

Testplan

Für das studentische Projekt *Sichere Eisenbahnsteuerung*

Datum	08.04.2010
Quelle	ProVista
Autoren	Norman Nieß Kai Dziembala
Version	0.2
Status	In Bearbeitung

1 Historie

Version	Datum	Autor	Bemerkung
0.1	25.03.2010	Kai Dziembala	Erstellung der Kapitel 1 - 5
0.2	08.04.2002	Kai Dziembala	Komplettüberarbeitung auf Grund fehlerhafter Arbeit

2	Inhaltsverzeichnis	
1	Historie	2
2	Inhaltsverzeichnis	3
3	Einleitung	4
3.1	Referenzierte Dokumente	4
4	Planungsdaten	5
4.1	Verantwortlichkeiten	5
4.2	Zeitliche Planung	5
5	Strukturplan	6
5.1	Zu testende Hardware-Komponenten:	6
5.2	Zu testende Software-Module:	6
6	Hardware-Konfiguration	7
8	Testziele	9
9	Testverfahren	10
10	Testbeginn	11
11	Testreihenfolge	12
12	Testende	13
13	Testdaten	14

3 Einleitung

In diesem Dokument wird die Testplanung für das Hochschulprojekt 'Sichere Eisenbahnsteuerung' beschrieben. Es wird darauf eingegangen in welcher Reihenfolge die einzelnen Module in dem Gesamtsystem getestet werden. Dies dient der Definition eines vollständigen Testablaufs, damit die sichere Funktion der Eisenbahnsteuerung gewährleistet ist. Zu dem wird hier die Grundlage der Testspezifikation gebildet.

3.1 Referenzierte Dokumente

Software-Design, Version 1.0, Aulis: Design → Subsystemdesign → Software

Pflichtenheft, Version 1.1, Aulis: Anforderungsmanagement → Pflichtenheft

Meilensteine, Version 0.2, Aulis: Projektmanagement → Projektplanung

4 Planungsdaten

4.1 Verantwortlichkeiten

Für die Testdurchführung sind alle Projektmitglieder verantwortlich.

4.2 Zeitliche Planung

Zeitpunkt	Tätigkeit
25.03.2010	Beginn der Testplanung.
15.04.2010	Beginn der Testspezifikation.
Mitte Mai	Beginn der Tests

Tabelle 1: Zeitplanung der Tests

5 Strukturplan

Der Gesamtsystemtest wird in einen Test der Hardware-Komponenten und einen Test der Software-Module unterteilt.

5.1 Zu testende Hardware-Komponenten:

- Aufbau des Schienennetz
- Loks (manuelle fahrt) → testet zugleich:
 - Stromversorgung (Transformator)
 - Multimaus
 - Verstärker
 - Weichen
 - Entkoppler
- Not-Aus-Relais
- Hall-Sensoren
- Verkabelung der S88-Rückmeldemodule
- Stromversorgung und Verkabelung der Mikrocontroller
- Verkabelung RS232/XpressNet-Adapter

5.2 Zu testende Software-Module:

- Fahrprogramm 1
- Fahrprogramm 2
- Leitzentrale
- Befehlsvalidierung
- Ergebnisvalidierung
- S88-Treiber
- SSC-Treiber
- RS232-Treiber
- Not-Aus-Treiber
- Betriebsmittelverwaltung
- Auditing System
- Software Watchdog

6 Hardware-Konfiguration

Die Hardware auf der die Tests durchgeführt werden, entspricht der späteren produktiven Hardware. Hierbei handelt es sich sowohl um die Mikrocontroller, die Sensoren, das Gleissystem, wie auch die Züge. Genauere Informationen bezüglich der verwendeten Hardware ist im Hardwaredesign Dokument zu finden.

Als Testwerkzeuge dienen ein Multimeter und der Logikanalysator „Agilent Logic Wave“.

7 Software-Konfiguration

Die zu testenden Software-Module entsprechen den später im Real-Einsatz verwendeten und sind im Software-Design Dokument beschrieben.

Dabei dient die Programmierumgebung „µKeil Vision4“ und der Logikanalysator „Agilent Logic Wave“ als Testwerkzeuge.

8 Randbedingungen

8.1

9 Testziele

Die folgenden Testziele stammen aus dem Pflichtenheft.

- **Anwendung(-sschicht)**
 - Führt die Bahn in die richtige Richtung?
 - Wird die Fahraufgabe korrekt umgesetzt?
 - Werden Kollisionen vermieden?
 - Wird das Verlieren von Wagons ermittelt?
 - Voraussetzung: Gesamtanzahl und Startposition der Wagons bekannt
 - Reagiert die Software angemessen auf Signale der Hardware?
 - Funktioniert die Gleisfreiheitsprüfung?
- **Sicherheit(-sschicht)**
 - Reagiert die Software angemessen auf Signale der Hardware?
 - Stimmt die Anzahl der Wagons hinter dem Triebwagon nach Koppelvorgang?
 - Funktioniert die Gleisfreiheitsprüfung?
 - Werden Kollisionen verhindert?
- **Hardware(-komponenten)**
 - Sind die Weichen verstellbar?
 - Liefern die Sensoren korrekte Daten?
- **Schnittstellen**
 - Werden die Anweisungen der Anwendung in korrekte DCC-Befehle umgesetzt?
 - Werden die Hardware-Signale von der Software korrekt Ausgewertet?

10 Testverfahren

Als Testverfahren während des Integrationstests wird das Blackbox-Testverfahren verwendet. Die Integration der Module soll weitestgehend nach dem Bottom-Up-Verfahren erfolgen, wobei die Modulreihenfolge so gewählt wurde, dass möglichst wenig Testtreiber verwendet werden müssen.

11 Testbeginn

Um mit dem Integrationstest beginnen zu können, müssen

- das Gleissystem vollständig aufgebaut,
- mindestens ein Mikrocontroller an den Sensoren angeschlossen,
- mindestens ein Mikrocontroller an einem Not-Aus-Relais angeschlossen und
- das Modul 'Betriebsmittelverwaltung' vollständig implementiert und getestet

sein.

12 Testreihenfolge

1. Betriebsmittelverwaltung
2. Software Watchdog
3. Auditing System
4. Not-Aus-Treiber
5. S88-Treiber

Ab diesem Zeitpunkt sollte der zweite Mikrocontroller mit seinem Sensornetz über den S88-Bus und mit dem ersten Mikrocontroller über die SSC-Schnittstelle verbunden sein. Ansonsten muss die Kommunikation zwischen diesen simuliert werden.

6. SSC-Treiber
7. Befehlsvalidierung

6. und 7. können in der Reihenfolge getauscht werden.

8. Ergebnisvalidierung

Der Mikrocontroller muss über die RS232-Schnittstelle mit dem XpressNet-Bus verbunden sein.

9. RS232-Treiber
10. Leitzentrale
11. Fahrprogramm 1
12. Fahrprogramm 2

Während des gesamten Integrationstests soll sich nur Zug 1 auf dem Gleissystem befinden um zum Einen echte Sensorwerte zu bekommen, gleichzeitig aber einen möglichen Zusammenstoß zu verhindern, solange die Sicherheitsschicht noch nicht vollständig integriert ist.

Wurden die Fahrbefehle aus 'Fahrprogramm 1' und 'Fahrprogramm 2' korrekt von Zug 1 umgesetzt, wird der Test der beiden Fahrprogramme mit nur Zug 2 auf dem Gleissystem wiederholt.

Erst jetzt werden die beiden Fahrprogramme mit beiden Zügen gleichzeitig auf dem Gleissystem durchgeführt.

13 Testende

Der Integrationstest ist erfolgreich verlaufen und damit abgeschlossen, wenn beide Fahrprogramme mit beiden Zügen auf dem Gleissystem erfolgreich durchgeführt worden sind.

14 Testdaten

Es werden Testtreiber mit folgender Funktionalität benötigt:

- Streckenbefehle in den Shared Memory zwischen Anwendungs- und Sicherheitsschicht schreiben.
- Gegebenenfalls für den S88-, den SSC- und den RS232-Treiber Testtreiber erstellen, die Fehlermeldungen in den Shared Memory schreiben um die Reaktion auf Fehler zu überprüfen.