



Entwicklung eines TIA-Projektes

Hausarbeit Industrielle Bussysteme

Studiengang Elektrotechnik

Studienrichtung Automation

Duale Hochschule Baden-Württemberg Ravensburg, Campus Friedrichshafen

von

Simon Schäffler, Alexander Drexler und Florian Prumbs

Abgabedatum:	28. November 2024
Bearbeitungszeitraum:	15.11.2024 - 06.12.2024
Matrikelnummer Simon Schäffler:	5710369
Matrikelnummer Alexander Drexler:	3982016
Matrikelnummer Florian Prumbs:	1848162
Kurs:	FN -TEA22
Ausbildungsfirma:	Webasto Roof & Components SE

Erklärung

gemäß Ziffer 1.1.13 der Anlage 1 zu §§ 3, 4 und 5 der Studien- und Prüfungsordnung für die Bachelorstudiengänge im Studienbereich Technik der Dualen Hochschule Baden-Württemberg vom 29.09.2017 in der Fassung vom 25.07.2018.

Wir versichern hiermit, dass unsere Hausarbeit mit dem Thema:

Entwicklung eines TIA-Projektes

selbstständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt wurden.

Friedrichshafen, den 28. November 2024

Simon Schäffler

Alexander Drexler

Florian Prumbs

Konzeptentwurf

State Machine

Initialzustand

Das System startet im Initialzustand und wechselt direkt in den sogenannten Idle-Zustand. Im Idle-Zustand sind beide LEDs der Anzeige AC2398 ausgeschaltet. In diesem Zustand kann das System je nach den erkannten Eingaben oder Ereignissen in andere Zustände wechseln. Wenn der grüne Knopf gedrückt wird und kein RFID-Tag erkannt wird, speichert das System die aktuelle Systemzeit und bleibt im Idle-Zustand. Dieser Vorgang wird im Diagramm als "SSave Time" bezeichnet.

Tag-Erkennung und Verarbeitung

Wird ein NFC-Tag erkannt und liegt kein Fehlerzustand vor, wechselt das System in den Zustand "Tag Detected Handling". In diesem Zustand blinkt die grüne LED mit einer Frequenz von einer Sekunde, um anzuzeigen, dass ein Tag erkannt wurde. Es gibt zwei mögliche Aktionen, vorausgesetzt, es liegt kein Fehlerzustand vor:

Write Tag Handling: Wenn der grüne Knopf gedrückt wird, während das RFID-Tag erkannt wird und die Systemzeit vorhanden ist, wechselt das System in den

Zustand "Write Tag Handling". In diesem Zustand wird die zuvor gespeicherte Systemzeit auf das RFID-Tag geschrieben. Die grüne LED leuchtet dauerhaft, um anzuzeigen, dass der Schreibvorgang erfolgreich abgeschlossen wurde. Sobald das RFID-Tag nicht mehr erkannt wird, kehrt das System in den Idle-Zustand zurück.

Delete Tag Handling: Alternativ kann der rote Knopf gedrückt werden, während das RFID-Tag erkannt wird und kein Fehlerzustand vorliegt. In diesem Fall wechselt das System in den Zustand "Delete Tag Handling". Hier werden die gespeicherten Daten des RFID-Tags gelöscht, und beide LEDs leuchten dauerhaft, solange das Tag erkannt wird. Sobald das RFID-Tag nicht mehr erkannt wird, kehrt das System in den Idle-Zustand zurück.

Fehlerbehandlung

Wenn während des Prozesses ein Fehlerzustand auftritt, wechselt das System in den Zustand "Error State Handling". In diesem Zustand blinkt die rote LED mit einer Frequenz von einer Sekunde, während die grüne LED ausgeschaltet bleibt, um den Fehler anzuzeigen. Der Fehlerzustand bleibt bestehen, bis die Ursache des Fehlers behoben ist. Anschließend erreicht das System den Finalzustand und der gesamte Ablauf beginnt erneut.

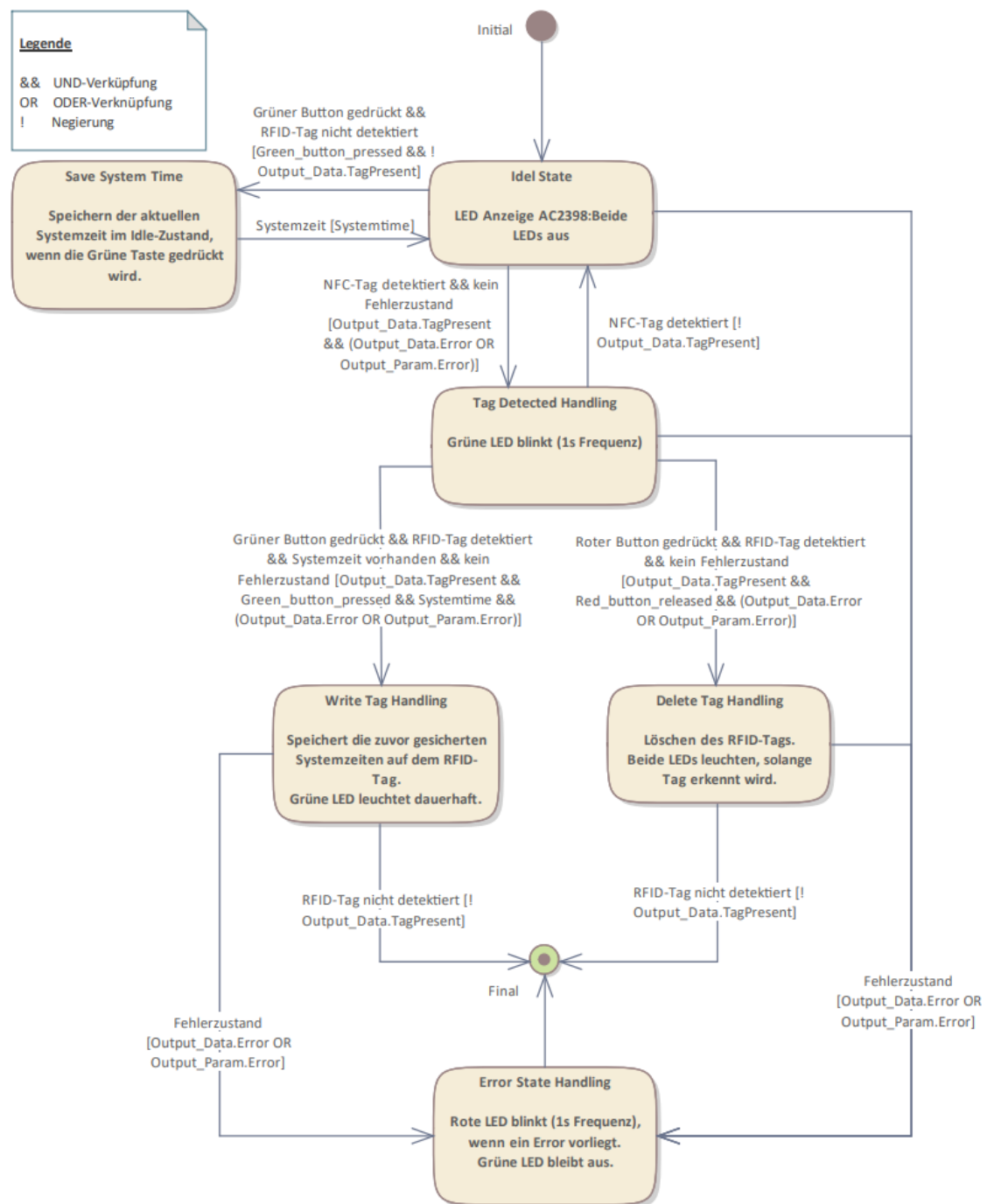


Abbildung 1: State-Machine-Diagramm

Name	Datentyp	Adresse	Kommentar
RED_button_released	Bool	%I193.2	1 wenn roter Taster nicht gedrückt
Green_button_pressed	Bool	%I193.3	1 wenn grüner Taster gedrückt
Red_button_LED_ON	Bool	%Q192.0	wenn 1 dann rote LED vom Taster an
Green_button_LED_ON	Bool	%Q192.1	wenn 1 grüne LED vom Taster an

Tabelle 1: Variablentabelle von AC2398 (Tasterblock)

Name	Datentyp	Adresse	Kommentar
Output_Param.Done	Bool	%I6.0	
Output_Param.Busy	Bool	%I6.1	
Output_Param.Error	Bool	%I6.2	
Output_Param.Status	Word	%IW8	
Output_Param.ExtStatus	DWord	%ID10	
Output_Param.RdValue	UInt	%IW14	
Output_Data.TagPresent	Bool	%I92.0	
Output_Data.Done	Bool	%I92.1	
Output_Data.Busy	Bool	%I92.2	
Output_Data.Error	Bool	%I92.3	
Output_Data.Status	Word	%IW94	
Output_Data.ExStatus	Word	%IW96	
Input_Param.Execute	Bool	%Q0.0	
Input_Param.Mode	UInt	%QW2	
Input_Param.SetValue	UInt	%QW4	
Input_Data.DT_InAddr	UInt	%QW16	
Input_Data.DT_OutAddr	UInt	%QW18	
Input_Data.Execute	Bool	%Q20.0	
Input_Data.Force	Bool	%Q20.1	
Input_Data.Mode	UInt	%QW22	
Input_Data.TagMemAddr	UInt	%QW24	
Input_Data.Length	UInt	%QW26	
Input_Data.WrData	Array[0..31] of Byte	%Q28.0	
Input_Data.RdData	Array[0..31] of Byte	%Q60.0	

Tabelle 2: Variablentabelle von DTI515 (NFC-Modul)

Umsetzung

Im Folgenden soll die Umsetzung der State Machine als SPS-Prpogramm beschrieben werden. Dabei werden die einzelnen Programmbausteine hierarchisch und beginnend beim Organisationsbaustein (OB1) kurz erläutert.

PLC Main

Im Organisationsbaustein befinden sich in Summe drei Programmbausteine. Dabei handelt es sich um die Bausteine "DTI515 parametrization", "DTI515 Read/Write Data" und "DTI515 RFID Manager". Die ersten beiden Bausteine müssen mit Werten versorgt werden, um die RFID-Antenne zu betreiben. Der Block DTI515 parametrization bietet drei Eingänge, die für die Umsetzung relevant sind. Die Eingänge heißen "Mode", SSetValue und Execute". Mit dem Eingang Mode kann festgelegt werden, ob gelesen oder geschrieben werden soll. Mit dem Eingang SetValue wird im Schreibfall der Wert, welcher geschrieben wird, übergeben. Mit Execute wird das letztendliche Schreiben oder Lesen getriggert. DTI515 parametrization bietet außerdem insgesamt 6 Ausgänge. Relevant für das Projekt ist nur der Ausgang ,Error', welcher anzeigt, dass ein Fehler aufgetreten ist. Alle Eingänge und Ausgänge sind mit PLC-Variablen verknüpft. Der Block DTI515 Read/Write Data hat sieben Eingänge. Vier dieser Eingänge sind für die Umsetzung des Projekts relevant. Diese werden durch PLC-Variablen versorgt, welche vom RFID-Manger gesetzt werden. Es handelt sich um die Eingänge ,Execute', ,Mode', ,Length' und ,WrData'. Mit Mode wird gewählt, ob gelesen oder geschrieben wird. Über Length

wird festgelegt, wie viele Byte im Schreibfall geschrieben werden sollen. WrData ist der Eingang, der mit den zu versendenden Daten versorgt wird. Execute triggert das letztendliche Schreiben von Daten auf den RFID-Tag. Von den sechs Ausgängen dieses Bausteins sind 2 relevant. Zum einen der Ausgang ‚TagPresent‘. Zum anderen der Ausgang ‚Error‘. Mittels TagPresent kann ermittelt werden, ob sich ein RFID-Tag in der Nähe der Antenne befindet. Error wird überprüft, um festzustellen, ob das Schreiben fehlerfrei verlaufen ist. Der Wert beider wird in einer PLC-Variable abgelegt (siehe Abb. 1, 2, 3, 5, 4).

RFID-Manager

Netzwerk 1: Error_State_Handling

Dieses Netzwerk hat den Zweck der Bestimmung des Wertes zweier Merker und einer lokalen Variable. Dies geschieht auf Grundlage der von DTI515 parametrization und DTI515 Read/Write Data erzeugten Error Outputs. Diese werden an den DTI515 RFID Manager übergeben. Haben beide Errors den Wert null, wird der Merker ‚M1_Error_State Handling‘ mittels SR-Glied rückgesetzt. Andernfalls ist dieser Merker dauerhaft eins. Dessen Ausgang entspricht dem Set Eingang eines SR-Glieds, welches den Merker ‚M2_Error_State_Handling_Clock‘ setzt. Falls M1_Error_State Handling null ist, sind auch der Merker M2_Error_State_Handling_Clock und die lokale Variable ‚Red_Led_Error_State‘ null. Ist dieser eins, wechseln Merker und Variable in einer Frequenz von 1Hz zwischen High und Low (siehe Abb. 6).

Netzwerk 2: Tag_Detected_Handling

Zweck des zweiten Netzwerks ist die Bestimmung der Werte für die Merker ‚M3_Tag_Detected_Handling‘ und ‚M4_Tag_Detected_Handling_Clock‘. Außerdem werden zwei Ausgänge von DTI515 RFID Manager gesetzt, falls ein Tag erkannt wird. Zusätzlich wird die lokale Variable ‚Green_Led_Tag_Detected_Handling‘ aktualisiert. Falls kein Fehler in Netzwerk 1 detektiert wurde und zusätzlich der Block DTI515 Read/Write Data einen Tag registriert, wird M3_Tag_Detected_Handling

mittels RS-Glied gesetzt. Dessen Ausgang triggert zwei Move-Blöcke, welche die Ausgänge ‚Data_Length‘ und ‚Data_Mode‘ aktualisieren. Data_Length wird auf den Wert 32 Bits (4 Byte) gesetzt. Data_Mode bekommt den Wert 5, was ‚Schreiben‘ entspricht. Diese Ausgänge konfigurieren die RFID-Antenne. Der Setz-Eingang des SR-Glieds von M4_Tag_Detected_Handling wird über die Move-Blöcke ebenfalls mit dem Ausgang des SR-Gliedes von M3_Tag_Detected_Handling verbunden. Wird dieser Eingang auf eins gesetzt, wechselt der Merker M4_Tag_Detected_Handling_Clock und die mit dem Ausgang des zugehörigen Flipflops verbundene Variable Green_Led_Tag_Detected mit einer Frequenz von 1Hz zwischen High und Low (siehe Abb. 7).

Netzwerk 3: Save_System_Time

Es soll möglich sein die aktuelle Systemzeit auf den NFC-Tag zu schreiben. Zu diesem Zweck muss es möglich sein diese zu generieren. Dies geschieht in Netzwerk 3. Gesetzt wird der Merker ‚M5_Save_System_Time‘ und eine lokale Variable. Dabei handelt es sich um die Variable ‚Systemtimeconv‘. Falls der Merker M3_Tag_detected_Handling in Netzwerk 2 gesetzt wurde und zusätzlich ‚Green_button‘ den Wert eins hat, wird mittels eines ‚Time_TCK‘ Blocks die aktuelle Systemzeit ermittelt und in der lokalen Variable ‚Systemtime‘ abgespeichert. Der Wert in Systemtime muss anschließend noch in einen Integer gewandelt werden. Das geschieht im Anschluss an den Time_TCK Block im Block ‚T_CONV‘, welcher Systemtimeconv als Ausgang hat. Abschließend wird M5_Save_System_Time mittels eines SR-Glieds einmalig gesetzt. Ein Rücksetzen dessen ist nicht vorgesehen. Es hat den Zweck sicherzustellen, dass mindestens einmal die aktuelle Systemzeit ermittelt und in Systemtimeconv abgelegt wurde (siehe Abb. 8).

Netzwerk 4 und 5: Write_Tag_Handling und Delete_Tag_Handling

Die beiden Netzwerke sind für das letztendliche Beschreiben und Löschen des NFC-Tags vorgesehen. Deren Aufbau ist identisch, weshalb diese hier zusammengefasst werden. Sie unterscheiden sich lediglich darin, von welche Variablen und Inputs sie gesteuert werden und welche Variablen und Merker sie definieren. In beiden Fällen wird zunächst mal ein Merker definiert. In Netzwerk 4 ist das ‚M6_Write_Tag_Handling‘. In Netzwerk 5 handelt es sich um ‚M7_Delete_Tag_Handling‘. Beide sind mit einem SR-Glied verbunden. Beim Schreiben wird M6_Write_Tag_Handling

gesetzt, falls der Eingangsparameter `Green_Button`, der Merker `M3_Tag_Detected_Handling` aus Netzwerk 2 und der Merker `M5_Save_System_Time` aus Netzwerk 3 den Wert `True` haben. Rückgesetzt wird, falls Merker `M3_Tag_Detected_Handling` aus Netzwerk 2 den Wert `False` hat oder ein Fehler in Netzwerk 1 detektiert und in `M1_Error_State_Handling` abgelegt wurde. Beim Löschen wird `M7_Delete_Tag_Handling` gesetzt, falls der Eingang `Red_button` und der Merker `M3_Tag_Detected_Handling` aus Netzwerk 2 wahr sind. Rückgesetzt wird dieser genauso wie `M6_Write_Tag_Handling` beim Schreiben. Der Ausgang der beiden Merker dient im jeweiligen Netzwerk der Definition eines Ausgangs von DTI515 RFID Manager. Es handelt sich um den Ausgang `Data_Write`, welcher ein Array ausgibt. An den Ausgang des zu den Merkern gehörenden Flipflops ist in Netzwerk 4 und 5 eine Kette von vier Move-Blöcken angeschlossen. Diese haben den Zweck das Array von `Data_Write` mit Werten zu befüllen. Beim ersten Block wird Adresse 0 beschrieben, beim zweiten Adresse 1, beim dritten Adresse 2 und beim letzten Adresse 3. In Netzwerk 4 (Schreiben) werden die Move-Blöcke mit der zuvor in der Variable `Systemtimeconv` abgelegten aktuellen Systemzeit befüllt, sodass diese sich anschließen im Array von `Data_Write` befindet. In Netzwerk 5 (Löschen) werden die Move-Blöcke mit nullen versorgt und das Array von `Data_Write` enthält anschließend nur nullen. Die Werte innerhalb des Arrays werden später auf den NFC-Tag geschrieben. Sobald die Befüllung dessen abgeschlossen ist, wird beim Schreiben in Netzwerk 4 die lokale Variable `Execute_Write_Handling` und `Green_LED_Write_Tag_Handling` gesetzt. Beim Löschen in Netzwerk 5 wird die Variable `Execute_Delete_Handling`, `Red_LED_Delete_Tag_Handling` und `Green_LED_Delete_Tag_Handling` gesetzt (siehe Abb. 9, 10).

Netzwerk 6: LED_Control

Netzwerk 6 dient der Steuerung der LED-Anzeigen für die verschiedenen Betriebszustände, basierend auf dem Status der relevanten Merker. Wenn einer der grünen Merker, wie `Green_LED_Tag_Detected_Handling`, `Green_LED_Write_Tag_Handling` oder `Green_LED_Delete_Tag_Handling`, aktiv ist und kein Fehlerzustand vorliegt (`M1_Error_State_Handling` ist null), wird der Merker `Green_Button_LED_State` gesetzt. Dies aktiviert die grüne LED, um anzuzeigen, dass der entsprechende Betriebsmodus erfolgreich erkannt wurde. Für die rote LED gilt ein ähnliches Prinzip.

Wenn der Merker `Red_LED_Delete_Tag_Handling` aktiv ist und kein Fehlerzustand vorliegt (`M1_Error_State_Handling` ist null), wird `Red_Button_LED_State` gesetzt, sobald `Red_LED_Error_State` aktiv ist. Dadurch wird die rote LED eingeschaltet, um einen spezifischen Fehlerzustand anzuzeigen. Dies bietet eine visuelle Rückmeldung im Falle eines Problems (siehe Abb. 11).

Netzwerk 7: Execution_Operations

Netzwerk 7 ermöglicht die Durchführung von Schreib- oder Löschoperationen auf dem RFID-Tag. Wenn entweder der Merker `Execute_Write_Handling` oder `Execute_Delete_Handling` aktiv ist, wird der Ausgang `Execute` gesetzt. Dieser Ausgang triggert die eigentliche Aktion, entweder das Schreiben oder Löschen der Daten auf dem RFID-Tag. Dadurch wird sichergestellt, dass die gewünschte Operation korrekt ausgeführt wird und der Prozess ordnungsgemäß abgeschlossen ist (siehe Abb. 12).

Anhang

Organisationsbaustein "Main"

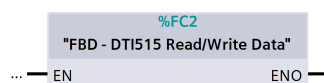


Abbildung 1: Main [OB1] Netzwerk 1: DTI515 Parametrization

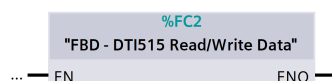


Abbildung 2: Main [OB1] Netzwerk 2: DTI515 Read/Write Data

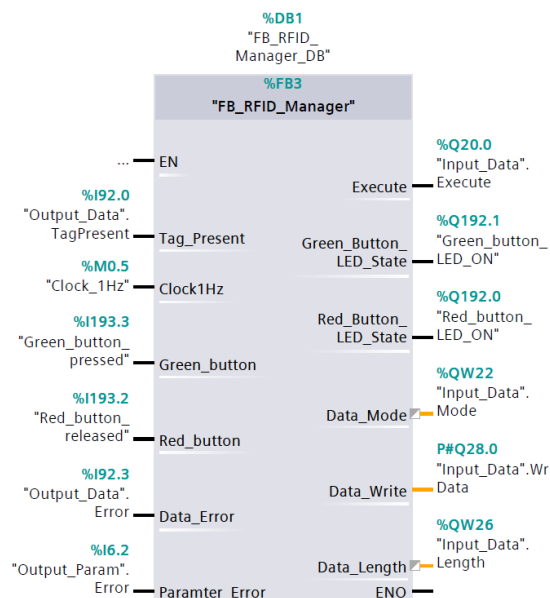


Abbildung 3: Main [OB1] Netzwerk 3: DTI515 RFID Manager

Funktion "FBD_DTI515 parametrization"

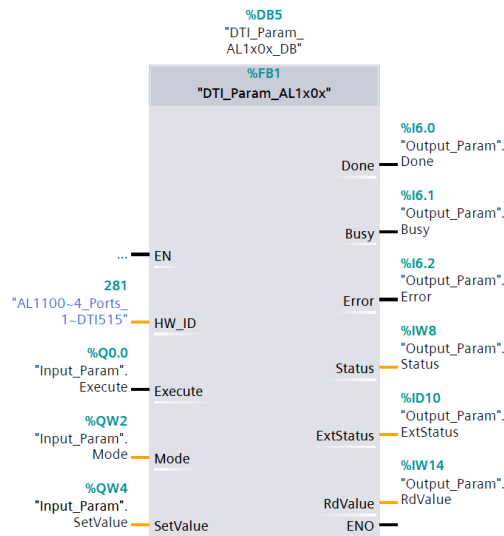


Abbildung 4: FBD - DTI515 parametrization [FC3] Netzwerk 1: Config

Funktion "FBD_DTI515 Read-Write_Data"

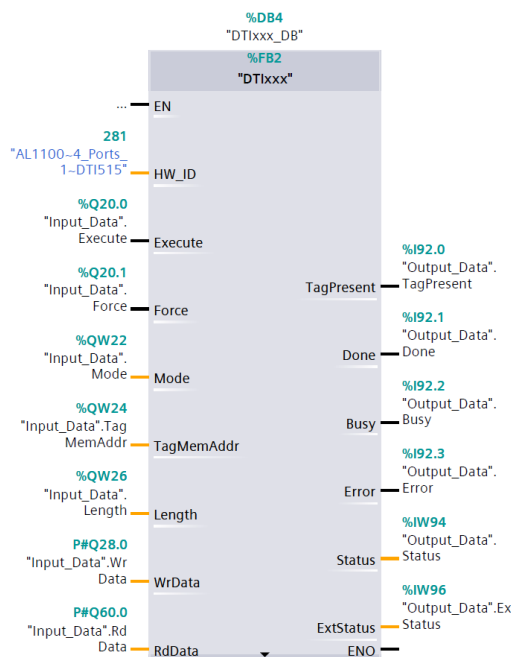


Abbildung 5: FBD - DTI515 Read/Write Data [FC2] Netzwerk 1: Config

Funktionsbaustein "FB_RFID_Manager"

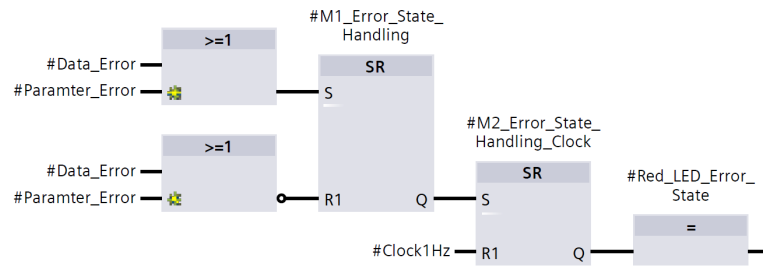


Abbildung 6: FB_RFID_Manager [FB3] Netzwerk 1: Error_State_Handling

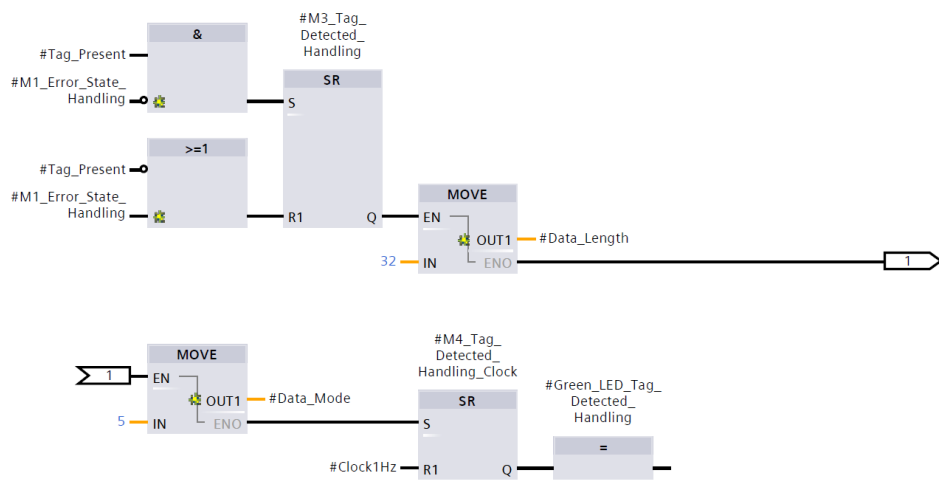


Abbildung 7: FB_RFID_Manager [FB3] Netzwerk 2: Tag_Detected_Handling

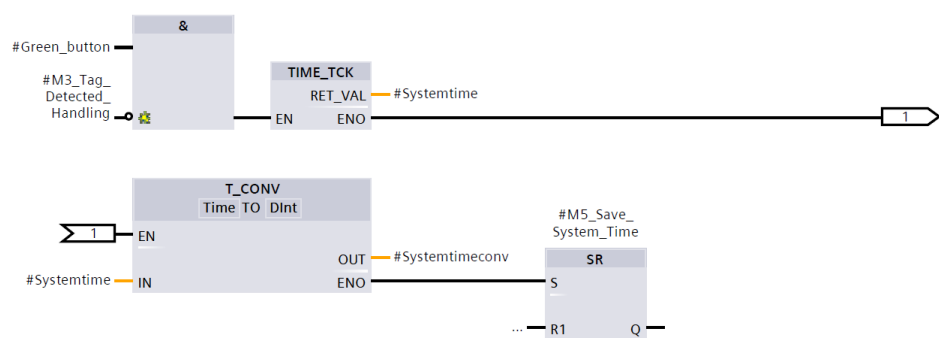


Abbildung 8: FB_RFID_Manager [FB3] Netzwerk 3: Save_System_Time

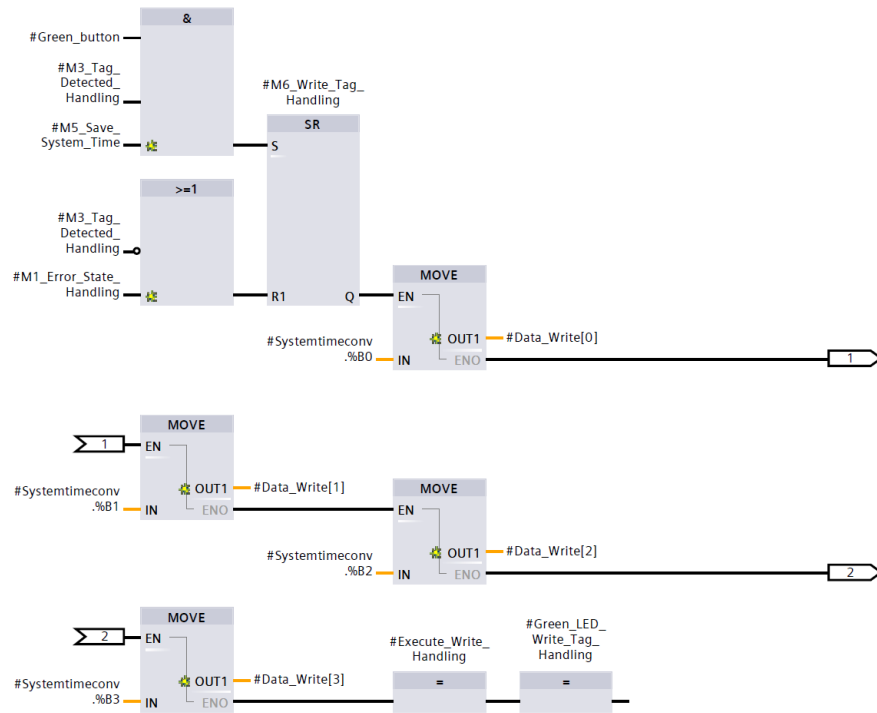


Abbildung 9: FB_RFIID_Manager [FB3] Netzwerk 4: Write_Tag_Handling

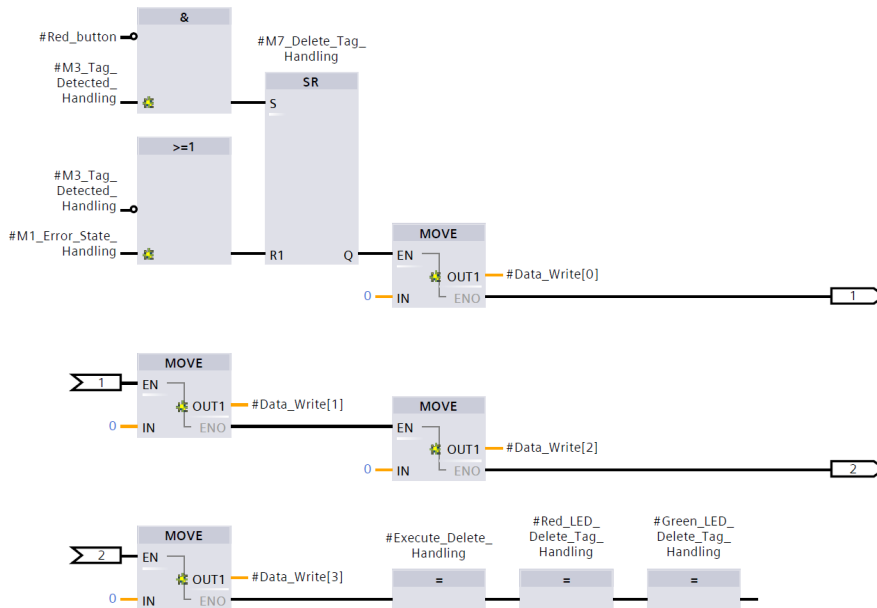


Abbildung 10: FB_RFIID_Manager [FB3] Netzwerk 5: Delete_Tag_Handling

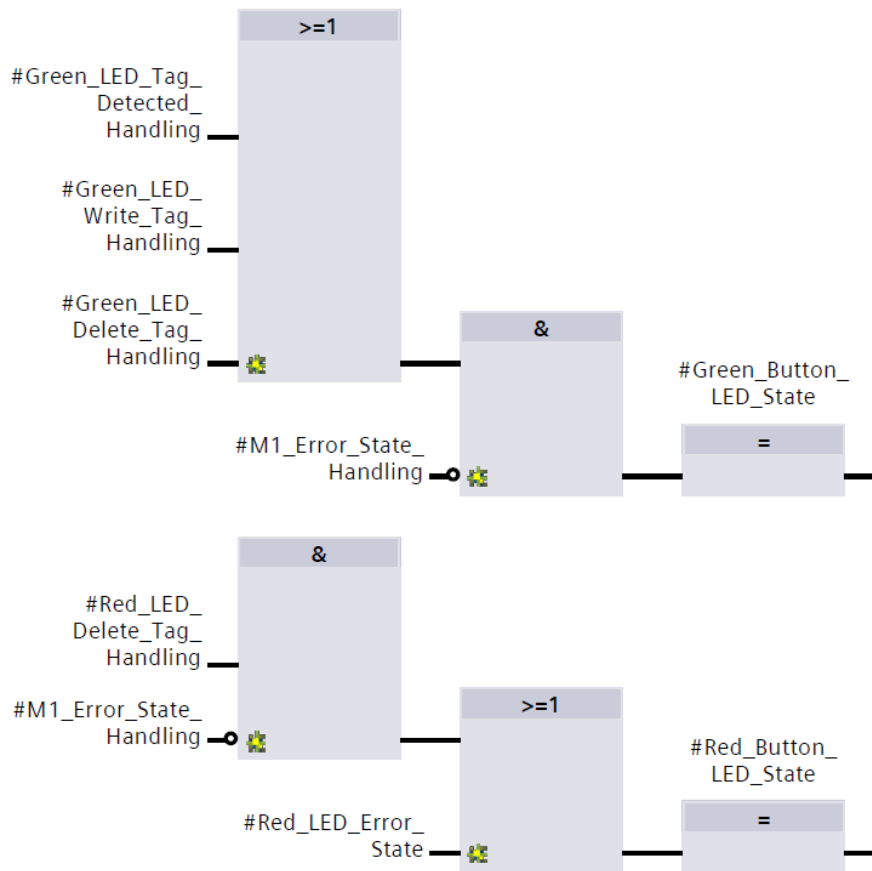


Abbildung 11: FB_RFID_Manager [FB3] Netzwerk 6: LED_Control

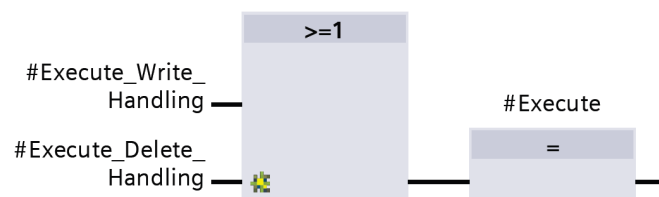


Abbildung 12: FB_RFID_Manager [FB3] Netzwerk 7: Execution_Operations