

Testspezifikation Hardware- und Umgebungskomponenten

Für das studentische Projekt *Sichere Eisenbahnsteuerung*

Datum	24.06.2010
Quelle	Google Code → Dokumente → 04_Test → 04.01_Testspezifikation
Autoren	Norman Nieß Kai Dziembala
Version	1.0
Status	freigegeben

1 Historie

Version	Datum	Autor	Bemerkung
0.0	03.06.2010	Kai Dziembala Norman Nieß	Initialisierung der Testspezifikation für die Hardware- und System-Komponenten
0.1	09.06.2010	Kai Dziembala Norman Nieß	Vorbereitung Testfall 5 und 6, Layoutanpassung, Behebung von Rechtschreibfehlern
0.2	10.06.2010	Kai Dziembala Norman Nieß	Arbeit an Testfall 5 und 6
0.3	16.06.2010	Kai Dziembala Norman Nieß	Fertigstellung Testfall 5 und 6
0.4	17.06.2010	Kai Dziembala Norman Nieß	Einfügen der ausformulierten Haupttestziele in Kapitel 3
1.0	24.06.2010	Kai Dziembala Norman Nieß	Korrektur von Rechtschreibfehlern, Ergänzungen zur Verwendung des Logikanalysators in Kapitel 9.4, Korrektur von Referenzangaben auf Testskripte

2 Inhaltsverzeichnis

1 Historie.....	2
2 Inhaltsverzeichnis.....	3
3 Testziele.....	5
4 Testfall 1 „Schienennetz, Sensor- und Magnetpositionen“.....	6
4.1 Identifikation der Testobjekte.....	6
4.2 Check-Liste.....	6
5 Testfall 2 „Manuelle Verstellbarkeit der Weichen und Entkoppler“.....	7
5.1 Identifikation der Testobjekte.....	7
5.2 Check-Liste.....	7
6 Testfall 3 „Ferngesteuerte Verstellbarkeit der Weichen und Entkoppler“.....	8
6.1 Identifikation der Testobjekte.....	8
6.2 Check-Liste.....	8
7 Testfall 4 „Ferngesteuertes Fahren der Lokomotiven“.....	9
7.1 Identifikation der Testobjekte.....	9
7.2 Check-Liste.....	9
8 Testfall 5 „Mikrocontroller gesteuerter Fahrbefehl “.....	10
8.1 Identifikation der Testobjekte.....	10
8.2 Test-Identifikation.....	10
8.3 Testfallbeschreibung.....	10
8.4 Testskript.....	10
8.5 Testreferenz.....	11
8.6 Test-Protokoll.....	11
9 Testfall 6 „Test des S88-Rückmeldemoduls und der Hall-Sensoren“.....	12
9.1 Identifikation der Testobjekte.....	12
9.2 Test-Identifikation.....	12
9.3 Testfallbeschreibung.....	12
9.4 Testskript.....	13
9.5 Testreferenz.....	14
9.6 Test-Protokoll.....	14

10 Testfall 7 „Test der Not-Aus-Relais“.....	15
10.1 Identifikation der Testobjekte.....	15
10.2 Check-Liste.....	15
11 Testfall 8 „Position der Lokomotiven und Wagons“.....	16
11.1 Identifikation der Testobjekte.....	16
11.2 Check-Liste.....	16
12 Auswertung.....	17

3 Testziele

Der Test der Hardware- und Systemumgebungs-Komponenten dient der Erfüllung des ersten Haupttestziels, sowie dem Hardware-Anteil des zweiten Haupttestziels, welche aus dem zehnten Kapitel des Testplans stammen.

Diese Haupttestziele lauten wie folgt:

- 1) Die Systemumgebung erfüllt die im Pflichtenheft spezifizierten Bedingungen (Kapitel 4.2).
- 2) Das Gesamtsystem erfüllt die Fahraufgabe gemäß der Vorgabe im Pflichtenheft (Kapitel 6).

4 Testfall 1 „Schienennetz, Sensor- und Magnetpositionen“

4.1 Identifikation der Testobjekte

Es werden folgende Komponenten getestet:

- Schienennetz
- Sensoren
- Lokomotiven, Wagons, Magnete

4.2 Check-Liste

Was	Prüfhäufigkeit	Zustand	Bemerkung bei 'niO'
Schienennetz laut Abbildung 3 im Kapitel 5.1 des Pflichtenhefts aufgebaut	1 mal	<input type="checkbox"/> iO <input type="checkbox"/> niO	
Position der Sensoren laut Abbildung 3 im Kapitel 5.1 des Pflichtenhefts	1 mal	<input type="checkbox"/> iO <input type="checkbox"/> niO	
Ausstattung der Lokomotiven und Wagons mit Magneten entsprechend dem Kapitel 5.2.1 und 5.2.2 des Pflichtenhefts	1 mal	<input type="checkbox"/> iO <input type="checkbox"/> niO	

5 Testfall 2 „Manuelle Verstellbarkeit der Weichen und Entkoppler“

5.1 Identifikation der Testobjekte

Es werden folgende Komponenten getestet:

- Weichen
- Entkoppler

5.2 Check-Liste

Was	Prüfhäufigkeit	Zustand	Bemerkung bei 'niO'
Weiche 'a' lässt sich manuell hin und her verstellen	2 mal	<input type="checkbox"/> iO <input type="checkbox"/> niO	
Weiche 'b' lässt sich manuell hin und her verstellen	2 mal	<input type="checkbox"/> iO <input type="checkbox"/> niO	
Weiche 'c' lässt sich manuell hin und her verstellen	2 mal	<input type="checkbox"/> iO <input type="checkbox"/> niO	
Entkoppler 'E1' lässt sich manuell verstellen	1 mal	<input type="checkbox"/> iO <input type="checkbox"/> niO	
Entkoppler 'E2' lässt sich manuell verstellen	1 mal	<input type="checkbox"/> iO <input type="checkbox"/> niO	

6 Testfall 3 „Ferngesteuerte Verstellbarkeit der Weichen und Entkoppler“

6.1 Identifikation der Testobjekte

Es werden folgende Komponenten getestet:

- Weichen
- Entkoppler
- Multi-Maus (nicht alle Funktionalitäten)
- DCC-Verstärker + Transformator

6.2 Check-Liste

Was	Prüfhäufigkeit	Zustand	Bemerkung bei 'niO'
Weiche 'a' lässt sich mittels Multi-Maus verstellen	4 mal	<input type="checkbox"/> iO <input type="checkbox"/> niO	
Weiche 'b' lässt sich mittels Multi-Maus verstellen	4 mal	<input type="checkbox"/> iO <input type="checkbox"/> niO	
Weiche 'c' lässt sich mittels Multi-Maus verstellen	4 mal	<input type="checkbox"/> iO <input type="checkbox"/> niO	
Entkoppler 'E1' lässt sich mittels Multi-Maus verstellen	2 mal	<input type="checkbox"/> iO <input type="checkbox"/> niO	
Entkoppler 'E2' lässt sich mittels Multi-Maus verstellen	2 mal	<input type="checkbox"/> iO <input type="checkbox"/> niO	

7 Testfall 4 „Ferngesteuertes Fahren der Lokomotiven“

7.1 Identifikation der Testobjekte

Es werden folgende Komponenten getestet:

- Lokomotive 1 + 2
- Multi-Maus (nicht alle Funktionalitäten)
- DCC-Verstärker + Transformator

7.2 Check-Liste

Was	Prüfhäufigkeit	Zustand	Bemerkung bei 'niO'
Fährt Lokomotive 1 vorwärts	Gesamtes Streckennetz 1 mal abfahren(auch Abstell- und Nebengleis)	<input type="checkbox"/> iO <input type="checkbox"/> niO	
Fährt Lokomotive 1 rückwärts	Gesamtes Streckennetz 1 mal abfahren(auch Abstell- und Nebengleis)	<input type="checkbox"/> iO <input type="checkbox"/> niO	
Fährt Lokomotive 2 vorwärts	Gesamtes Streckennetz 1 mal abfahren(auch Abstell- und Nebengleis)	<input type="checkbox"/> iO <input type="checkbox"/> niO	
Fährt Lokomotive 2 rückwärts	Gesamtes Streckennetz 1 mal abfahren(auch Abstell- und Nebengleis)	<input type="checkbox"/> iO <input type="checkbox"/> niO	

8 Testfall 5 „Mikrocontroller gesteuerter Fahrbefehl “

8.1 Identifikation der Testobjekte

- Mikrocontroller C515C
- Arduino
- XPressNet-Adapter
- Multi-Maus
- DCC-Verstärker + Transformator
- Lokomotive 1

8.2 Test-Identifikation

Testname: Test_Hardware

Verzeichnisse

Testskripts: Google Code → Dokumente → 04_Tests → 04.02_Testskript →
04.02.11_Hardware+Umgebung

Testprotokolle: Google Code → Dokumente → 04_Tests → 04.03_Testprotokolle →
04.03.11_Hardware+Umgebung

8.3 Testfallbeschreibung

Mit einem Mikrocontroller-Programm wird die Lokomotive 1 dauerhaft auf dem Hauptgleis vorwärts bewegt. Gleichzeitig werden über den Arduino Debug-Ausgaben getätigt. Somit kann sichergestellt werden, dass der Mikrocontroller, die Multimaus, der XpressNet-Adapter, der DCC-Verstärker, die entsprechenden Transformatoren, die Lokomotive 1, der Arduino und die jeweilige Verkabelung funktionstüchtig sind.

Dieser Test muss sowohl mit dem Mikrocontroller eins und zwei durchgeführt werden.

8.4 Testskript

Es wird getestet, ob die Lokomotive 1 mindestens drei Runden auf dem Hauptgleis bewegt werden kann. Dazu werden zuerst die Funktionalitäten und somit die Methoden der Software-Module 'RS232-Treiber' und 'Auditing-System' (entsprechend den jeweiligen Moduldesigns) nachgebildet. Anschließend wird der Shared-Memory-Variable 'EV_RS232_streckenbefehl.weiche' der Wert '0x6' zugewiesen und die Funktion workRS232() aufgerufen. Dann die Variable 'EV_RS232_streckenbefehl.weiche' auf den Wert '0x9' gesetzt und die Funktion workRS232() aufgerufen. Dann die Variable 'EV_RS232_streckenbefehl.weiche' auf den Wert '0xB' gesetzt und die Funktion workRS232() aufgerufen. Die Weichen sind nun so gestellt, dass die Lokomotive das Hauptgleis befährt.

Danach muss die Shared-Memory-Variable 'EV_RS232_streckenbefehl.lok' auf den Wert '0xE' gesetzt und anschließend die Funktion workRS232() aufgerufen werden, um die Lokomotive 1 mit Fahrgeschwindigkeit vorwärts fahren zu lassen.

In einer endlosen while-Schleife wird nun die Funktion 'sendMsg(byte module_id, const byte* msg)' mit den Übergabewerten '0x0' und 'nachricht' aufgerufen. Eine Nachricht ist ein Byte-Array mit folgenden Inhalten: nachricht = [0xFF, 0x0, 0x0, 0xFF, 0xFF, 0x0]. Außerdem wird innerhalb der while-Schleife noch die Funktion 'workAS()' aufgerufen. Danach ist ein Wartezeit zu implementieren, die eine Pause von ca. 5s bewirkt. Somit werden alle 5 Sekunden Nachrichten über den Arduino ausgegeben.

Dies wird mit folgendem Test-Skript realisiert:

siehe 'Google Code → 04_Test → 04.02_Testskripts → 04.02.11_Hardware+Umgebung → Test_Hardware

Nachdem die Lokomotive drei komplette Rundfahrten absolviert hat, kann der Testfall manuell abgebrochen werden.

8.5 Testreferenz

Die Lokomotive 1 fährt mindestens drei Runden auf dem Hauptgleis. Zeitgleich wird auf einem angeschlossenen Rechner im fünf Sekunden-Takt folgende Nachricht ausgegeben:

„Unbekannte Fehlermeldung, Fahrend, Fahrend, Unbekannte Fehlermeldung, Unbekannte Fehlermeldung, Fahrbefehl“

8.6 Test-Protokoll

Was	Prüfhäufigkeit	Ergebnis	Bemerkung bei 'niO'
Mikrocontroller1 gesteuerter Fahrbefehl	1 mal	<input type="checkbox"/> iO <input type="checkbox"/> niO	
Mikrocontroller2 gesteuerter Fahrbefehl	1 mal	<input type="checkbox"/> iO <input type="checkbox"/> niO	

9 Testfall 6 „Test des S88-Rückmeldemoduls und der Hall-Sensoren“

9.1 Identifikation der Testobjekte

Es werden folgende Komponenten getestet:

- Mikrocontroller C515C
- S88-Rückmeldemodul 1 + 2
- Hall-Sensoren
- Multi-Maus
- DCC-Verstärker + Transformator
- Lokomotive 1 + 2
- Wagons
- Magneten

9.2 Test-Identifikation

Testname: Test_Sensoren

Verzeichnisse

Testskripts: Google Code → Dokumente → 04_Tests → 04.02_Testskript →
04.02.11_Hardware+Umgebung

Testprotokolle: Google Code → Dokumente → 04_Tests → 04.03_Testprotokolle →
04.03.11_Hardware+Umgebung

9.3 Testfallbeschreibung

Die beiden Lokomotiven werden mit angekoppelten Wagons nacheinander Multi-Maus-ferngesteuert, jeweils mit langsamer, mittlerer und maximaler Geschwindigkeit über das Streckennetz bewegt. Bei diesem manuellen Fahren muss sichergestellt sein, dass jeder Lokomotiven- und Wagon- Magnet alle Hall-Sensoren mindestens einmal passiert und somit schaltet. Zeitgleich läuft auf dem Mikrocontroller dauerhaft ein Steuerprogramm für das S88-Rückmeldemodul, wodurch die jeweiligen Sensordaten kontinuierlich eingelesen werden. Ein Abgriff dieser jeweils vom S88-Rückmeldemodul gelieferten Sensorzustände erfolgt über einen angeschlossenen Logikanalysator „Agilent Logic Wave“. Dessen Binärmesswerte sind mittels Rechnerkopplung und entsprechender Software einsehbar. Somit kann überprüft werden, ob jeder Magnet bei den drei verschiedenen Fahrgeschwindigkeiten ein Sensorsignal für jeden Hall-Sensor erzeugt und somit die hardwareseitige Sensordatenerfassung funktioniert. Dieser Test muss jeweils für das Rückmeldemodul 1 und 2 durchgeführt werden, wodurch auch beide Sensorsätze auf Funktion überprüft werden.

9.4 Testskript

Die Steuerung der S88-Rückmeldemodule muss entsprechend dem Moduldesign 'S88-Treiber' nachgebildet werden. Dazu müssen die Signale 'PS', 'Reset' und 'Clock' in vorgegebener zeitlicher Abfolge erzeugt werden. Die folgende Tabelle zeigt die dabei zu verwendende Pinbelegung:

Signal	Pin
PS	P5^0
Reset	P5^1
Clock	P5^2

Zur kontinuierlichen Sensordatenerfassung muss die Signalgenerierung wiederholend in einer Endlosschleife ablaufen.

Dies wird mit folgendem Test-Skript realisiert:

siehe 'Google Code → 04_Test → 04.02_Testskripts → 04.02.11_Hardware+Umgebung → Test_Sensoren

An den in obiger Tabelle benannten Pins, sowie an der 'Data'-Signalleitung des S88-Rückmeldemoduls werden nun die Messleitungen von Pad1 des Logikanalysators „Agilent Logic Wave“ angeschlossen.

Messleitung Pad1 Agilent Logic Wave	Signal
0	PS
1	Reset
2	Clock
3	Data

Dieser ist des weiteren mit dem PC zu verbinden und die Analyse-Software 'Agilent LogicWave' mit dem Projekt 'Test_Sensoren_Agilent.lwc' (Google Code → 04_Test → 04.02_Testskripts → 04.02.11_Hardware+Umgebung) zu starten. Ein bereits beschriebenes manuelles Fahren der Lokomotiven incl. Wagons führt nun zu auswertbaren Messsignalen auf dem PC.

9.5 Testreferenz

Jeder Magnet erzeugt für jede der drei verschiedenen Fahrgeschwindigkeiten ein Sensorsignal bei wiederum jedem einzelnen Hall-Sensor.

9.6 Test-Protokoll

Was	Prüfhäufigkeit	Ergebnis	Bemerkung bei 'niO'
Alle Lokomotiven- und Wagon-Magnete erzeugen bei langsamer Fahrgeschwindigkeit Signale am Sensorsatz des Rückmeldemoduls 1	1 mal	<input type="checkbox"/> iO <input type="checkbox"/> niO	
Alle Lokomotiven- und Wagon-Magnete erzeugen bei mittlerer Fahrgeschwindigkeit Signale am Sensorsatz des Rückmeldemoduls 1	1 mal	<input type="checkbox"/> iO <input type="checkbox"/> niO	
Alle Lokomotiven- und Wagon-Magnete erzeugen bei schneller Fahrgeschwindigkeit Signale am Sensorsatz des Rückmeldemoduls 1	1 mal	<input type="checkbox"/> iO <input type="checkbox"/> niO	
Alle Lokomotiven- und Wagon-Magnete erzeugen bei langsamer Fahrgeschwindigkeit Signale am Sensorsatz des Rückmeldemoduls 2	1 mal	<input type="checkbox"/> iO <input type="checkbox"/> niO	
Alle Lokomotiven- und Wagon-Magnete erzeugen bei mittlerer Fahrgeschwindigkeit Signale am Sensorsatz des Rückmeldemoduls 2	1 mal	<input type="checkbox"/> iO <input type="checkbox"/> niO	
Alle Lokomotiven- und Wagon-Magnete erzeugen bei schneller Fahrgeschwindigkeit Signale am Sensorsatz des Rückmeldemoduls 2	1 mal	<input type="checkbox"/> iO <input type="checkbox"/> niO	

10 Testfall 7 „Test der Not-Aus-Relais“

10.1 Identifikation der Testobjekte

Es werden folgende Komponenten getestet:

- Not-Aus-Relais

10.2 Check-Liste

Was	Prüfhäufigkeit	Zustand	Bemerkung bei 'niO'
Not-Aus-Relais 1 schließt bei angelegter Spannung von 5V	3 mal	<input type="checkbox"/> iO <input type="checkbox"/> niO	
Not-Aus-Relais 1 öffnet bei Wegnahme der Spannung	3 mal	<input type="checkbox"/> iO <input type="checkbox"/> niO	
Not-Aus-Relais 2 schließt bei angelegter Spannung von 5V	3 mal	<input type="checkbox"/> iO <input type="checkbox"/> niO	
Not-Aus-Relais 2 öffnet bei Wegnahme der Spannung	3 mal	<input type="checkbox"/> iO <input type="checkbox"/> niO	

11 Testfall 8 „Position der Lokomotiven und Wagons“

Es ist zu beachten, dass dieser Test vor jedem mikrocontrollergesteuertem Fahrbetrieb durchzuführen ist.

11.1 Identifikation der Testobjekte

Es werden folgende Komponenten getestet:

- Lokomotiven
- Wagons

11.2 Check-Liste

Was	Prüfhäufigkeit	Zustand	Bemerkung bei 'niO'
Position Lokomotive 1 ist auf Gleisabschnitt 7, Fahrtrichtung entgegen dem Uhrzeigersinn	1 mal	<input type="checkbox"/> iO <input type="checkbox"/> niO	
Position Lokomotive 2 ist auf Gleisabschnitt 8, Fahrtrichtung entgegen dem Uhrzeigersinn	1 mal	<input type="checkbox"/> iO <input type="checkbox"/> niO	
Personenwagon 1 fest an Lokomotive 1 gekoppelt	1 mal	<input type="checkbox"/> iO <input type="checkbox"/> niO	
Personenwagon 2 fest an Personenwagon 1 gekoppelt	1 mal	<input type="checkbox"/> iO <input type="checkbox"/> niO	
Position Güterwagon 'schwarz' ist auf Gleisabschnitt 2	1 mal	<input type="checkbox"/> iO <input type="checkbox"/> niO	
Position Güterwagon 'rot' ist auf Gleisabschnitt 2 und fest an Güterwagon 'schwarz' gekoppelt	1 mal	<input type="checkbox"/> iO <input type="checkbox"/> niO	
Position Güterwagon 'braun' ist auf Gleisabschnitt 2 und fest an Güterwagon 'rot' gekoppelt	1 mal	<input type="checkbox"/> iO <input type="checkbox"/> niO	

12 Auswertung

wird nach Testdurchführung erstellt