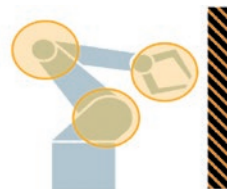
**SLS-POINT**  
Beliebige Punkte an der Kinematik

#### SICHERE ZONENÜBERWACHUNG (SZM)

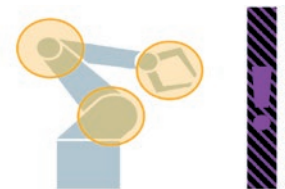
Sichere **Kollisionsprüfung** zwischen Kinematik- und Arbeitsraumzonen



Kinematik innerhalb der Arbeitszone (2D/3D)



Kinematik außerhalb der Warnzone (2D/3D)



Kinematik außerhalb der Schutzzone (2D/3D)

#### SICHERE ORIENTIERUNGSÜBERWACHUNG (SLO)

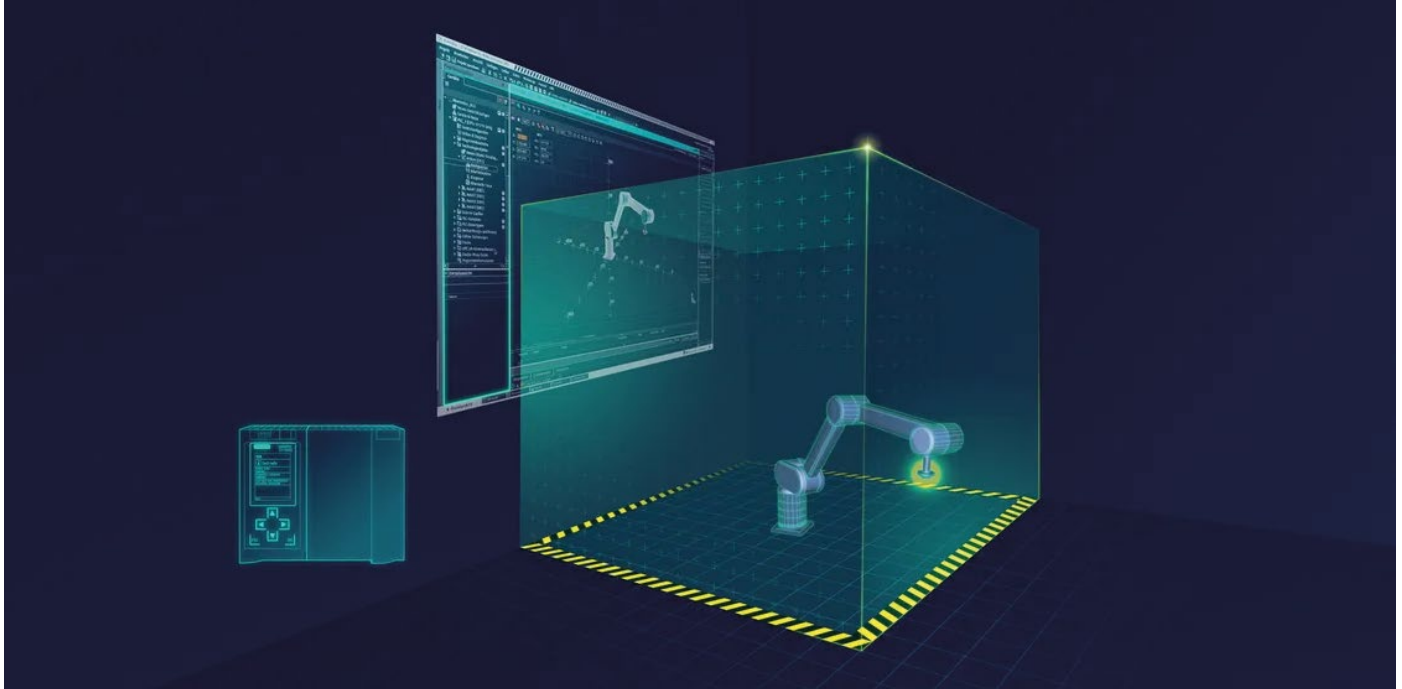
des Flansches für die anwenderdefinierte serielle Kinematik



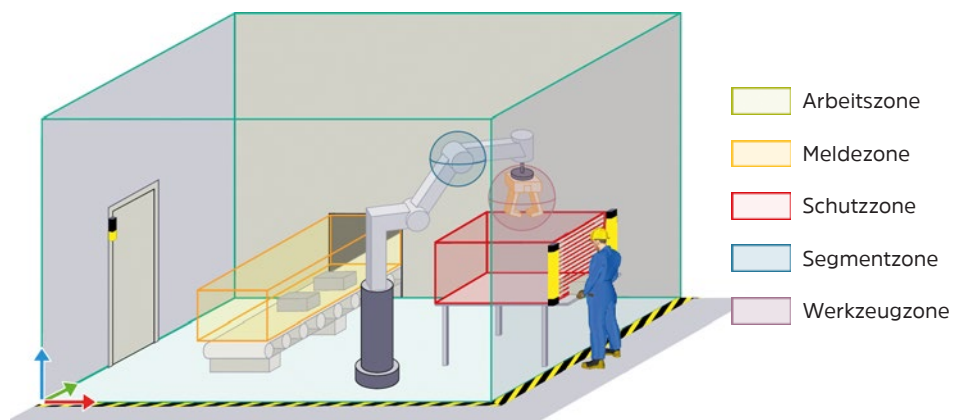
Kinematik innerhalb des kegelförmigen Toleranzbereichs

#### DIE VORTEILE

- kompaktes und flexibles Maschinendesign
- Erfüllung der Sicherheitsanforderungen nach EN ISO 10218
- erhöhte Verfügbarkeit und Produktivität
- durchgängiges Engineering
- kurze Inbetriebnahmezeiten



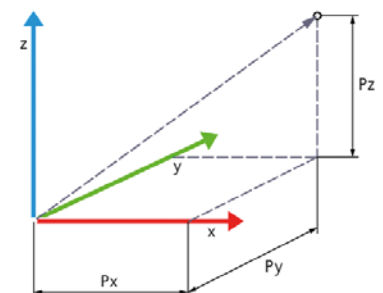
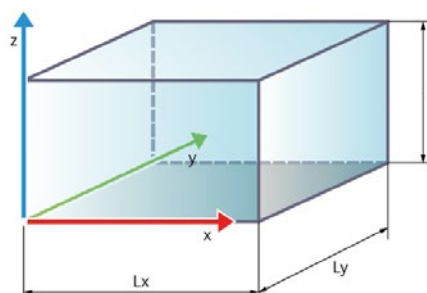
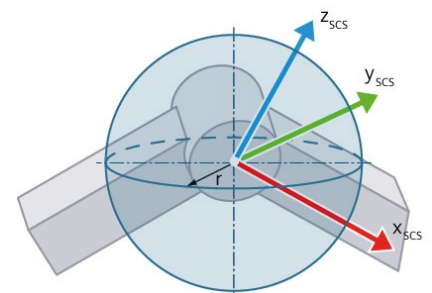
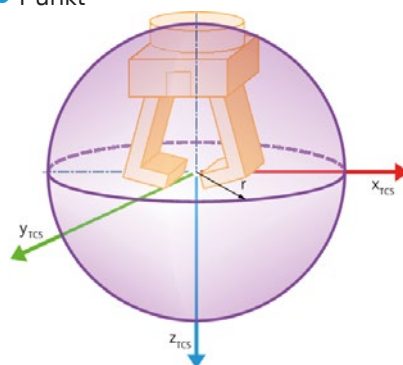
- **Arbeitszonen** definieren Bereiche, in denen sich die Kinematik bewegen darf. Sie können bis zu zehn Arbeitszonen definieren
- **Schutzzonen** sind Bereiche innerhalb des Verfahrraums der Kinematik, in die eine Kinematik nicht eintreten darf (z.B. Schaltschrank, Schutzwand oder Bereiche, in denen sich ein Anlagenbediener bewegt). Sie können bis zu zehn Schutz- oder Meldezonen definieren. Wenn eine Kinematikzone eine Schutzzone verletzt, wird ein Stopp der Kinematik gefordert
- **Meldezonen** sind Bereiche innerhalb des Verfahrraums der Kinematik. Sie können bis zu zehn Schutz- oder Meldezonen definieren. Meldezonen zeigen eine Zonenverletzung durch eine Kinematik an. Dabei wird kein Stopp der Kinematik gefordert
- **Segmentzonen** definieren Sie im jeweiligen Segmentkoordinatensystem (SCS) oder im Flanschkoordinatensystem (FCS). Die folgende Grafik zeigt eine kugelförmige Segmentzone am Beispiel der Kinematik „Knickarm“
- **Werkzeugzonen** definieren Sie im Werkzeugkoordinatensystem (TCS)




## Zonengeometrie

Abhängig vom Zonentyp können Sie Zonen mit folgenden geometrischen Körpern konfigurieren:

- Kugel
- Quader
- Punkt





 Für weitere Informationen kontaktieren Sie uns bitte unter [sales@reisrobotics.com](mailto:sales@reisrobotics.com)

Angaben über die Beschaffenheit und Verwendbarkeit der Produkte stellen keine Zusicherung von Eigenschaften dar, sondern dienen lediglich zu Informationszwecken. Maßgeblich für den Umfang unserer Lieferungen und Leistungen ist der jeweilige Vertragsgegenstand. Die Abbildungen enthalten zum Teil auch optionale Ausstattungen, die nicht zum serienmäßigen Lieferumfang gehören. Technische Daten und Abbildungen unverbindlich für Lieferungen. Änderungen vorbehalten.



Erfahren Sie mehr über  
unser Unternehmen

Follow us on:



**Reis Robotics GmbH & Co. KG**  
Walter-Reis-Straße 1  
63785 Obernburg / Deutschland  
Telefon +49 6022 503-0  
[sales@reisrobotics.com](mailto:sales@reisrobotics.com)

[www.reisrobotics.com](http://www.reisrobotics.com)

A company of **ARETÈ COCCHI TECHNOLOGY**

