



Hochschule für Technik,
Wirtschaft und Kultur Leipzig

FAKULTÄT INGENIEURWISSENSCHAFTEN

E209 - FACTORY AUTOMATION

Riesenrad Praktikumsanlage

Vorgelegt von: Philipp Nöcker
Studiengang: Elektrotechnik und Informationstechnik
Seminargruppe: 24EIM-AT
Matrikelnummer: 85980
Emailadresse: philipp.noecker@stud.htwk-leipzig.de

Inhaltsverzeichnis

Abbildungsverzeichnis	ii
1 Einleitung	1
2 Inbetriebnahme der Praktikumsanlage	2
2.1 Hardwaremodule	2
2.2 Sicherheitskonfiguration	3
2.3 Handbetrieb	4
2.4 Automatikbetrieb	5
2.5 Visualisierung	7
3 Tests und Verifikation	10
4 Fazit und Ausblick	12

Abbildungsverzeichnis

1	Modellanlage des Riesenrad Jupiter	1
2	Ausschnitt der globalen Variablenliste mit Variablen für die Hardwareverknüpfung als Prozessvariablen	2
3	Verwendung von globalen Variablen als Prozessvariablen am Beispiel des Powerlink-Moduls „X20DO4321“ zur Motorensteuerung	2
4	Zustandsautomat für die Sicherheitslogik	3
5	Programmablaufplan der grundlegenden Handbetriebsfunktionen	4
6	Zustandsautomat für den Automatikbetrieb	6
7	Navigationsleiste der Visualisierung mit Knöpfen für den Seitenwechsel und Anzeige für Datum und Uhrzeit	7
8	Sicherheitsvisualisierung bei Normalbetrieb	7
9	Sicherheitsvisualisierung bei Schutzstopp	7
10	Visualisierung für die Einstellung der LED, Scheinwerfer und Ampel	8
11	Visualisierung für die Gondeln des Riesenrads mit Gondelfarbe und Personenanzahl	8
12	Hauptbedienfeld bei Handbetrieb mit Sicherheitsmeldung wegen Not-Taster	9
13	Hauptbedienfeld bei Automatikbetrieb in Setup-Phase mit Hinweis, dass Gondeln bisher nicht leer sind und Automatik somit noch nicht gestartet werden kann	9
14	vollständige Visualisierung mit Handbetriebsseite	9

1 Einleitung

Im Mastermodul Factory Automation des Studienganges Elektro- und Informationstechnik wurde als praxisorientierte Prüfungsleistung ein Modell eines Riesenrads in Betrieb genommen. Dafür wurde die Hardware, eine Aufgabenstellung mit detaillierter Hardwarebeschreibung, sowie eine Abschlussarbeit, in welcher die Modellanlage entwickelt wurde, bereitgestellt. Der Aufbau besteht aus dem Riesenradmodell, s. Abb. 1, einem *SITOP power 5* Netzteil von Siemens, einem Leitstand, mit integriertem Display und vier Tastern, einem ASi-Netzteil von *Siemens*, einer Lichtschranke von *Leuze electronic* und vier ASi-Tastern. Zusätzlich steht ASi Sicherheitsmonitor *ASM2E/2* von *Leuze electronics* zur Verfügung, sowie einem Sicherheitsgitter und einem Not-Halt-Taster. Zur Steuerung des Riesenrades wurde ein Mikrocontroller bereitgestellt, welcher Daten über ein Powerlink Modul empfängt und somit den Motor des Riesenrads, die Scheinwerfer, LED-Leisten und die Ampel steuert.



Abbildung 1: Modellanlage des Riesenrad Jupiter

Die Aufgabe sollte sein, eine Sicherheitsfunktion und -logik umzusetzen, wodurch das Auslösen des Not-Tasters oder das Durchschreiten des Sicherheitsgitters die Anlage in einen gefahrenlosen Zustand überführt werden sollte. Zudem sollte ein Handbetrieb und ein Automatikbetrieb entwickelt werden. Auch eine Visualisierung sollte erstellt werden. Für die Steuerungslogik sollte ein Projekt in *B&R Automation Studio* erstellt werden. Die Sicherheitslogik sollte mit der Software *Asimon* erstellt werden.

Das vorliegende Dokument dokumentiert die entwickelten Programmierlösungen zur Erfüllung der gegebenen Anforderungen. Es werden Programmausschnitte und schematische Programmabläufe bereitgestellt. Das vollständige Automation Studio Projekt, sowie die .AS3 Datei kann aus dem nachfolgenden **Repository** entnommen werden. Beide Inhalte sind gleichzeitig lokal auf dem Entwicklungscomputer im Raum WI-113 in der Wächterstraße abgelegt.

2 Inbetriebnahme der Praktikumsanlage

In folgenden Unterkapiteln wird die Aufgabenlösung präsentiert. Zunächst wird auf die Einbindung der Hardware und die Realisierung der Sicherheitslogik eingegangen. Im Anschluss wird der Handbetrieb vorgestellt und auf die Visualisierung eingegangen. Abschließend wird dann der Automatikbetrieb vorgestellt. Es ist anzumerken, dass die Entwicklung der Lösung nicht klar in diesen Schritten erfolgte und währenddessen stets verschiedenen Verifikationstests und anschließende Anpassungen durchgeführt wurden. Diese Dokumentation präsentiert also nur die endgültige Fassung, welche als Abgabe bereitgestellt wurde.

2.1 Hardwaremodule

In einem ersten Schritt wurden innerhalb der *Automation Studio* Software die Hardwaremodule im innerhalb der *Configuration View* gemäß Praktikumsanleitung hinzugefügt und miteinander in der Software verbunden. Anschließend wurden für die Signalverarbeitung globale Variablen innerhalb der *logical View* erzeugt, s. Abb. 2. Gemäß der bereitgestellten Coding Guidelines wurden die globalen Variablen mit dem Präfix „g“ gekennzeichnet. Diese wurden dann mit den entsprechenden I/O-Ports der Module über die *Physical View* als Prozessvariablen verknüpft, s. Abb. 3. Die Verknüpfung erfolgte dabei gemäß der bereitgestellten Aufgabenstellung. Dabei wurde auch nach der Verknüpfung die Veränderung der Werte der Prozessvariablen untersucht, um ein verbessertes Verständnis der Werte und Zustände zu erhalten.

RFID			Kanalname	Prozessvariable	Datentyp
■♦ gRFIDColor	UINT		→● ModuleOk		BOOL
Operating Light			→● SerialNumber		UDINT
■♦ gRedTop	BOOL		→● ModuleID		UINT
■♦ gGreenTop	BOOL		→● HardwareVariant		UINT
■♦ gRedDown	BOOL		→● FirmwareVersion		UINT
■♦ gGreenDown	BOOL		● DigitalOutput01	::gDriveDirection	BOOL
Drive			● DigitalOutput02	::gDriveSpeed1	BOOL
■♦ gDriveDirection	BOOL		● DigitalOutput03	::gDriveSpeed2	BOOL
■♦ gDriveSpeed1	BOOL		● DigitalOutput04		BOOL
■♦ gDriveSpeed2	BOOL		● StatusDigitalOutput01		BOOL

Abbildung 2: Ausschnitt der globalen Variablenliste mit Variablen für die Hardwareverknüpfung als Prozessvariablen

Abbildung 3: Verwendung von globalen Variablen als Prozessvariablen am Beispiel des Powerlink-Moduls „X20DO4321“ zur Motorensteuerung

Es ist anzumerken, dass die lokalen Variablen auch je nach Art einen Präfix erhalten haben. Eine Übersicht wird in Tab. 1 gezeigt.

Präfix	Datentyp der Variable
g	globale Variable (beliebiger Typ)
b	Bool
n	Integer
s	String
a	Array
e	Enumerator
r	Detektor steigende Flanke
rf	Detektor steigende und fallende Flanke
ton	Einschaltverzögerung

Tabelle 1: Aufschlüsselung der Variablenpräfix

2.2 Sicherheitskonfiguration

Die Erstellung der Sicherheitskonfiguration erfolgte gemäß der bereitgestellten Aufgabenstellung. Dabei wurde die Sicherheitskonfiguration erstellt und auf den Sicherheitsmonitor geladen. Als passwort für die Konfiguration „RiesneradSafety_Noecker“ wurde wie gefordert „student“ verwendet. Des Weiteren wurde Sicherheitslogik innerhalb des *Automation Studio* Projektes entwickelt. Ziel der Sicherheitslogik war es, dass wenn das Lichtgitter oder der Not-Aus Taster eine Meldung geben, dass das Riesenrad gestoppt wird. Erst nach einer Beseitigung des Fehlers und dem Drücken des Quittieren Tasters sollte es wieder möglich sein, das Riesenrad zu steuern. Zudem wurde eine weitere Sicherheitslogik implementiert. Diese sollte bei einem Wechsel des Betriebszustandes auch eine Quittierung fordern. Diese Implementierung wurde als sinnvoll erachtet, da so das Riesenrad nach einem Moduswechsel zunächst in Ruhe startet und zudem der Bediener zu einer bewussten Entscheidung gezwungen wird durch anschließende Bestätigung durch die Quittierung.

Zusammenfassend kann die entwickelte Sicherheitslogik durch folgenden Zustandsautomaten repräsentiert werden, s. Abb. 4.

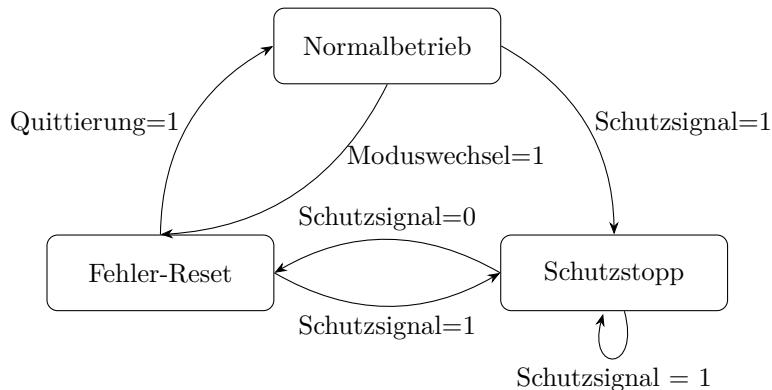


Abbildung 4: Zustandsautomat für die Sicherheitslogik

Das Schema dient dabei zur vereinfachten Darstellung. Der Zustand „Normalbetrieb“ bedeutet, dass Handbetrieb oder Automatikbetrieb ausgeführt wird. Falls das Sicherheitsgitter durchschritten wird oder der Not-Taster betätigt wird (damit wird im Schema „Schutzsignal = 1“ symbolisiert), springt der Automat in den Zustand „Schutzstopp“. Solange wie solch ein Schutzsignal anliegt verharrrt der Zustandsautomat auch im Schutzstopp Modus. Sobald kein Schutzsignal mehr anliegt, indem das Hindernis aus dem Sicherheitsgitter entfernt wurde, oder der Not-Taster deaktiviert wurde, wird in den Zustand „Fehler-Reset“ gewechselt. In diesem wird solange verharrrt und der Motor des Riesenrades blockiert, bis der Quittieren-Taster betätigt wurde. Wenn der Modus gewechselt wird, soll, wie beschrieben, auch aus dem Normalbetrieb in den Fehler-Reset Zustand gegangen werden. So wird sichergestellt, dass das Riesenrad bei einem Wechsel keine Bewegungen mehr ausführt und der Bediener den Betriebswechsel aktiv Bestätigen muss.

Die Logik des Zustandsautomaten wurde anschließend innerhalb des *Automation Studio* Projektes implementiert, wobei zusätzliche Funktionen hinzugefügt wurden. Die Implementierung erfolgte durch Hinzufügen einer Aktion, welche innerhalb des *Cyclic.st*-Programms zyklisch aufgerufen wird. Es wurde dabei implementiert, dass wenn der Zustand „Fehler-Reset“ aktiv ist, die Quittieren-Taste Blinkt mit einer Periodendauer von 1 s. Dies wurde durch die Verwendung einer Einschaltverzögerung mittels *TON()*-Funktion realisiert. Das Umschalten des aktuellen Betriebszustandes zwischen Automatik und Handbetrieb wird durch einen *RF_TRIG()* detektiert, da dieser von boolschen Variablen steigende und fallende Flanken detektiert. Nachdem der Benutzer Reset-Zustand durch Quittieren beendet, befindet sich die Anlage im Handbetrieb. So soll sichergestellt sein, dass keine automatische Bewegung abläuft. Zudem wurde implementiert, dass wenn die Anlage im Schutzstopp- oder Reset-Zustand sich befindet, dass die Betriebsleuchten auch Rot geschaltet werden und somit eine Blockierung anzeigen. Nach einer Quittierung durch den Bediener, werden diese wieder auf Grün umgeschaltet.

2.3 Handbetrieb

In einem nächsten Schritt wurden die Funktionen des Handbetriebs implementiert. Dabei sollte der Bediener das Riesenrad mit verschiedenen Geschwindigkeiten steuern können und dies zum einen über die Taster des Leitstandes und über die Visualisierung. Zunächst wurde die Funktion zur Einstellung der Motorengeschwindigkeit innerhalb einer Aktion „Drive.st“ implementiert. Diese Aktion wird im Hauptprogramm stets zyklisch aufgerufen. Innerhalb der „Drive.st“ Aktion wird durch eine **Case**-Verzweigung mit dem Enumerator **eSpeed** je nach Zustand die digitalen Ausgänge für die Motorensteuerung gesetzt. Es lassen sich somit die vier Geschwindigkeiten, stopp, langsam, normal und schnell einstellen.

Zur interaktiven Bedienung wurde sich zunächst auf die Taster konzentriert. Es wurde dafür die Aktion „Manual.st“ geschrieben, welche nur aufgerufen wird, wenn die Variable „bAutoMode“ den Wert „False“ besitzt und gleichzeitig, die Safety weder im Schutzstop, noch im Reset-Zustand sich befindet. Für die manuelle Geschwindigkeitssteuerung wurden die boolschen Prozessvariablen der Taster innerhalb einer **IF-ELSIF**-Verzweigung mit Werten für den Enumerator **eSpeed** gebunden, sodass beispielsweise bei der Betätigung des Schnell-Tasters, das Riesenrad mit der Geschwindigkeitseinstellung „schnell“ solange dreht, bis der Benutzer auf die Stopp-Taste drückt. Die **IF-ELSIF**-Verzweigung wurde anschließend um Variablen für die Benutzeroberfläche erweitert mit **OR** Verknüpfungen. So konnte innerhalb der Benutzeroberfläche durch „MomentaryPushButton“ das gleiche Verhalten, wie mit den physischen Tastern des Leitstandes erreicht werden. Zusätzlich wurde innerhalb der Visualisierung ein Pushbutton für die Normalgeschwindigkeit hinzugefügt.

Zudem wurde innerhalb der Visualisierung ein „ToggleButton“ hinzugefügt, welcher einen „Push-Betrieb“ An-und Ausschalten kann. Der „Push-Betrieb“ bedeutet, dass sich das Riesenrad nur solange dreht, wie aktiv ein Taster gedrückt wird. Dies kann vor allem für den Benutzer von Vorteil sein, wenn bestimmte Ausrichtungen des Riesenrades, manuell erzielt werden sollen. Im Push-Betrieb ist die Besonderheit, dass auch die Normalgeschwindigkeit über die physischen Taster des Leitstandes erreicht werden kann. Dafür müssen die Taster für Schnell und Langsam gleichzeitig betätigt werden. Die beschriebenen Funktionen für den Handbetrieb werden in Abb. 5 schematisch als Programmablaufplan dargestellt. Die Variablennamen sind dabei an den Variablen des *Auto-*

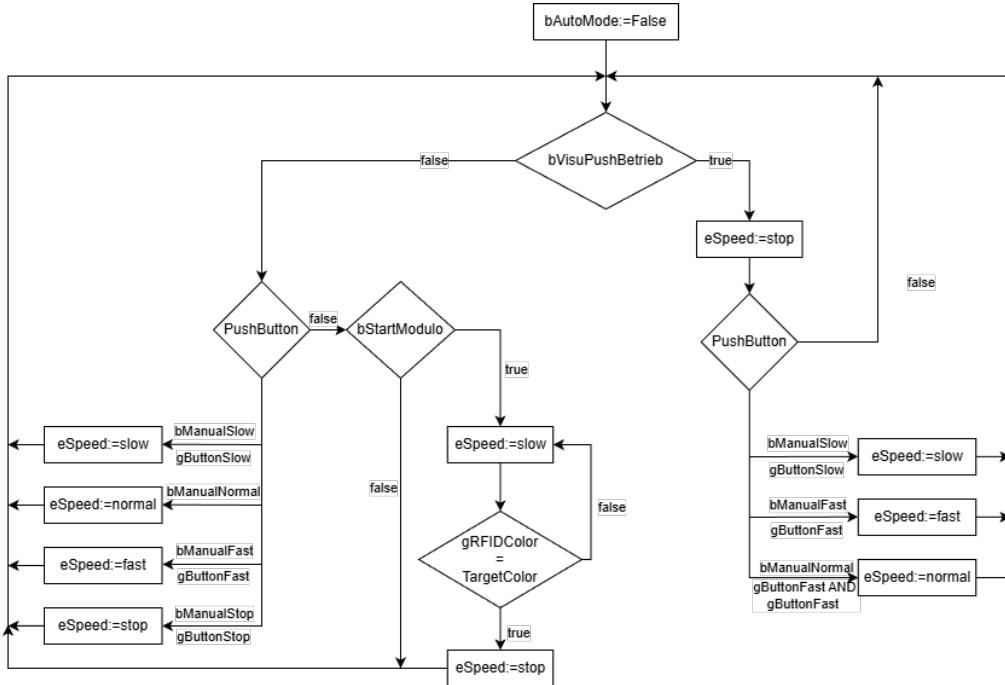


Abbildung 5: Programmablaufplan der grundlegenden Handbetriebsfunktionen

mation Studio Projektes orientiert. Als *PushButton* werden die Taster des Leitstandes und aus der Visualisierung bezeichnet. Zur verbesserten Übersichtlichkeit wurden die Abfragen nach den Taster-Aktivierungen vereinfacht abgebildet. Dabei gilt, dass eine Verbindungsleitung zu einer Geschwindigkeitseinstellung durch die jeweiligen Bedingungen für die Aktivierung begleitet wird. Die Bedingungen unmittelbar ober- oder unterhalb eines Pfeils können als Disjunktionen betrachtet werden. Es wurden im Verlauf zusätzliche Funktionen für den Handbetrieb implementiert, welche in Abb. 5 nicht abgebildet werden. Diese werden dennoch folgend beschrieben.

Für die Einstellung der Drehrichtung wurde in der Visualisierung ein „ToggleSwitch“ implementiert. Dieser ist mit der globalen Variable „gDriveDirection“ verknüpft, welche als Prozessvariable über die Powerlink-Verbindung die Drehrichtung bestimmt. Die Änderung der Drehrichtung über die physischen Taster des Leitstandes wurde durch Verwendung eines **TON()**-Funktionsblocks ermöglicht. Dafür wurde eine Verzögerungszeit von 5 s eingestellt, sodass der Zustand der boolschen Variable „gDriveDirection“ nach fünfsekündiger Betätigung des Stopp-Tasters invertiert wird.

Eine weitere Funktionalität wurde auf der graphischen Benutzeroberfläche noch für den Handbetrieb implementiert. Diese wird folgend als Modulo-Funktion aufgeführt. Dabei kann der Benutzer aus einer „Drop-Down-Liste“ eine Gondelfarbe auswählen und durch die Betätigung eines „Start“-Tasters fährt dann das Riesenrad, bis die entsprechende Gondelfarbe am RFID-Leser registriert wird und somit die Gondelfarbe beim Ein- und Ausstieg positioniert ist.

Eine weitere „automatische“ Funktionalität für den Handbetrieb wurde für die Ampelfarben des Riesenrads entwickelt. Dabei wurde sichergestellt, dass sobald sich das Riesenrad bewegt, die Ampel auf Rot geschaltet wird, sodass keine möglichen Unfälle geschehen könnten. Falls das Riesenrad keine Bewegung ausführt, kann die Ampelfarbe über Buttons innerhalb der Visualisierung gesteuert werden.

Anforderung an den Handbetrieb war auch, dass der aktuelle Betriebsmodus auf dem Display angezeigt wird. Dafür wurde auch eine Aktion geschrieben, welche zyklisch durch das Hauptprogramm aufgerufen wird.

2.4 Automatikbetrieb

Nach dem Handbetrieb wurde anschließend der Automatikbetrieb implementiert. Im Automatikbetrieb sollte die Anzahl der Fahrgäste pro Gondel, sowie der Umläufe des Riesenrads sinnvoll eingestellt werden. Zudem sollten über das Display und die Visualisierung die Betriebszustände angezeigt werden und Fehlerfälle durch die Betriebsleuchte und die Visualisierung deutlich werden. Zudem sollen Scheinwerfer und LED-Leisten steuerbar sein.

Dafür wurde zunächst eine Abfolge von Zuständen für den Automatikbetrieb festgelegt. Einen vereinfachten Zustandsautomaten wird in Abb. 6 dargestellt. Es beginnt mit einer Einrichtung. Dabei wird die Anzahl der Drehungen, sowie die Anzahl der Personen pro Gondel festgelegt. Zudem kann die Drehrichtung über den Einrichten Taster festgelegt werden. Das Betätigen des Start-Tasters führt in zur Initialisierung, wobei das Riesenrad solange sich dreht, bis die Rote Gondelgruppe am Eingang und somit am RFID-Leser sich befindet.

Dann folgt die Einsteigen-Sequenz. Hierbei wird der Zustand des Empfängers der Lichtschranke kontinuierlich abgefragt und steigende Flanken führen zu einem Erhöhen der Zählvariable. Die Werte für die Personenzählung der Gondelgruppen wurden dafür in Arrays angelegt. Es wurde sich bewusst für die Detektion der steigenden Flanke entschieden, da die Prozessvariable **gReceive**, also der Wert des Empfängers der Lichtschranke bei einem Hindernis in der Lichtschranke den Wert **False** ausgibt. Somit werden die Zählvariablen erst erhöht, wenn das Objekt aus der Lichtschranke wieder entfernt ist, also die Person durchgelaufen ist. Wenn nun die Zählvariable so groß wie die

in der Einrichtungsphase gesetzten Anzahl an Personen pro Gondelgruppe übereinstimmt, dann fährt das Riesenrad weiter, bis eine anderer Wert des RFID-Lesegerätes ausgegeben wird, als null und als der der letzten Gondelgruppe. Falls also der Wert einer „neuen Gondelgruppe“ vorliegt, hält das Riesenrad an und der Einstiegszustand wird wieder erreicht. Im Einstiegen-Zustand wird zusätzlich überprüft, ob eine Kombination aus „gDriveDirection“ und „gRFIDColor“ vorliegt. Dies würde die letzte Gondel der Einstiegsequenz bedeuten, je nach Drehrichtung des Motors. Falls dies Eintritt wird das Riesenrad in den Zustand „Umlauf“ versetzt.

Dabei läuft das Riesenrad mit einer kontinuierlich schnellen Geschwindigkeit und überprüft stets den Wert der „gRFIDColor“. Dabei wird überprüft, ob die Rote Gondelgruppe mit dem Wert 125 für die Prozessvariable „gRFIDColor“ anliegt. Genauer gesagt, wird das Anliegen des Wertes für eine Flankenerkennung genutzt, sodass pro Umdrehung nur einmalig eine Zählvariable inkrementiert wird. Wenn die in der Einrichtungsphase eingestellte Umdrehungsanzahl erreicht wurde, ist der Aussteigen-Zustand erreicht. In diesem wird analog zur Einstiegssequenz der Wert des Lichtschrankenempfängers mit einer Flankenerkennung überprüft und bei Detektion die Zählvariable um einen Wert verkleinert. Wenn die Gondelgruppe den Wert 0 für die Personenzählung besitzt, dreht sich das Riesenrad wieder bis zur nächsten Gondelgruppe. Auch analog zur Einstiegssequenz, wird beim Aussteigen auf bestimmte Kombinationen aus RFID-Wert und Motorrichtung geachtet, welcher dann nach dem „Entleeren“ der Gondel einen Zustandswechsel zum Zustand „Beendet“ führen würde.

In diesem Wird solange abschließend verharrt, bis der Start-Taster betätigt wird. Der Taster leuchtet dabei kontinuierlich. Das Betätigung würde den Zustandsautomaten wieder in die Settupphase führen, sodass der Prozess erneut durchgeführt werden kann. Um das zu symbolisieren, blinkt in der Settupphase der Start-Taster mit einer Periodendauer von 2 s. Bei der Implementierung wurde auch darauf geachtet, dass die Betätigung des Start-Tasters über eine Flankenerkennung registriert wird, da sonst ein für den Benutzer zu schneller Zustandswechsel von Beendet, über Einrichtung zu Initialisierung, da mit einem Tastendruck mehrere SPS-Zyklen durchlaufen werden würden.

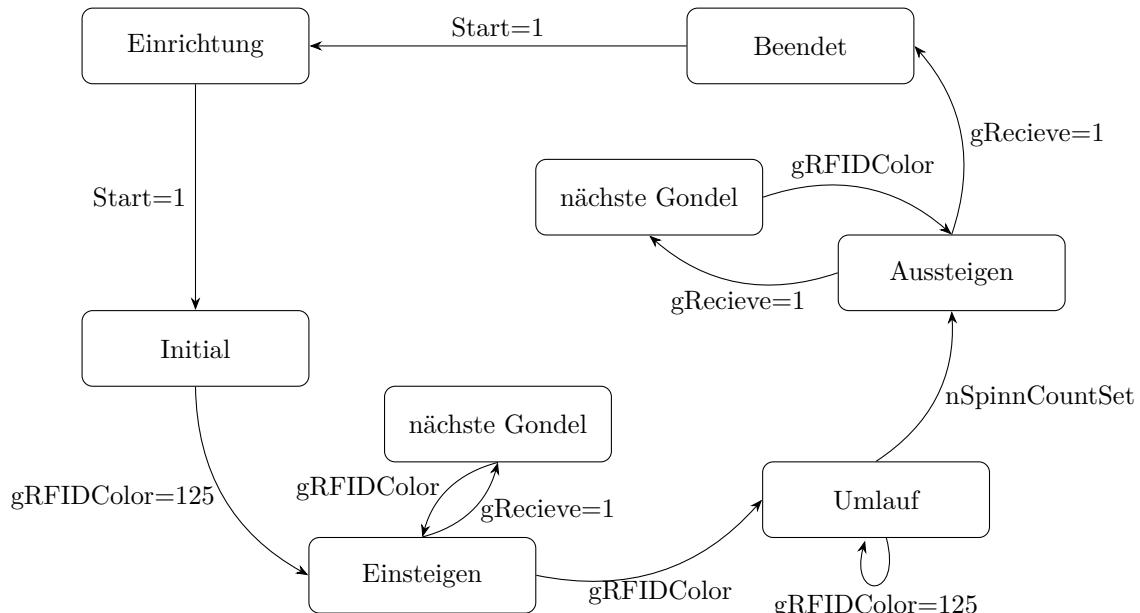


Abbildung 6: Zustandsautomat für den Automatikbetrieb

Während des Automatikbetriebes kann der Benutzer, wie gefordert, die Lichter einstellen. Zudem wird der aktuelle Betriebszustand über das Display und innerhalb der Visualisierung ausgegeben. Auf dem Display wird dabei zwischen den Anzeigen „Ein-/ Aussteigen“, „Umlaf“, „Stop“ und leeren Anzeigen unterschieden. In der Visualisierung werden die Zustände verständlich in Text ausgegeben.

2.5 Visualisierung

In den vorherigen Kapiteln wurden bereits Inhalte und Funktionen der Visualisierung vorweggenommen. Dennoch wird in diesem Kapitel detaillierter die entwickelte graphische Benutzeroberfläche beschrieben. Die Visualisierung kann in drei Bereiche unterteilt werden, einer Navigationsleiste, einer Statusleiste und einem Hauptbereich. Die drei Bereiche sind farblich voneinander getrennt. Die Navigationsleiste besitzt zwei Knöpfe, um zwischen der Automatik-Seite und der Handbetriebs-Seite wechseln zu können. Zudem wird in der Navigationsleiste das aktuelle Datum und die Uhrzeit angezeigt, s. Abb. 7.



Abbildung 7: Navigationsleiste der Visualisierung mit Knöpfen für den Seitenwechsel und Anzeige für Datum und Uhrzeit

Die Statusleiste birgt verschiedene Funktionen und Anzeigen. Im oberen Bereich ist der Sicherheitsbereich, dieser hat als Hintergrundfarbe die Farbe gelb erhalten. Es werden drei Lampen angezeigt, welche den Zustand der Sicherheit symbolisieren: „Schutzstopp“, „Notauftaster“ und „Lichtgitter“. Wenn alle 3 Lampen grün leuchten, ist das System im Normalbetrieb, s. Abb. 8. Falls ein Sicherheitssignal aktiv ist, also der Nottaster gedrückt wurde oder sich ein Hindernis im Sicherheitsgitter befindet, dann leuchten die die Lampe für den Schutzstopp und das entsprechende Sicherheitsereignis Rot, s. Abb. 9. Liegt kein Sicherheitsereignis mehr vor, dann leuchtet nur noch die Lampe „Schutzstopp“ Rot, welche ihre Farbe dann mit Betätigung des Quittieren-Tasters wieder auf Grün wechselt.



Abbildung 8: Sicherheitsvisualisierung bei Normalbetrieb



Abbildung 9: Sicherheitsvisualisierung bei Schutzstopp

Unterhalb des Sicherheitsbereichs folgt dann der Bereich, in welchem die Lichter des Riesenrads eingestellt werden können, s. Abb. 10. Es gibt einem Schieberegler, mit welchem die 16 Zustände der LED-Leisten des Riesenrads manuell verstellt werden können. Neben dem Schieberegler ist zudem eine Zahlenausgabe, welche den aktuellen Lichter-Modus ausgibt. Unter dem Schieberegler befinden sich auch zwei „ToggleButton“ mit welchen sich die beiden Scheinwerfer des Riesenrads An- und Ausschalten lassen. Die Scheinwerfer und die LED-Leisten des Riesenrads können in jedem Modus und auch im Schutzstopp noch bedient werden. Dabei ist der Sinn, dass in einem Fehlerfall möglicherweise der Bediener zusätzliches Licht benötigt, um Probleme auf dem Riesenrad festzustellen, oder um die festliche Beleuchtung bei einem Unfall auch Ausschalten zu können. Neben den Buttons für die Scheinwerfer gibt es auch zwei „MomentaryPushButton“, mit welchen sich die Ampelfarben einstellen lassen. Dabei haben die Knöpfe jedoch die Eigenschaft, dass sie nur bedienbar werden, wenn der Handbetrieb aktiv ist. Im Handbetrieb kann dann auch nur die Farbe geändert werden, wenn das Riesenrad gerade nicht in Bewegung ist. Schließlich wäre es gefährlich wenn das Riesenrad in Bewegung ist und den Gästen symbolisiert werden würde, dass sie Einstigen können.

Unterhalb des Bereichs für die Lichter gibt es dann noch eine Anzeige, welche die aktuelle Gondelgruppe anzeigt. Dabei wird die Farbe der aktuellen Gondelgruppe dargestellt. Zudem wird die aktuelle Personenanzahl, welche sich in der Gondel befindet neben der Gondelfarbe angezeigt,

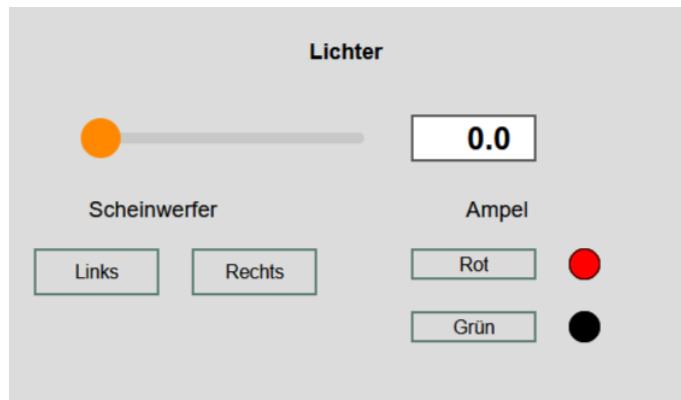


Abbildung 10: Visualisierung für die Einstellung der LED, Scheinwerfer und Ampel

s. Abb. 2. Im Handbetrieb gibt es dann zusätzlich noch einen weiteren Button bei der Gondelgruppe. mit dem Knopf kann dann der Benutzer bestätigen, dass keine Gäste mehr in einer Gondel sind. Dies wurde als sinnvolle rachtet, da nach einem Fehlerfall und dem anschließenden Quittieren, das Riesenrad zunächst im Handbetrieb genutzt werden kann. Um dann die Personen aus den Gondeln zu holen, müsste also der Bediener zunächst die Gondelgruppe zum Ausstieg fahren und kann dann für die Software bestätigen, dass nun die Gondelgruppe leer ist.

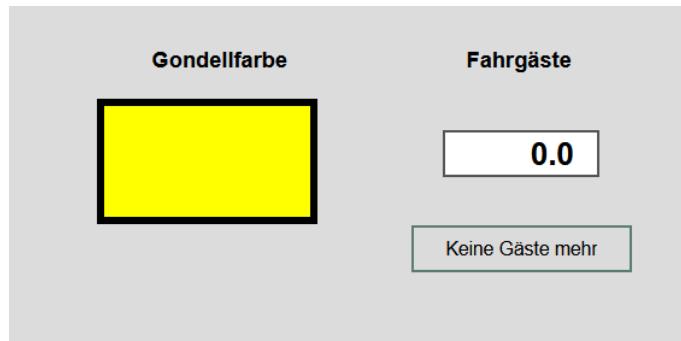


Abbildung 11: Visualisierung für die Gondeln des Riesenrads mit Gondelfarbe und Personenanzahl

Den größten Bereich der Visualisierung nimmt der Hauptbereich ein. Hier wird je nach Seitenauswahl die Seite für den Handbetrieb oder die Seite für den Automatikbetrieb angezeigt. Die Startseite ist stets die Seite des Handbetriebs. Beide Seiten haben jedoch auch gleiche Elemente. So haben beide Seiten zwei Textausgaben, welche als Displays betrachtet werden sollen. Darüber wird jeweils der Aktuelle Betriebsmodus, also Handbetrieb oder Automatik, ausgegeben. Zudem wird über die zweite Textausgabe der Zustand konkretisiert, indem die Geschwindigkeit, oder der Zustand des Automatikbetriebs angezeigt wird. Bei Störungen wird zudem die Meldung „Störung“ ausgegeben. Zusätzlich werden bei beiden Hauptseiten bei Fehlermeldungen entsprechende Warnmeldungen angezeigt. Dabei wird bei einem Schutzsignal eine Rote Meldung mit dem jeweiligen Sicherheitsereignis gezeigt. Wenn keine Störung mehr vorliegt, aber noch keine Quittierung erfolgte, wird dies zusätzlich durch eine orange Meldung visualisiert. Des Weiteren gibt es auch eine Warnung, wenn der Bediener den Handbetriebsmodus an hat, jedoch Personen noch in den Gondeln sich befinden. Dies führt zu einer Warnung, sodass das Riesenrad nicht abgeschaltet wird, solange noch Personen sich auf diesem befinden.

Der Handbetrieb hat dann verschiedene Knöpfe, welche bei der Beschreibung des Handbetriebs schon erläutert wurden. Es gibt PushButton für die Einstellung der Geschwindigkeit, sowie einen Button um zwischen Handbetrieb und Automatik zu wechseln. Zudem gibt es einen Drop-Down-Liste aus welcher eine Farbe ausgewählt werden kann und durch die Bestätigung des Start-Knopfes fährt das Riesenrad dann bis diese Gondelfarbe am Einstieg sich befindet. Außerdem gibt es einen „ToggleSwitch“ mit welchem sich die Drehrichtung einstellen lässt, s. Abb. 12.

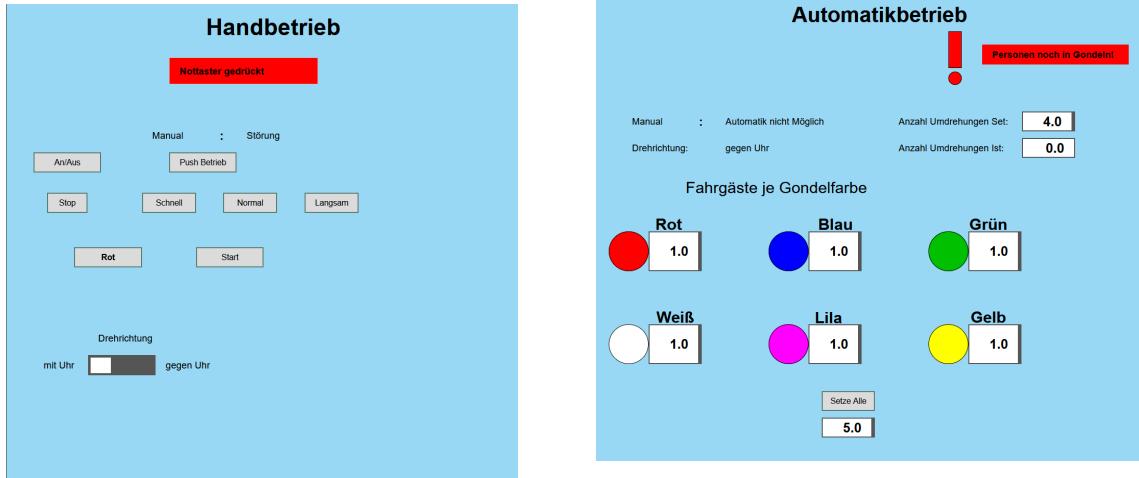


Abbildung 12: Hauptbedienfeld bei Handbetrieb mit Sicherheitsmeldung wegen Not-Taster

Abbildung 13: Hauptbedienfeld bei Automatikbetrieb in Setup-Phase mit Hinweis, dass Gondeln bisher nicht leer sind und Automatik somit noch nicht gestartet werden kann

Die Seite für den Automatikbetrieb hat die gleiche Hintergrundfarbe, wie die des Handbetriebs. Bei der Automatikseite gibt es einige Funktionen, wenn sich das Riesenrad im Setup-Zustand des Automatikbetriebs befindet, s. Abb. 13. Dann kann der Bediener selbst einstellen, wie viele Umdrehungen das Riesenrad im Betrieb machen soll und zudem kann der Bediener einstellen wie viele Personen in die Gondeln einsteigen können. Dabei kann der Bediener über die Bedienoberfläche einen Wert für alle Gondeln festlegen oder auch jede Gondel individuell anpassen. In der Setupphase wird zudem auch angezeigt in welche Richtung sich das Riesenrad drehen wird. Die Richtung kann dabei über den Einrichten-Taster eingestellt werden. Wenn der Betrieb gestartet wurde, dann werden die Felder für die Zahleneingabe ersetzt durch Felder für die Zahlausgabe, sodass keine Wertmanipulation mehr stattfinden kann. Zudem wird auch für jede Gondel angezeigt wie viele Personen in der Gondel sind und wie Maximal hineindürfen.

Die vollständige Visualisierung mit allen Subelementen kann in Abb. 14 eingesehen werden.

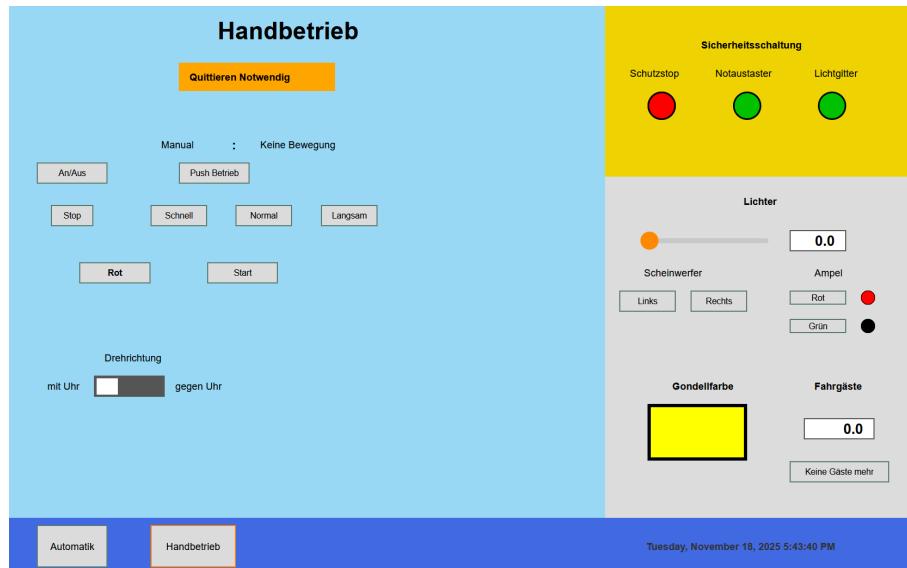


Abbildung 14: vollständige Visualisierung mit Handbetriebsseite

3 Tests und Verifikation

Um die verschiedenen Funktionalitäten zu überprüfen wurden bereits während der Entwicklung verschiedene Tests stets durchgeführt, um das entwickelte Projekt zu Testen und wenn möglich auch zu Optimieren. Verschiedene Testszenarien wurde in Tab. 2 zusammengetragen. Dabei wurde der Testrahmen und der Beobachtete Ausgang notiert.

Testrahmen	Ausgang und Beobachtungen
Warmstart SPS	Quittierung erforderlich, nach Quittierung im Handbetrieb. Notwendige Quittierung wird durch blinkenden Quittieren-Taster dargestellt
Geschwindigkeitseinstellung im Handbetrieb	verschiedene Geschwindigkeiten nach einmaliger Betätigung von Tastern am Leitstand oder der Buttons in der Visu
Push-Betrieb im Handmodus	Solang wie Knopf gedrückt Drehung des Riesenrads. Bei gleichzeitigem Drücken von Tastern „Langsam“ und „Schnell“ dreht sich das Rad mit Geschwindigkeit Normal. Auch Knöpfe der Visualisierung erzielen gewünschte Geschwindigkeiten
Änderung der Drehrichtung durch STop-Taster	Durch Betätigung von 5 s des Schalters konnte Drehrichtung geändert werden. Test erfolgte als das Riesenrad still stand
Anzeige der aktuellen Gondelfarbe in der Visualisierung	Beim Automatischen Drehen des Riesenrads haben sich die Farben stets gewechselt. Dabei wurde jedoch ein Zahlendreher bei den RFID Farben festgestellt. Die rote und lila Gondel hatten die richtigen Werte bei den RFID-Tags. Bei den anderen Gondeln gab es jedoch Probleme mit den angegebenen Werten in der Dokumentation. Weiß hat den Wert 1536, Blau 512, Gelb 1024 und Grün 256. Nach Korrektion im Code stimmt nun Gondelfarbe in der Visu mit der Gondel beim Ein- und Ausstieg überein.
Anpassung der Lichter	Bei Veränderung des Schiebereglers wurden die verschiedenen LED-Muster am Riesenrad angezeigt. Der Wert welcher über die Visualisierung ausgegeben wird, ändert sich auch entsprechend des Schiebereglers.
Scheinwerfer	Betätigung der Button für die Scheinwerfer ermöglicht korrekte Benutzung
Ampel	Wenn das Riesenrad keine Bewegung ausführt, kann manuell das Grüne Licht angeschaltet werden und auch das rote Licht verändert werden. Wenn das Riesenrad in Betrieb ist, ist dies nicht mehr möglich und die Ampel leuchtet durchgehend Rot
Modulo Modus	Auswahl über Drop-Down Menü möglich. Wenn der Pushbetrieb ausgeschaltet ist, funktioniert der Modus korrekt. Bei Pushbetrieb wird nur eine kurze Impulsartige Bewegung ausgeführt, da das Riesenrad nach einem Takt direkt wieder gestoppt wird.
Sicherheitsereignis Not-Taster	Bei laufendem Betrieb Nottaster betätigt. Riesenrad stoppt direkt. Quittieren Knopf leuchtet auf und blinkt. Betriebsleuchte schaltet auf Rot um. Solange wie Not-Taster weiterhin gedrückt wird, keine Quittierung möglich. Wenn Not-Taster wieder auf Ausgangszustand, dann Quittierung möglich.
Sicherheitsereignis Lichtgitter	Bei laufendem Betrieb Hindernis in Sicherheitsgitter. Riesenrad stoppt sofort und Quittieren Taster blinkt. Solange wie Objekt weiterhin in Hindernis keine Sicherheitsfreigabe. Betriebsleuchte auch Rot, bis Quittierung erfolgreich durchgeführt.
Umschalten in Automatikbetrieb	Änderung von Anzeige in der Visualisierung. Quittierung notwendig für weiteren Betrieb.
Einstellen von Werten im Setup-Modus	Durch Widgets in der Visualisierung können die entsprechenden Personenanzahlen oder Umdrehungen festgelegt werden. Das Betätigen des Einrichten-Tasters führt auch zu einer Änderung der angesetzten Drehrichtung

Ablauf Automatikmodus	Automatikmodus läuft korrekt ab und kann nach erfolgreicher Beendigung erneut gestartet werden. Dabei leuchtet zunächst der Start-Taster dauerhaft, was die Beendigung symbolisiert und anschließend nach der Betätigung blinkt der Taster, was die Setup-Phase symbolisiert.
Sicherheitsereignis während Automatik	Stoppen des Betriebs. Nach Beseitigung und Quittierung in Handbetrieb, sodass wenn noch Personen in Gondel vorhanden sind, diese über den manuellen Betrieb aus den Gondeln geholt werden können. Erst wenn Gondeln alle leer sind, kann im Automatikbetrieb wieder der Ablauf gestartet werden.

Tabelle 2: Übersicht über verschiedene Tasszenarien und deren Ausgang

Es ist anzumerken, dass mit diesen Tests die Grundfunktionalität und die Realisierung der Anforderungen weitestgehend getestet wurden. Dabei wurden weitere Fälle getestet, welche zur Übersichtlichkeit nicht in dieser Tabelle aufgenommen wurden. Die Funktionalität wird auch in einer direkten Vorführung gezeigt, bei welcher versucht wird die beschriebenen Testszenarien erneut vorzuführen.

4 Fazit und Ausblick

Innerhalb der Belegleistung wurde ein Steuerungsprojekt geschrieben, mit welchem die Riesenradanlage mit ihren Komponenten gesteuert werden kann. Es wurden dafür entsprechende Variablen mit Eingangs- und Ausgangsdaten verknüpft, Zustandslogik in Form eines Programms in Strukturiertem Text geschrieben und eine Visualisierung entwickelt. Diese wurden auf verschiedenen Wegen und unter veränderten Bedingungen getestet, wobei das geforderte Verhalten stets erreicht werden konnte. Es gibt einen funktionierenden Handbetrieb, welcher über die Aufgabenstellung hinaus an Funktionalität implementiert bekommen hat und einen Automatikbetrieb, welcher einen definierten Ablauf von Zuständen durchläuft und dabei einen vollständigen Ablauf durchführen kann. Es wurden gleichzeitig Maßnahmen für die Sicherheit integriert, sodass bei einem Sicherheitsfall ein gefahren-loser Zustand erreicht wird und durch Quittierung beendet werden kann. Die erstellte Software wurde durch Verwendung von Aktionen modular aufgebaut, sodass diese durch Hinzufügen von weiteren Hardwaremodulen programmierfreundlich erweitert werden kann. Zudem wurde bei der Entwicklung drauf geachtet, dass die Variablennamen stets englisch sind und einen Präfix zur Datentypklassifizierung besitzen, sodass mit diesen Kenntnissen andere Programmierer schneller im Code zurechtfinden und diesen nachvollziehen und erweitern können. Es wurde eine übersichtliche und klar strukturierte Bedienoberfläche entwickelt, welche die wichtigen Zustände der Anlage anzeigt und eine intuitive Bedienung zulässt.

Dem Entwickler sind stets noch Erweiterungspotentiale für die Anlage in den Sinn gekommen. So wäre innerhalb der Benutzeroberfläche eine Benutzerverwaltung sinnvoll. So könnten beispielsweise die Handfunktionen nur Power-Usern zur Verfügung stehen. Zusätzlich sollte an ein Alarm-Logging gedacht werden. Bei diesem werden die Fehlerfälle automatisch protokolliert. Dafür könnte eine neue Hauptbedienfläche hinzugefügt werden können. Für eine noch einfachere Bedienoberfläche könnte auch ein Touch-Display für die Visualisierung in Frage kommen. Dafür wurden bei den Zahleneingaben innerhalb der Visualisierung schon die Tastaturen mit aktiviert, sodass eine Umstrukturierung auf Touch zu keinen größeren Umstellungen führen sollte. Optimierungspotential bietet aber auch der Automatikbetrieb. Bei diesem wird aktuell jede Gondel nacheinander angefahren. Bei einer Modellanlage mag dies zu keinen Problemen führen. Bei einem richtigen Riesenrad, müsste jedoch bei der Einstiegs- und Ausstiegssequenz auf eine gleichzeitige Personenverteilung geachtet werden.