

HOCHSCHULE BREMERHAVEN

EMBEDDED SYSTEMS

Software Requirements Specifications

*Description of the expected software features, constraints,
interfaces and other attributes.*

Autoren: Jan Löwenstrom (34937) und Johann Hoffer (34461)
begleitet von
Prof. Dr. Lipskoch

1. Februar 2019

Inhaltsverzeichnis

1	Introduction	2
1.1	Project Overview	2
1.2	Quality Scenarios	2
2	Specific Requirements	3
2.1	External Interface Requirements	3
2.1.1	User Interfaces	3
2.1.2	Hardware Interfaces	3
2.1.3	Software Interfaces	4
2.1.4	Communications Protocols	4
2.2	Software Product Features	4
2.3	Software System Attributes	5
2.3.1	Reliability	5
2.3.2	Availability	6
2.3.3	Security	6
2.3.4	Maintainability	6
2.3.5	Portability	6
2.3.6	Performance	6

1 Introduction

1.1 Project Overview

Dieses Projekt wird im Rahmen der Veranstaltung „Eingebettete Systeme“ an der Hochschule Bremerhaven erstellt. Als Product-Owner und Steller der Anforderungen bzw. Rahmenbedingungen lässt sich der begleitende Professor Herr Dr. Lipskoch bestimmen. Im Projekt selber geht es um die Gestaltung einer Uhr, deren Zeit mit Hilfe des DCF77-Signals synchronisiert wird.

UPDATE: Aufgrund von Zeitmangel wurden für den aktuellen Prototypen nur die REQ-001, REQ-002, REQ-003 und REQ-004 verwirklicht. Der Fokus lag also auf der Synchronisation mit dem DCF77-Signal und der Darstellung der Zeit in einem nutzerfreundlichem Format, sprich auf einem Display. Der Anteil des Projekts, der die DCF-Zeit an einen RaspberryPi schickt, diese mit der GPS-Zeit vergleicht und die Funktion bietet diese Zeit mit dem NTP an Endgeräte zu schicken, wurden **nicht** umgesetzt.

1.2 Quality Scenarios

- Der Benutzer drückt Knopf 1 auf dem ATMega und spätestens nach 3 Sekunden wird die aktuelle Zeit im Format HH:mm:ss, DD:MM:YYYY und Wochentag in der Zeitzone Europe/Berlin auf einem Display angezeigt.
- Der DCF-77 Empfänger erhält ein Zeitsignal und leitet es umgehend an den ATMega weiter.
- Der ATMega erhält kontinuierlich Informationen im DCF-77 Format, wandelt es in ein Date-Format um, synchronisiert sich mit diesem und inkrementiert es anschließend sekundenweise.
- Der Raspberry Pi sendet eine Anfrage an den ATMega, erhält daraufhin die aktuelle Zeit im Date-Format, synchronisiert sich mit dieser und inkrementiert sie anschließend sekundenweise.
- Der Raspberry Pi erhält über die Ethernetverbindung die aktuelle Zeit, die mit Hilfe eines GPS Empfängers empfangen wurde.
- Der Raspberry Pi stellt seine aktuelle Zeit mittels NTP zur Verfügung.
- Der Benutzer kann die Systemzeit seinen Endgeräts mit der Zeit des Raspberry Pis mittels NTP synchronisieren.

- Der Raspberry Pi berechnet die Differenz zwischen der empfangen Zeit des ATMegas, die ursprünglich vom DCF-77 Empfänger stammt, mit der, über das Ethernet erhaltenen, GPS Zeit.
- Der Benutzer drückt einen Knopf auf dem ATMega und auf einem separaten Display wird Datum, Uhrzeit und der Wochentag angezeigt.
- Der Benutzer drückt einen Knopf auf dem ATMega und das Display zeigt nichts mehr an.

2 Specific Requirements

2.1 External Interface Requirements

2.1.1 User Interfaces

- Ein Knopf auf dem Evaluations-Board
- Das angeschlossene Telefondisplay
- Der Benutzer kann mit dem System interagieren, indem er einen Knopf auf dem Evaluations-Board drückt, um etwas auf dem Display anzeigen zu lassen.
- Der Benutzer drückt erneut auf den Knopf und das Display wird ausgeschaltet.

2.1.2 Hardware Interfaces

- Das DCF77-Empfängermodul muss vom ATMega verbunden sein und dessen Datensignale auslesen können.
- Der ATMega muss Anweisungen senden, die das Interface des Displays bedienen.
- Der ATMega ist über ein RS232-Gerät mit dem Raspberry Pi verbunden. (Das RS232 Gerät ist notwendig, weil der ATMega mit 0-5V arbeitet und der Raspberry Pi mit 0-3,3V)
- Der Raspberry Pi ist mittels Ethernet an die Hochschulinfrastruktur angebunden. Der Raspberry Pi, welcher das GPS Signal verteilt, ist ebenfalls Teil dieses Netzwerkes.

2.1.3 Software Interfaces

- Unser Programm muss das Interface für den Simulator „simavr“ bedienen
- Der Simulator erzeugt eine Datei als Output in dem Format .vcd, welches wiederum von dem Programm „GTK-Wave“ dargestellt werden kann

2.1.4 Communications Protocols

- Der ATMEGA und der RaspberryPi kommunizieren über ein **RS-232** Gerät
- Das Endgerät synchronisiert seine Systemzeit mit der Zeit, die der Raspberry Pi zur Verfügung stellt, mittels des **NTP (Network Time Protocol)**
- Die Zeit, die das RPI im Fährhaus durch den GPS-Empfänger erhalten hat, erhält man mittels **TCP/IP** (beide Geräte sind an die Hochschulinfrastruktur angebunden und die aktuelle Zeit des RPI im Fährhaus ist über "hopperreinsehbar").
- Das Zeitsignal stammt vom Langwellensender **DCF77**, der sich in Mainflingen bei Frankfurt am Main befindet. Es wird auf einer Frequenz von 77,5 kHz übertragen. Eine Nachricht enthält 59 Bits und dauert 60 Sekunden (61 Sekunden, falls am Ende der Minute eine Schaltsekunde stattfindet). Zu Beginn jeder Sekunde gibt es eine Amplitudenabsenkung. Hat die Absenkung eine Dauer von 100ms, handelt es sich um ein 0-Bit, bei 200ms Absenkung um ein 1-Bit. Danach folgen 900ms bzw. 800ms maximale Amplitude. Somit werden von Sekunde 0 bis Sekunde 58 Bit 0 bis Bit 58 übertragen. Die Sekunde 59 (letzte Sekunde einer Minute) enthält keine Absenkung und auf diese Weise wird Sekunde 0 der nächsten Minute mit der nächsten Absenkung markiert. Falls eine Schaltsekunde stattfindet, enthält Sekunde 59 eine Absenkung und es gibt eine Sekunde 60 ohne Absenkung, sodass die nächste Absenkung wieder Sekunde 0 markiert.

Link: Detaillierte DCF-Beschreibung (dcf77logs.de)

2.2 Software Product Features

Folgende Features bietet die Software auf dem ATMEGA:

- REQ-001: errechnet aus der binären Folge des DCF-Empfängers die entsprechenden Werte für Minuten, Stunden, Tag, Monat, Jahr und Wochentag
- REQ-002: synchronisiert sich mit der erhaltenen Zeit, wenn diese valide ist

- REQ-003: zählt die interne Zeit (ATime) sekundenweise weiter
- REQ-004: steuert das Display an und lässt die o.g. (weiter gezählten) Werte (ATime) anzeigen
- REQ-005: leitet die Zeit an den Raspberry Pi weiter
- optional:
 - REQ-006: erhält die GPS-Zeit und berechnete Differenz des Raspberry Pis
 - REQ-007: stellt die erhaltene GPS-Zeit und die Differenz auf einem Display dar.

Folgende Features bietet die Software auf dem RaspberryPi:

- REQ-008: nimmt die Zeit des ATMegas entgegen
- REQ-009: wandelt sie in das für NTP notwendige Format um, zählt sie sekundenweise weiter und stellt sie via NTP zur Verfügung
- REQ-010: vergleicht sie mit der erhaltenen GPS-Zeit (Differenzberechnung)
- optional:
 - REQ-011: sendet die GPS-Zeit und die berechnete Differenz an den ATMega

2.3 Software System Attributes

2.3.1 Reliability

- Es müssen stets die richtigen Zeitwerte berechnet, dargestellt und zur Verfügung gestellt werden, invalide Daten wie z.B. Daten in der Vergangenheit oder Zukunft werden nicht mit der ATime synchronisiert oder gar angezeigt
- Die Information, wann sich die ATime das letzte Mal mit der DCF77-Zeit synchronisiert hat, muss von außen einsehbar sein
- Die Information, ob die angezeigte Zeit noch zuverlässig ist, muss einsehbar sein
- Für den Code, der im Raspberry Pi und im ATMega läuft, sollen (Unit)-Tests existieren

2.3.2 Availability

- Das System muss darauf ausgelegt sein 24/7 zu laufen.
- Das System muss bei jedem Einschalten in vollem Umfang funktionsfähig sein

2.3.3 Security

- Die Zeit des Systems muss von außen nicht manipulierbar sein

2.3.4 Maintainability

- Der Quellcode an sich muss umfangreich kommentiert, sodass Außenstehende diesen leicht verstehen können.
- Die Software soll modular aufgebaut sein, damit neuer Code leichter zu integrieren ist.
- Eine umfangreiche Dokumentation im IEEE Standard erläutert u.a. Designentscheidungen und liefert Beschreibung zum ausführlichen Testen der Hard- und Software.

2.3.5 Portability

- Das System/ Hardware ist darauf ausgelegt, statisch zu sein. D.h. Verbindungen zwischen den einzelnen Komponenten sind i.d.R. zu fragil, da sie nur aus dünnen Kupferdrähten bestehen.
- Das System benötigt eine externe Stromzufuhr und ist nicht batteriebetrieben.
- Die Software des ATMegas ist spezifisch auf diesen Prozessortyp (ATMega32) und das Atmel Evaluations-Board Version 2.0 von Pollin zugeschnitten und kann somit nicht ohne weiteres auf einen andere Mikrokontroller oder Evaluations-Board übertragen und eingesetzt werden.

2.3.6 Performance

- Nachdem der Benutzer den KNOPF1 gedrückt hat soll die interne Zeit des ATMegas (ATime) spätestens nach drei Sekunden auf dem externen Display angezeigt werden.
- Bei der Bereitstellung der Zeit für ein externen Endgerätes mittels NTP sollen nach der Initialisierung der Synchronisation maximal drei Sekunden vergehen bis diese abgeschlossen ist.

- Die Software und Hardware muss performant genug sein, so dass das Weiterzählen der Zeit und das Anzeigen auf dem Display ohne Probleme sekundenweise funktioniert