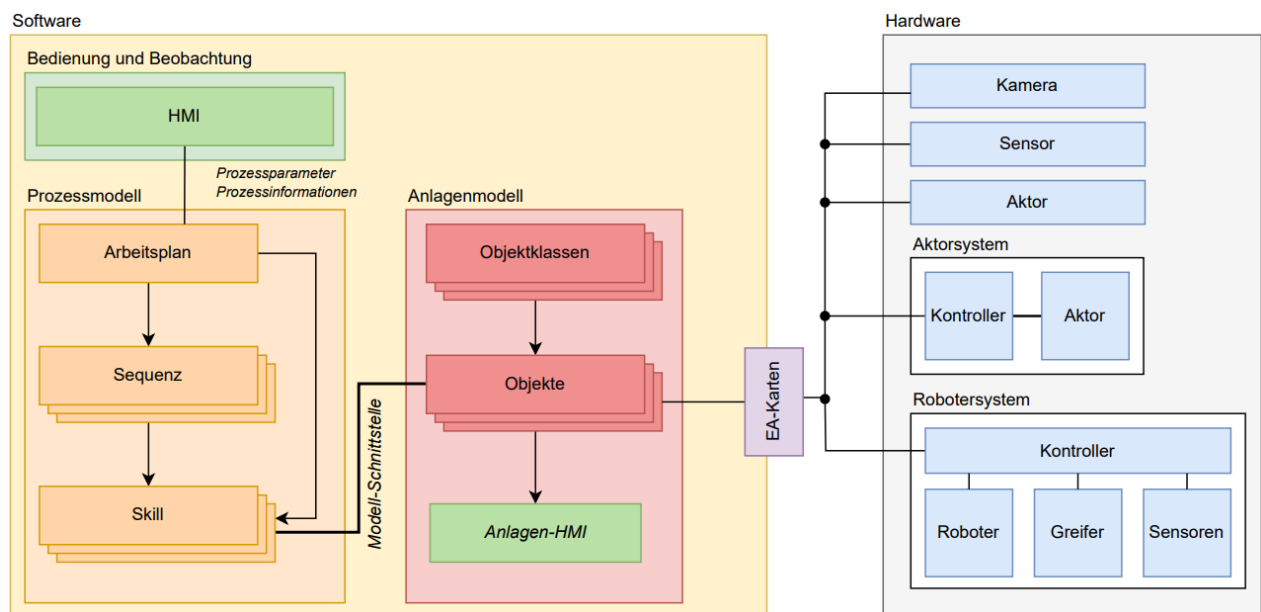


Arbeitspaket 2: Struktur

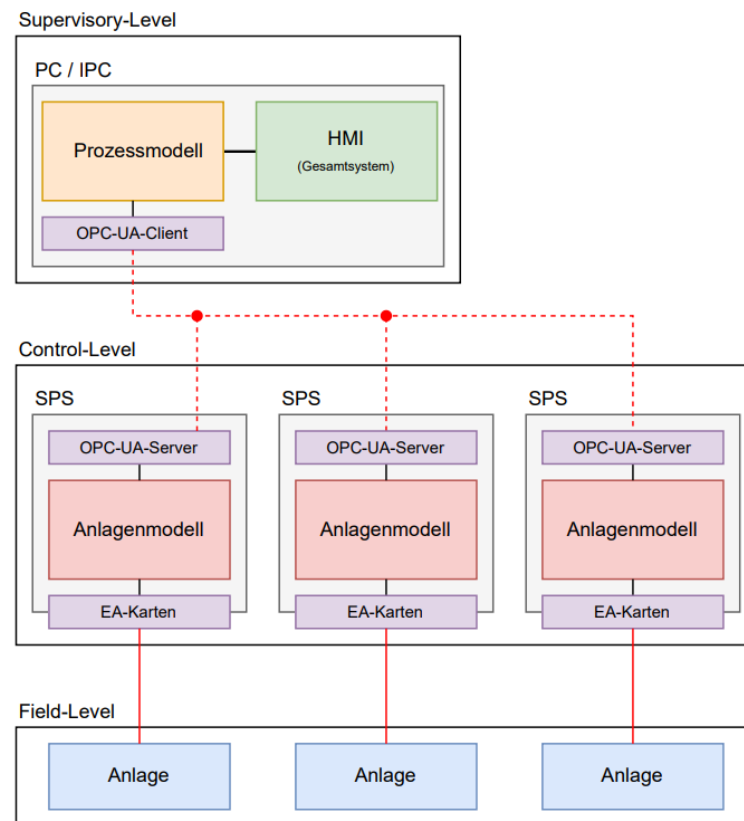
Die Struktur der Software muss auf einen skill-basierten Ansatz ausgelegt werden. Dafür müssen die Anforderungen an eine solche Struktur klar definiert und die Möglichkeiten, die TwinCat bietet, analysiert werden. In einem ersten Schritt wird die allgemeine Grobstruktur des Systems dargestellt. Dieses kann wie folgt definiert werden:



Wie durch die ANSI/ISA-88-Norm vorgegeben (Referenz), besteht die Software aus einem Prozess- und einem Anlagenmodell. Das Prozessmodell steuert den Ablauf, während das Anlagenmodell die Schnittstelle zu den einzelnen Anlagenkomponenten darstellt. Innerhalb des Prozessmodells werden die Skills definiert, die entweder von Sequenzen oder direkt aus dem Arbeitsplan zur Ablaufsteuerung genutzt werden können. Der Arbeitsplan beschreibt den gesamten Prozess.

Das Anlagenmodell implementiert die Objektklassen der verschiedenen Systemelemente und bildet deren Funktionalitäten ab. Die Struktur des Anlagenmodells ist klar und übersichtlich: Die Objektklassen werden instanziiert und diese Objekte mit den entsprechenden Ein- und Ausgängen verknüpft, welche die Schnittstelle zum realen System darstellen. Der Grundgedanke dabei ist, dass alle Funktionalitäten zentral im Anlagenmodell gebündelt werden, um möglichst wenig Funktionalität auf den einzelnen Komponenten selbst zu belassen. Ziel ist es, dass sämtliche Elemente, von Robotern bis zu Kameras, über die SPS gesteuert werden können. Voraussetzung dafür ist, dass alle Komponenten über eine funktionale Schnittstelle zur SPS verfügen. Alle Komponenten werden in der Anlagen-HMI visualisiert und können dort manuell gesteuert werden. Dies ermöglicht es beispielsweise, Roboterpositionen zu speichern, die später von einem Skill für Bewegungsabläufe genutzt werden können.

Abschliessend gibt es eine Bedien- und Beobachtungsebene, die als Benutzeroberfläche dient, um Prozessparameter einzugeben und Prozessinformationen anzuzeigen. Diese Systemstruktur ermöglicht den modularen Betrieb von Teilsystemen. Das folgende Schema zeigt, wie ein solches System aussehen könnte:



Das Schema orientiert sich an der Automatisierungspyramide und zeigt die ersten drei Ebenen (Supervisory-, Control- und Field-Level). Auf dem Supervisory-Level befindet sich das Prozessmodell. Hier wird über die HMI ein Arbeitsplan ausgewählt / zusammengestellt und gestartet. Eine OPC-UA-Schnittstelle übermittelt die prozessrelevanten Daten an die entsprechende SPS im Control-Level. Die SPS steuert die Anlage im Field-Level basierend auf dem Anlagenmodell. Die verschiedenen Anlagen können flexibel für unterschiedliche Aufgaben genutzt werden oder, falls sie identische Fähigkeiten besitzen, nach Auslastung zugewiesen werden. Dadurch ist das System äusserst flexibel und kann ohne grossen Aufwand erweitert werden.

Schnittstellen innerhalb der SPS-Software

Die Software besteht aus den drei Hauptelementen: Bedienung und Beobachtung (HMI), Prozessmodell und Anlagenmodell. Um die Schnittstellen zwischen diesen Elementen zu definieren, müssen die Kompetenzen klar definiert werden. Wer ist für was verantwortlich und welche Informationen werden dafür benötigt.

Bedienung und Beobachtung:

Über das HMI wird der Arbeitsplan erstellt und mit den erforderlichen Ablaufparameter versehen. Das System sowie der erstellte Arbeitsplan können über das HMI gestartet, gestoppt und gesteuert werden.

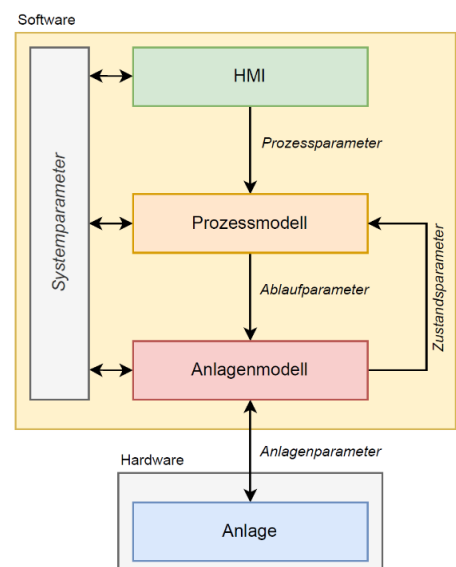
Prozessmodell:

Das Prozessmodell koordiniert die Ausführung des Arbeitsplans. Zusammen mit den Ablaufparameter werden die Prozessparameter festgelegt und weitergegeben. Der Arbeitsplan wird in einzelne Skills unterteilt, die wiederum die grundlegenden Funktionen der Anlagenkomponenten ausführen. Die effiziente und standardisierte Definition eines Skills, vereinfacht die Anwendung dieser in den Sequenzen und Arbeitsplänen.

Anlagenmodell:

Das Anlagenmodell bildet die Funktionalität der Systemkomponenten ab. Es stellt die Funktionen durch Methoden dar, während Zustände und Prozessinformationen über Eigenschaften wiedergegeben werden. Das Anlagenmodell verarbeitet die durch die Skills definierten Prozessparameter und übersetzt sie in Anlagenparameter, mit denen die realen Komponenten der Anlage betrieben werden. Die vom Anlagenmodell zurückgemeldeten Daten (wie Zustände und Messwerte) werden als Zustandsparameter bezeichnet. Diese sind für den Ablauf im Prozessmodell notwendig.

Die drei Hauptelemente müssen klar voneinander abgegrenzt werden. Der modellbasierte Ansatz in der Softwarestruktur bietet dabei mehrere Vorteile (siehe EVA-Referenz). Dank der klaren Struktur und Übersichtlichkeit lassen sich die Prozesse und Abläufe leicht nachvollziehen. Dies erleichtert auch die Kommunikation, da durch die einheitliche Verwendung von Begriffen alle Beteiligten dieselbe "Sprache" sprechen. Darüber hinaus können Risiken früher und einfacher erkannt werden. Die Abgrenzung der Hauptelemente geschieht über die Schnittstellen zwischen diesen. Die Schnittstellen werden durch die verschiedenen Parameter definiert. Der Begriff Parameter ist dabei ein Sammelbegriff für alle definierten Variablen, welche zwischen den Elementen ausgetauscht werden.



Prozessparameter:

Die Prozessparameter beschreiben den durchzuführenden Prozess auf eine möglichst abstrakte Weise. Es werden nur Informationen weitergegeben, welche nötig sind um den Prozess eindeutig zu definieren. Die Prozessparameter hängen dabei von den vorhandenen Skills und deren Fähigkeiten ab. Die genauen Parameter werden während der Erarbeitung des Prozessmodells definiert.

Ablaufparameter:

Ablaufparameter werden durch die Skills definiert und definieren relevante Parameter für den Ablauf. Die Informationen sind jedoch nicht konkret auf die Komponenten im System ausgelegt. Die Ablaufparameter sind noch Anlagenunabhängig und hängen z.B. nicht vom Typ des Roboters ab, welcher im System eingesetzt wird. Die genauen Parameter werden während der Erarbeitung des Prozessmodells definiert.

Anlagenparameter:

Die Parameter, welche von den instanziierten Objekten im Anlagemodell vorbereitet werden, dienen als Schnittstelle zur realen Anlage und somit nun anlagenspezifisch. Diese Art dieser Parameter hängt von den eingesetzten Komponenten im System ab. Die genauen Parameter werden während der Erarbeitung des Anlagenmodells definiert

Zustandsparameter:

Die durch die instanziierten Objekte ausgewerteten und verarbeiteten Anlageparameter werden als Zustandsparameter an das Prozessmodell zurückgegeben. Auf diese Parameter reagiert der Skill wie auch das System. Die genauen Parameter werden während der Erarbeitung des Anlagenmodells definiert

Systemparameter:

Systemparameter sind Systemübergreifende Parameter, welche zur Bedienung des gesamten Systems verwendet werden oder dessen Zustand darstellen.

Interaktionen innerhalb der Software

Um die Schnittstellen innerhalb der Software besser zu verstehen, muss die Interaktion zwischen Systemparameter, Prozessmodell (Skills) und Anlagenmodell (Objekten) abgegrenzt sein, da diese die wichtigsten Schnittstellen im System darstellen. Die Interaktion findet dabei mit 3 Schnittstellen statt.

Objektschnittstelle:

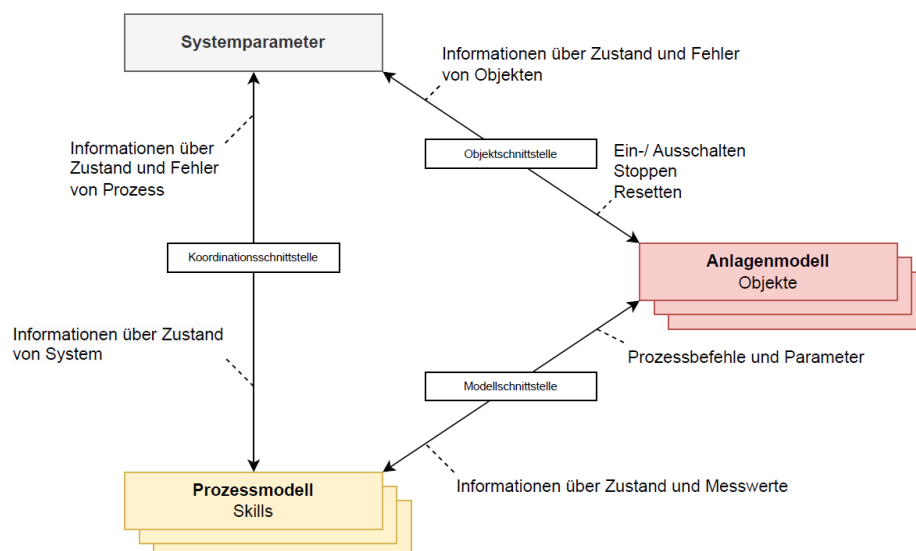
Die Objektschnittstelle regelt die Interaktion zwischen den Systemparametern und den Objekten des Anlagenmodells. Die Systemparameter steuern dabei die grundlegenden Funktionen der Objekte, wie Ein- und Ausschalten, Zurücksetzen oder Stoppen. Da diese Basisfunktionen nicht durch die Skills aktiviert werden, bleibt deren Aufgabe auf die Verwaltung des Prozesses beschränkt. Dies ist besonders sinnvoll, da ein Objekt mehrere Skills besitzen kann, und so Fragen zur Berechtigung der Skills vermieden werden. Im Gegenzug stellen die Objekte den Systemparametern Informationen über ihren Zustand und Fehler zur Verfügung.

Modellschnittstelle:

Die Modellschnittstelle ist für die Interaktion zwischen Prozessmodell und Anlagenmodell zuständig, genauer gesagt zwischen Skills und Objekten. Die Skills schicken Ablaufparameter an das Objekt, auf welche das Objekt reagiert. Das Objekt übergibt den aktuellen Zustand. Zusätzlich werden auch Messwerte vom Objekt an den Skill übergeben.

Koordinationsschnittstelle:

Die Koordinationsschnittstelle ist für die allgemeine Prozesskoordination verantwortlich. Es werden Informationen über den aktuellen Zustand und Fehler der Skills an die Systemparameter übergeben. Der Skill erhält den aktuellen Zustand des Systems. Der Skill kann somit auf systemübergreifende Situationen reagieren und das System kann auf Skill-Zustände reagieren.

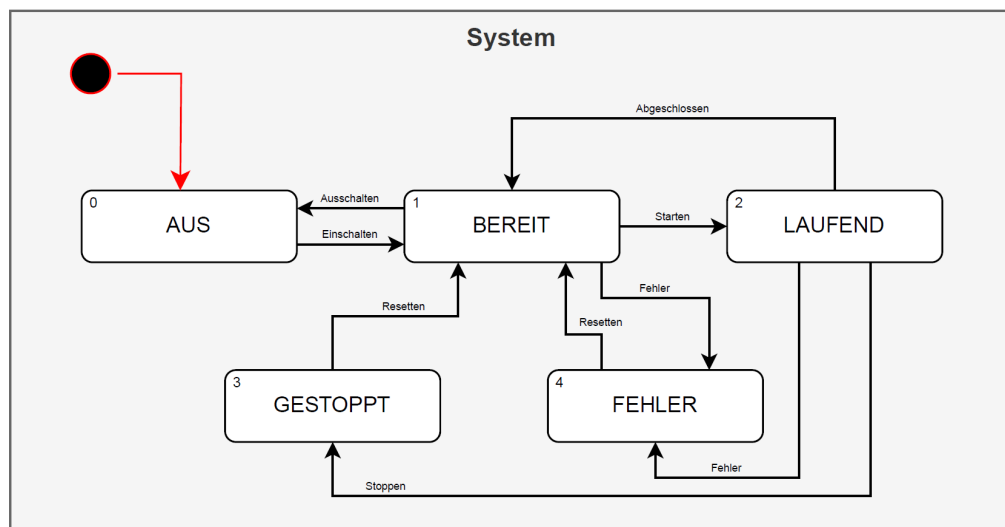


Die Schnittstellen und deren Abgrenzung dienen als Grundlage für die Bestimmung der Zustände. Dabei werden die Zustände für das System, die Skills und die Objekte bestimmt. Die System- und Objektzustände sind entscheidend für die grundlegende Struktur der Skills, da diese auf die jeweiligen Zustände reagieren müssen. Folglich stellen die definierten System- und Objektzustände lediglich die Mindestanforderungen dar, die notwendig sind, um eine Interaktion mit den Skills zu ermöglichen. Bei der Entwicklung der jeweiligen Systemelemente (Prozessmodell und Anlagenmodell) können noch weitere Zustände dazukommen.

System:

Das System besitzt mindestens folgende 5 Zustände:

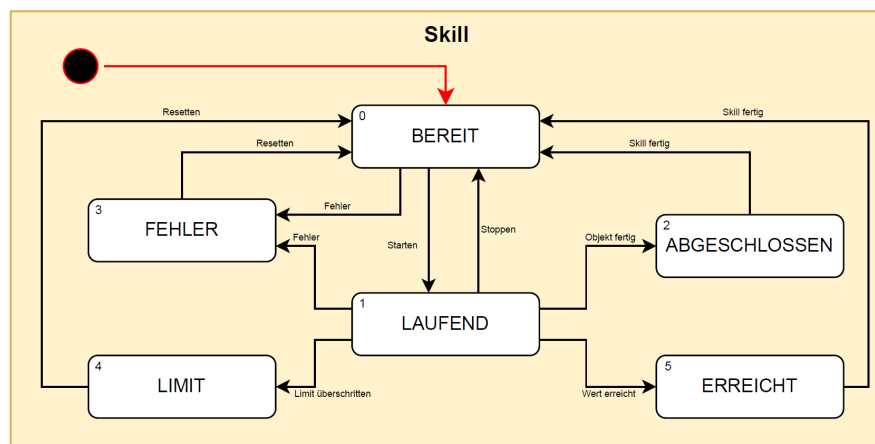
Zustand:		Beschreibung:
0	AUS	Das System ist ausgeschaltet (Startzustand)
1	BEREIT	Das System ist eingeschaltet und bereit einen Prozess durchzuführen
2	LAUFEND	Ein Prozess wird ausgeführt
3	GESTOPPT	Ein Prozess wurde gestoppt
4	FEHLER	Es gibt einen Fehler im System



Skill:

Ein Skill besitzt 6 Zustände:

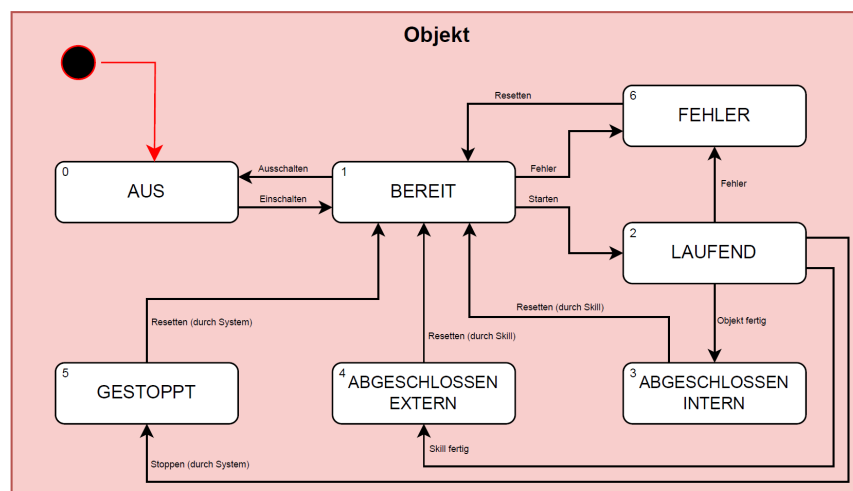
Zustand:	Beschreibung:
0 BEREIT	Der Skill ist bereit einen Prozess auszuführen (Startzustand)
1 LAUFEND	Der Skill führt einen Prozess aus
2 ABGESCHLOSSEN	Der Prozess wurde abgeschlossen (Durch Objekt abgeschlossen)
3 ERREICHT	Prozessziel wurde erreicht und Prozess wurde durch Skill beendet
4 LIMIT	Grenzwert wurde überschritten und Prozess wurde abgebrochen
5 FEHLER	Es gibt einen Fehler bezüglich des Prozesses



Objekt:

Ein Objekt benötigt mindestens folgende 6 Zustände:

Zustand:	Beschreibung:
0 AUS	Das Objekt ist ausgeschaltet (Startzustand)
1 BEREIT	Das Objekt ist eingeschaltet und bereit
2 LAUFEND	Das Objekt ist aktiv
3 ABGESCHLOSSEN_INTERN	Das Objekt hat den Prozess durchgeführt und hat selbständig gestoppt
4 ABGESCHLOSSEN_EXTERN	Das Objekt wurde durch den Skill gestoppt (ERREICHT / LIMIT)
5 GESTOPPT	Das Objekt wurde durch das System gestoppt
6 FEHLER	Es gibt einen Fehler bezüglich des Objektes



Fazit:

Die Softwarestruktur orientiert sich an den Prinzipien der ANSI/ISA-88-Norm, um einen möglichst klaren und anlagenunabhängigen Aufbau zu gewährleisten. Im Zentrum der Software stehen das Prozessmodell und das Anlagenmodell. Die Benutzeroberfläche wird durch das Bedien- und Beobachtungselement (HMI) dargestellt. Diese drei Kernelemente kommunizieren über definierte Schnittstellen miteinander, wobei alle Schnittstellen – mit Ausnahme der Anlagenparameter – anlagenunabhängig sind. Unabhängig von den eingesetzten Komponenten bleiben die anlagenunabhängigen Schnittstellen stets gleich. Allerdings müssen die Objektklassen im Anlagenmodell auf die jeweiligen Komponenten abgestimmt werden, was einen gewissen Programmieraufwand erfordert.

Damit ist die allgemeine Struktur der SPS-Software festgelegt. In weiteren Arbeitspaketen werden die drei Hauptelemente detailliert ausgearbeitet und näher betrachtet.