

Software Architektur P3AT - Verhalten

Inhaltsverzeichnis

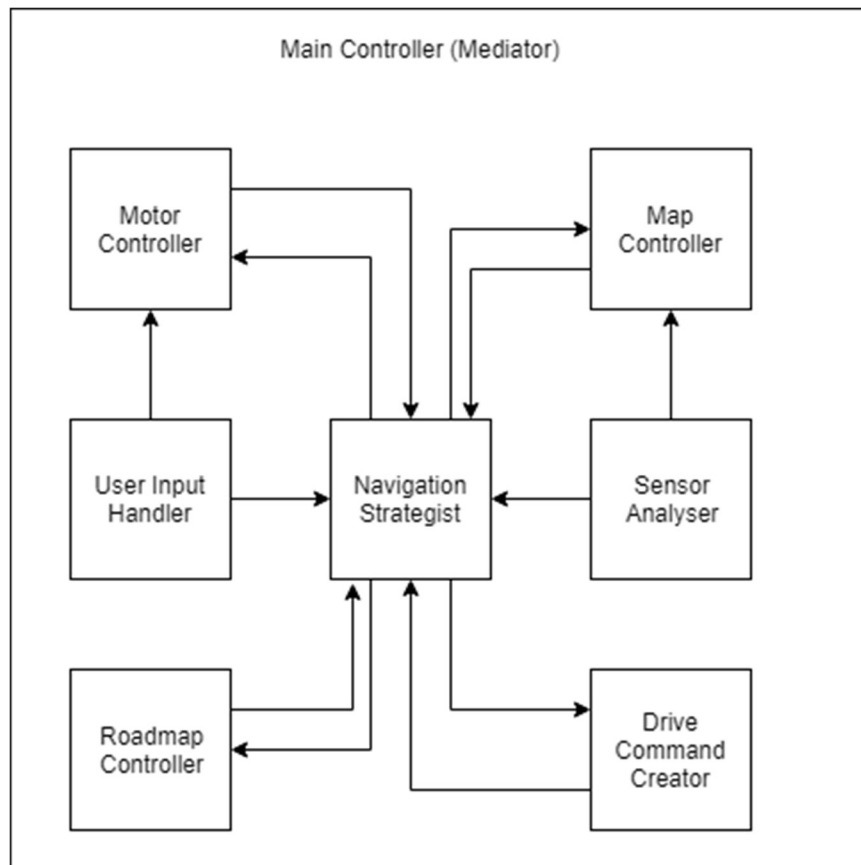
Vorwort:	1
Blockschaltbild Architektur:	2
Modulbeschreibung:	2
Main Controller:	2
Navigation Strategist:	2
Roadmap Controller:	3
Drive Command Creator:.....	3
Map Controller:	3
Motor Controller:	3
Sensor Analyser:	4

Vorwort:

Dieses Dokument dient der Beschreibung der Verantwortlichkeiten und Zusammenhänge zwischen den einzelnen Modulen der Robotersteuerung für den P3AT.

Dies ist der erste Entwurf der Architektur und stellt noch keine vollständige/endgültige Systembeschreibung dar!

Blockschaltbild Architektur:



Modulbeschreibung:

Main Controller:

Der Main Controller wurde als Mediator konzipiert und regelt die Kommunikation zwischen den einzelnen Modulen. So muss zum Beispiel geregelt werden, wann ein neuer Befehl ausgeführt wird, oder dem Navigation Strategist melden, wenn der Sensor Analyser meldet, dass ein Objekt im Weg des Roboters auftaucht.

Navigation Strategist:

Der Navigation Strategist bestimmt welcher Weg gerade verfolgt wird (Lokaler / Globaler Pfad). Dieser ist unterteilt in die globale und die lokale Strategie. Die Globale Strategie verfolgt das ursprüngliche Ziel des Roboters. Dies können unter anderem vorgegebene Koordinaten sein, welcher der Roboter abfahren soll.

Sollte nun ein (lokales) Hindernis auftauchen, zum Beispiel wenn etwas im Weg des Roboters steht, so übernimmt die Lokale Strategie und versucht einen Weg um das Problem zurück zum Globalen Pfad zu finden. Weiters soll die Lokale Strategie verhindern, in ein Objekt zu crashen.

Ausgelöst wird der Wechsel von der globalen zur lokalen Strategie vom Sensor Analyser. Wurde die Problemstelle umfahren und ein Weg zurück zur ursprünglichen Strecke gefunden, so wird zurück zur globalen Strategie gewechselt.

Die Globale Strategie holt sich vom Roadmap Controller die nächste anzufahrende Koordinate und schaut ob sich auf den Weg dorthin bekannte Problemstellen befinden (Holen der Karte vom Map

Controller). Sollte dies der Fall sein, werden Zwischenkoordinaten in die Roadmap eingefügt. Anschließend wird dem Drive Command Creator die aktuelle Position und das nächste Ziel mitgeteilt und dieser errechnet ein Command(Composite) welches zum Ansteuern der Motoren verwendet wird. Dieses Command(Composite) wird an dann zum Motor Controller weitergeleitet und auf die Durchführung gewartet.

Sollte in der Zeit der Ausführung des Commands ein Problem auftreten, muss dementsprechend vom Navigation Strategist gehandelt werden.

Roadmap Controller:

Der Roadmap Controller speichert und verwaltet die Zielkoordinaten (Koordinaten-Tuples). Ihm kann mitgeteilt werden, neue einzufügen und alte erledigte Ziele zu löschen.

Im ersten Schritt muss der Roadmap Controller mit Koordinaten geladen werden.

→ Zu einem späteren Punkt im Projekt soll es auch möglich sein ohne einer bestimmten Zielkoordinate einen Auftrag zu erledigen.

Drive Command Creator:

Diesem Modul werden die aktuelle Ausrichtung und Position, sowie die Zielkoordinate mitgeteilt und berechnet daraus ein Command(Composit) um dieses Ziel zu erreichen.

Map Controller:

Der Map Controller ist zuständig zum Speichern von abgefahrenen Strecken und bekannten Problemstellen. Im Prinzip werden hier die ermittelten Streckeninformationen abgespeichert.

Diese Karte soll immer nach Erreichen einer Koordinate oder einer Problemstelle, oder nach Gewinnung anderer wichtigen Informationen für die Karte upgedatet werden.

Die Karte besteht aus Prozentuellen Werten, mit welcher Sicherheit sich auf einem bestimmten Punkt ein Objekt befindet. Ab einem bestimmten Schwellenwert (~50%??) gilt eine Position als belegt / nicht befahrbar.

Motor Controller:

Der Motor Controller interpretiert die erhaltenen Commands und steuert die Motoren des Roboters dementsprechend an. Weiters meldet er zurück, wenn ein Befehl ausgeführt wurde. Sollte vom Navigation Strategist ein plötzliches Stop oder eine Unterbrechung des Befehls kommen, muss der Motor Controller auch mitteilen, wie weit gefahren wurde, um so die Position des Roboters auf der Karte bestimmen zu können.

Sensor Analyser:

Dieses Modul meldet dem Navigation Strategist und dem Map Controller, wenn ein Problem vorliegt (Objekt im Weg).

Für jede Art von Sensor ist eine Unterklasse zu implementieren.

Wie jedoch hier genau die Rückgabewerte und Funktionen aussehen sollen, um diese möglichst generalisieren zu können ist zum jetzigen Zeitpunkt noch nicht entschieden.

Es soll auf jeden Fall beim Auftauchen eines Objekts die Position von diesem Objekt ermittelt und in die Karte eingetragen werden, als auch der Navigation Strategist alarmiert werden, damit dieser gegebenenfalls darauf reagieren kann.