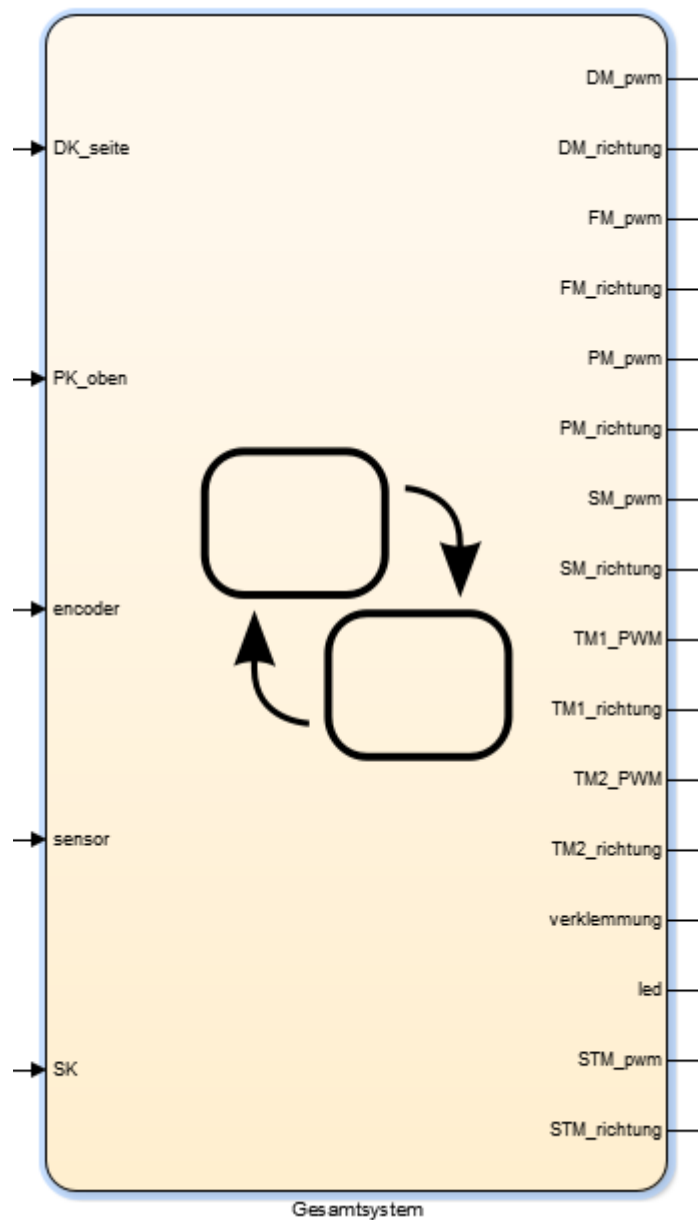


Dokumentation Gruppe 7 B

Allgemeine Info:

Das Hauptprogramm besteht aus einem großen Simulink-Block, in welchem die Teilsysteme als einzelne State-Blöcke mit UND-Verknüpfung implementiert sind. Für die Simulation sind die schon in Simulink gegebene Signal und Routing Blocks verwendet.



Namen:

- TS1,2,3 – Teilsystem 1,2,3
- *Taster:*
 - DK_seite: Der Taster an der Seite der Drehplattform. Wird verwendet zum Stoppe der rotierende Bewegung der Plattform, wenn sie die richtige Stelle erreicht.
 - PK_oben: Der Taster an der Turmvorrichtung, der ganz oben am Ende der Führung steht. Dieser Taster dient zum Stoppen der translatorischen Bewegung nach oben der Turmvorrichtung, wenn das Ende erreicht ist.
 - SK: Der Taster hinter der Stab, der die Würfel hält nicht auszufallen. Zeigt, wenn der Stab herausgenommen ist. (Es kann sein, dass dieser Taster in späteren Implementationen nicht mehr verwendet wird).
- *Signale (Input):*
 - encoder: Das Signal des Encoders des Motors an dem Förderbahn. Wird benutzt zum Erkennen von Verklemmungen des Förderbahns.
 - sensor: Das Signal von der Lichtschranke, die erkennt, wenn ein Würfel zur Eingabe in der Turmvorrichtung vorhanden ist.
- *Signale (Output):*
 - verklemmung: Zeigt wenn eine Verklemmung an dem Förderbahn passiert (nur zum Debuggen und interne Sachen verwendet).
 - led: Die Zustand-LED, die leuchtet wenn das Programm fertig ist.

Gruppe der Motoren:

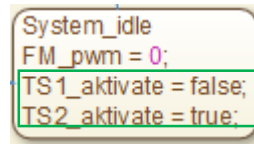
- **MOTOR_INITIALEN*_pwm*: PWM des Motors geben in Prozent
- **MOTOR_INITIALEN*_richtung*: Richtung des Motors (am meistens 0 nach vorne, 1 nach hinten, aber hängt von dem Verwendung ab. In dem Statefow Block ist es für jeden Motor besser erklärt)

Motor Initialen:

- DM: Der Motor für die rotierende Bewegung der Turmvorrichtung
- FM: Der Motor für die Steuerung des Förderbahns
- PM: Der Motor für die Steuerung der Ladeplattform des Turmovrrichtung
- SM: Der Motor für die Steuerung des Turmschiebers
- TM1, TM2: Die Motoren für die translatorische Bewegung der Turmvorrichtung
- STM: Der Motor für die Bewegung des Stabs an der Turmvorrichtung

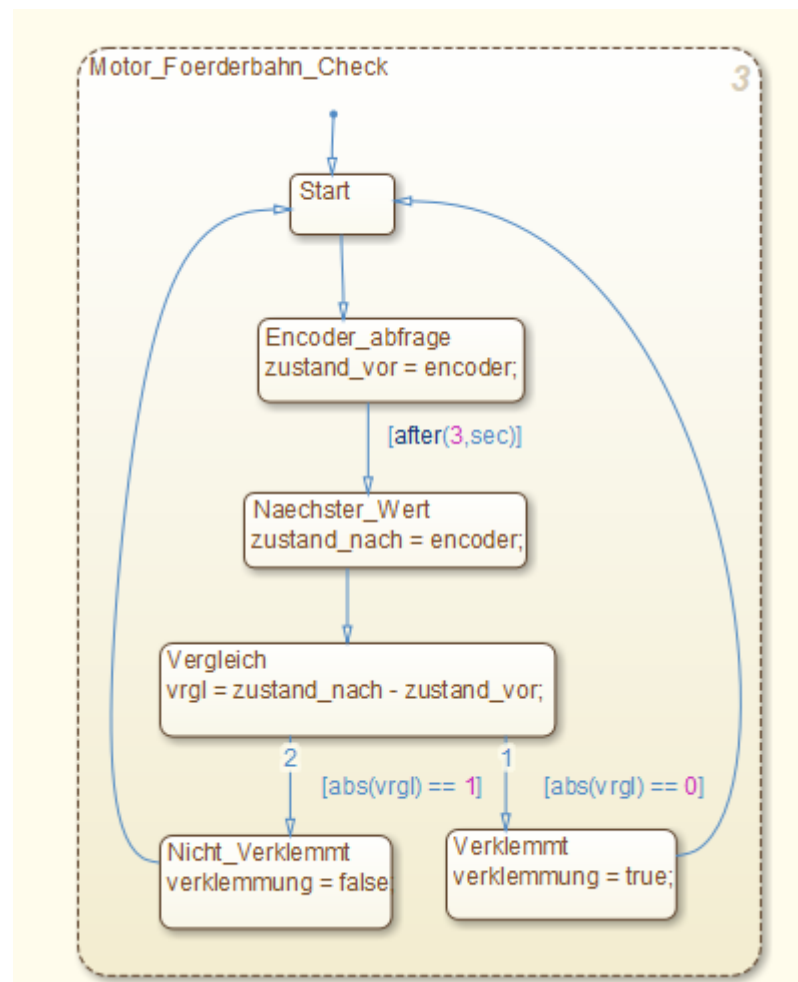
Funktionsweise

In dem Gesamtsystem-Block befinden sich vier Stateflow State-Blocks. Drei davon sind für die einzelne Teilsysteme zuständig (Teilsystem_1, Teilsystem_2, Teilsystem_3) und der Vierte ist der Verklemmungshandler des Förderbahns (Motor_Foerderbahn_Check). Alle State-Blocks der Teilsysteme sind zu jeder Zeit aktiv (UND-Verknüpfung), die aber intern in einem Zustand gesperrt sind und nur Ein davon in eine Schleife geht. Wenn ein bestimmtes Ereignis passiert, aktiviert der Block durch „Token passing“ das nächste Teilsystem. Der letzte aktive Block wird intern gesperrt, bis er von dem Block entsperrt wird, der er aktiviert hat. Das passiert bis die zwei Türme gebaut sind. Dann das Programm wird ganz gestoppt und eine LED leuchtet zur Signalisierung des Endes.



Beispiel von Teilsystem 1:

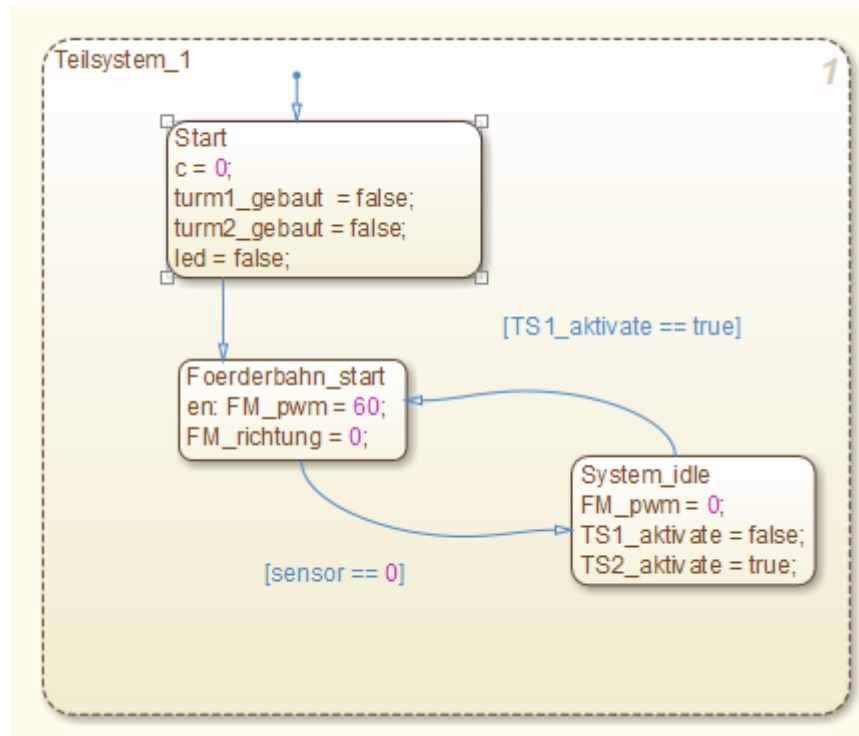
Motor_Foerderbahn_Check:



Da im Moment die Verklemmungen im System durch die mechanische Lage der Komponente behandelt sind, wird der Verklemmungshandler einfach übersprungen. Der Block ist, aber, nicht gelöscht, weil das Gesamtsystem nicht genug in der Wirklichkeit getestet ist und es immer noch die Möglichkeit besteht, Verklemmungen durch Systemänderungen zu passieren. Er kann in der Zukunft hilfreich sein, indem er Zeit zur Implementierung spart.

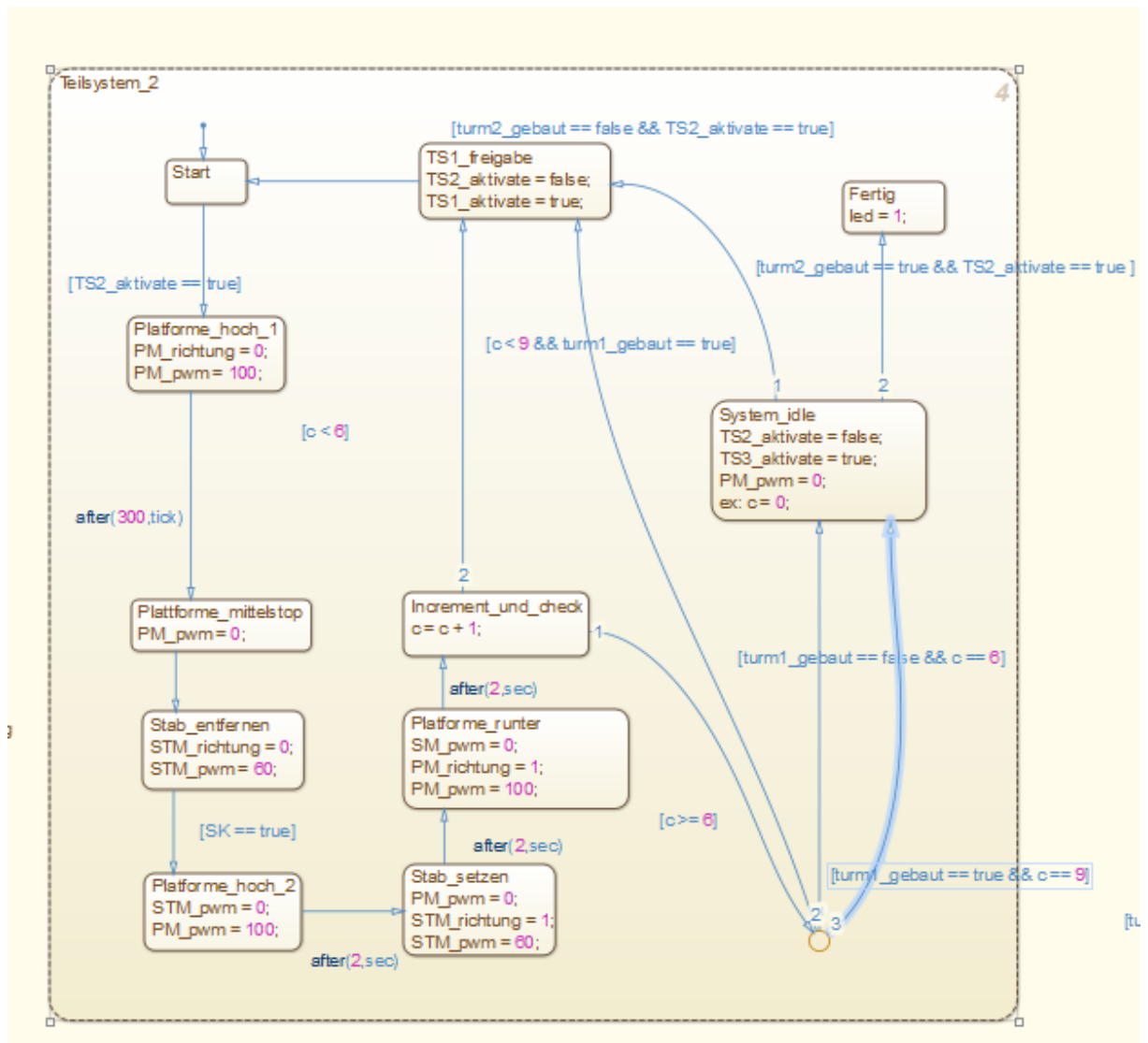
Da der FT-Motor mit Encoder für der Förderbahnsteuerung verwendet wird, kann man über das Signal von dem Encoder feststellen, ob der Motor sich dreht, oder nicht, bzw. ob es Verklemmung gibt. Der Block schaut auf dem Eingang des Encoders und vergleicht das Signal in zwei verschiedenen Zeitpunkten (im dem Beispiel oben ist die Zeitdifferenz 3 Sekunden, aber sie kann variieren). Der Encoder gibt Pulse aus und somit soll der Betrag der Differenz immer 1 sein, wenn der Motor sich dreht. Falls der Betrag 0 ist, bedeutet, dass das Signal immer gleich ist, also der Motor gestoppt ist, oder es eine Verklemmung gibt.

Teilsystem 1:



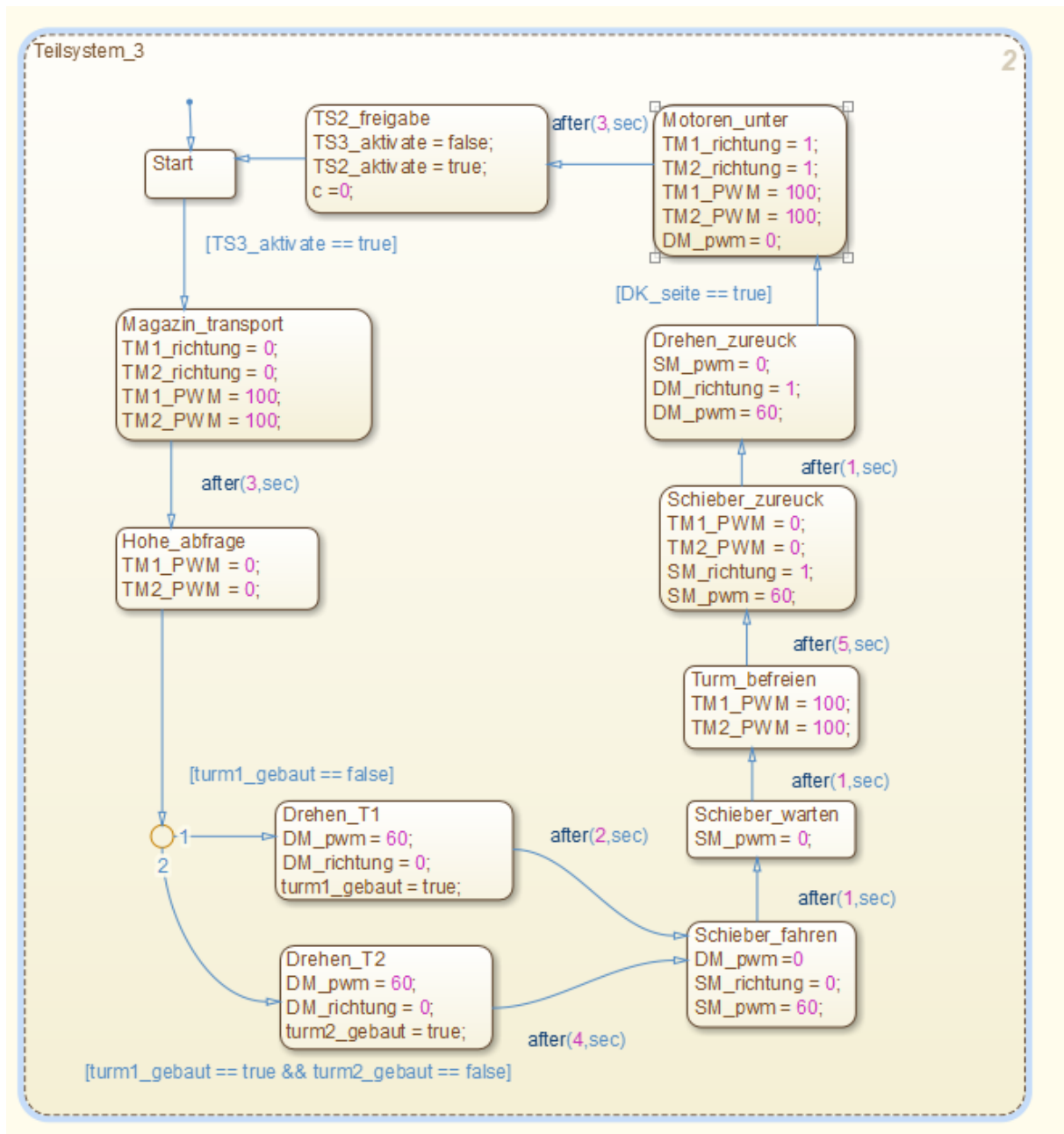
Die default Werte sind beim Starten dieses Teilsystems gegeben, dann das Teilsystem Steuert der Förderbahn, biss die Lichtschanke durch einen Würfel gebrochen wird. Dann der Sensor gibt ein „LOW-Pegel“ aus und das Teilsystem springt in „Idle“-Zustand, indem das Teilsystem 2 aktiviert ist.

Teilsystem 2:



Das Teilsystem wartet auf das Token von TS1, bzw. wartet auf einen Würfel. Wenn er vorhanden ist, dann wird eine Plattform gesteuert, die der Würfel in einem Magazin bringt. Der Anzahl der Würfel wird in diesem Teilsystem gezählt. Wenn in dem Magazin nicht 6 (für Turm 1), oder nicht 9 (Turm 2) würfeln gibt, wird TS1 wieder von diesem Teilsystem aktiviert und das passiert, bis in der Turmvorrichtung genug Würfel zum Bauen gibt. Wenn es schon genug Würfel gibt, dann wird TS3 aktiviert. Von TS3 bekommt man durch die globale Variablen Information darüber, welche Türme gebaut sind (turn1_gebaut und turn2_gebaut). Wenn die beide Variablen auf „true“ gesetzt werden, sperrt sich das Teilsystem (und alle andere TS auch). Das Programm geht in dem „Fertig“ Zustand, wo eine LED geleuchtet wird.

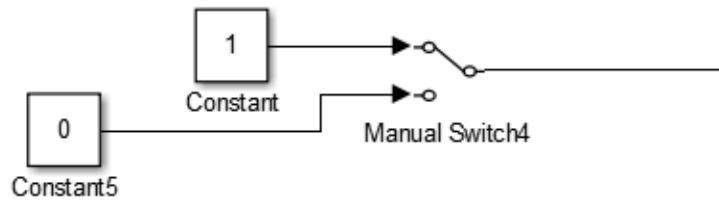
Teilsystem 3:



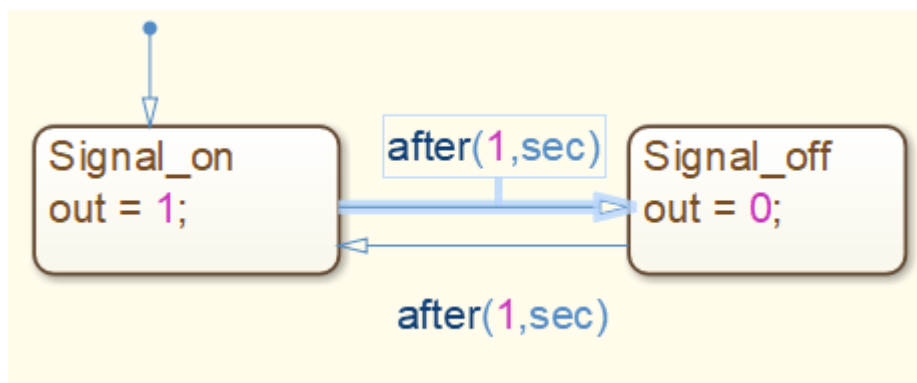
Das Teilsystem bekommt der Token zum Aktivieren, wenn es 6 oder 9 Würfel in dem Magazin der Turmvorrichtung gibt. Dann werden die Motoren zum Fahren nach oben mit der maximalen Geschwindigkeit gefahren. Im Moment sind sie nur zeitlich gesteuert und erfüllen ihren Zweck zwar gut, aber die Idee für eine Encoder-Regelung ist noch im Kopf gehalten, wenn man eine sicherer Implementierung haben will. Wenn sie auf die richtige Höhe sind, stoppt das Programm die Bewegung. Danach steuert das TS wieder zeitlich den Motor, der für die rotierende Bewegung der Turmvorrichtung zuständig ist, bis die gewünschte Winkel erreicht wird. Wenn das passiert, wird das Turm nach vorne geschoben, damit die Würfeln das Hochhaus als Untergrund haben. Die Plattform wird noch bisschen nach oben gefahren, bis der Turm von dem Magazin befreit ist. Der Schiebermechanismus geht zurück, die Plattform wird bis zur Startposition gedreht und gefahren und TS2 wird entsperrt.

Simulation in Simulink

- Taster Eingänge:
 - Die Pegeländerung, die beim Drucken eines Taster passiert, wird in Simulink durch zwei Konstanten (0 – LOW, 1 – HIGH) simuliert, indem man durch ein „Manual Switch“ zu jeder Zeit wählen der Simulation zwischen die beide Konstanten wählen kann.



- Encoder Simulator
 - Wird durch ein Stateflow Chart simuliert. Das Ausgang des Stateflow Blocks ändert seinen Wert jede Sekunde auf 0 oder 1.



- Ausgangssignale
 - Der Wert aller Ausgangssignale ist in einem Simulink Scope Block während und nach der Simulation zu sehen.

