



**School of
Engineering**

InES Institute of
Embedded Systems

Konzept

Messstation zur Registrierung von Geschiebe- Bewegungen im Fluss

Autoren

Tobias Keller
Tobias Welti

Datum

15.09.2014

Inhaltsverzeichnis

1. Einleitung	5
1.1. Überblick	5
1.2. Datenlogger	5
1.2.1. Messdaten sammeln	6
1.2.2. Kontrolle über das Bussystem	6
1.2.3. Steuerung des Betriebs	6
1.2.4. Schnittstelle nach Aussen	6
1.3. Sensoreinheit	6
1.3.1. Messdatenerfassung	6
1.3.2. Ereigniserkennung	7
1.3.3. Datenübertragung	7
1.4. Bussystem	7
2. Hardware-Konzept	8
2.1. Hardware-Architektur	8
2.1.1. Datenlogger	8
2.1.2. Sensoreinheit	8
2.1.3. Bussystem	9
2.2. Komponentenauswahl	9
2.2.1. Mikroprozessor	9
2.2.2. Bus-System	10
2.2.3. Speichermedium	11
2.2.4. Sensor	12
2.2.5. Schnittstelle	12
2.3. Komponenten	12
2.3.1. Cortex M4 Mikroprozessor	12
2.3.2. Beschleunigungs-Sensor	12
2.3.3. CAN Bus	12
2.3.4. SD Karte	13
2.3.5. UART Schnittstelle	13
2.4. Datenlogger	13
2.5. Sensoreinheit	13
3. Software-Konzept	15
3.1. Software-Stack	15
3.1.1. Überblick	15
3.1.2. Messdatenerfassung	16
3.1.3. Ereigniserkennung	16
3.1.4. Timestamp	17
3.1.5. Verwaltung der Messstation	17
3.1.6. Busprotokoll	19
3.1.7. Filesystem	19
3.1.8. UART-Kommandozeile	19
3.2. Funktionalität	19
3.3. Konfiguration	19
4. Verzeichnisse	20

Literaturverzeichnis	21
Literaturverzeichnis	21
Abbildungsverzeichnis	22
(Abbildungsverzeichnis)	22
Tabellenverzeichnis	23
(Tabellenverzeichnis)	23
(Listingverzeichnis)	25
A. Anhang	I
A.1. Projektmanagement	I
A.2. Weiteres	I

Liste der noch zu erledigenden Punkte

glossareintrag wieder entfernen	5
Abkürzungsverzzzzeichenbeispieleintrag wieder entfernen	5
remove following line	6
Tabelle verschiedener uCs zum Vergleich	9
Kriterienliste RS485/CAN einfügen, Lit-Referenz auf White Paper von IXXAT	11
Literatur-Referenzen in Tabelle 2.2	11
Sensorauswahl beschreiben	12
Schnittstellenauswahl beschreiben	12
Texten	12
Texten	12
Texten	12
Texten	12
Texten	12
Texten	12
Texten	13
Texten	13
Texten	13
Texten	13
Texten	13
Glossar: Impact, Bus, Mikroprozessor, A/D-Wandler, Pin, Sensor, Threshold/Grenzwert,	15
FSM-figur anpassen auf neue Stati	16
Ereigniserkennung beschreiben, welche Parameter können konfiguriert werden	17
Zusammenhänge A/D-Wandlung und Ereigniserkennung und Übertragung beschreiben	17
Verschiedene Betriebsmodi mit Grafiken beschreiben	17
Berechnungen, in welchem Modus wie lange gemessen werden kann, und wie lange ein Sensor mit dem vorhandenen Speicher die Resultate zwischenspeichern kann. Allenfalls ein System erwähnen, das automatisch zwischen verschiedenen Modi hin- und herschalten kann. (Ist allerdings heikel). Wie viele Sensoren können in welchem Modus gleichzeitig am System betrieben werden, bei welcher Ereignisrate ist Schluss mit Busbandbreite.	17
Timestamp beschreiben, Rechnung über die Dauer der eindeutigen Zuweisung.	17
Busverwaltung beschreiben	17
Figur 3.4 aufteilen auf zwei Seiten. (PDF-crop)	18
Busprotokoll austüfteln. Darstellung siehe HW-Konzept Rioxo, genaue Beschreibung der Nachrichtentypen. Timestamp der einzelnen Peaks bezieht sich auf Offset vom Beginn des Impacts. 19	

Kommunikationsdiagramm Bushandler	19
Interrupt-System des Bushandlers aufführen	19
Texten	19
Frage: wird für jeden Sensor ein eigenes File geführt? Kann man alle Files offen lassen oder ist das keine gute Idee? Was ist besser, jedes mal das File-Ende zu suchen um neues anzuhängen?	19
Dokumentation über die Kommandos, wird später für die Bedienungsanleitung gebraucht	19
Texten	19
Hier eine Art Bedienungsanleitung zur Konfiguration geben. Welches Kommando hat was zur Folge? (Wird Datenerfassung neu gestartet, werden allenfalls andere Sensoren deaktiviert etc.	19

1. Einleitung

1.1. Überblick

Das zu entwickelnde Messsystem kann grob in drei Komponenten aufgeteilt werden.

1. Datenlogger
2. Sensoreinheit
3. Bussystem

Der Datenlogger hat die Aufgabe, von mehreren Sensoreinheiten registrierte Ereignisse zu empfangen und zu speichern. Die Sensoreinheiten messen kontinuierlich Vibrationen, werten die Signale aus und erkennen Ereignisse, die einer vordefinierten Signalform entsprechen. Alle Einheiten sind über ein Bussystem verbunden, um miteinander kommunizieren zu können. Der prinzipielle Aufbau ist in Abbildung 1.1 ersichtlich. Die Stromversorgung der Anlage wird am Datenlogger angeschlossen. Parallel zum Kabel des Datenbusses wird die Stromversorgung der Sensoreinheiten geführt.

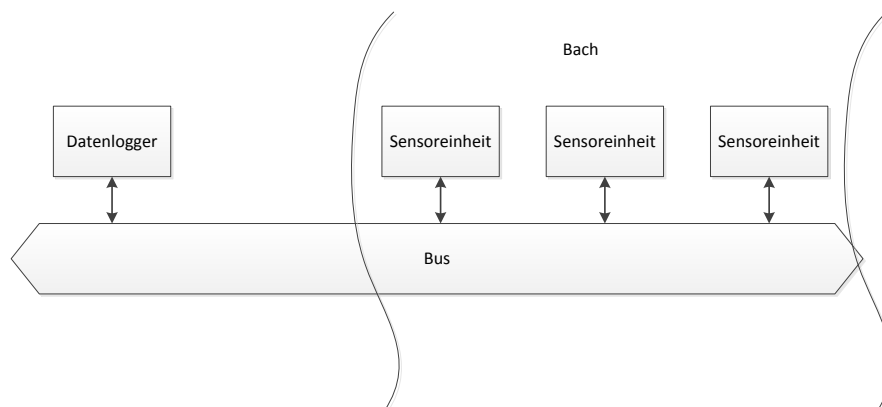


Abbildung 1.1.: Eine Messstation mit einem Datenlogger, der mehrere Sensoreinheiten im Bach steuert.

Hier wird der Beispieglossareintrag das erste Mal eingesetzt, hier gleich das zweite Mal: Beispieglossareintrag.

glossareintrag wieder entfernen

Erster Einsatz des Abkürzzzzzzzungsverzeichnisbeispieleintrag (AVZBE)s, zweiter Einsatz AVZBE.

Abkürzzzzzzz wieder entfernen

Diese drei Einheiten werden im Folgenden genauer definiert.

1.2. Datenlogger

Der Datenlogger hat verschiedene Aufgaben zu erfüllen:

- Sammeln und speichern der Messdaten der Sensoreinheiten.
- Kontrolle über das Bussystem.

- Steuerung des Betriebs der Anlage.
- Schnittstelle für die Konfiguration der Anlage und für das Auslesen der Messdaten.

1.2.1. Messdaten sammeln

Für jede angeschlossene Sensoreinheit führt der Datenlogger eine Datensammlung, in der die registrierten Ereignisse zeitlich sortiert abgespeichert werden. Die Datensammlungen werden in Dateien abgelegt, die auf einem externen, auswechselbaren Medium abgespeichert werden. So können die Messdaten auf einfache Art für die weitere Auswertung abgeholt werden.

1.2.2. Kontrolle über das Bussystem

Als Busmaster hat der Datenlogger die Aufgabe, allen angeschlossenen Einheiten eine eindeutige Identifikationsnummer (ID) zuzuweisen. Über diese ID erkennt der Datenlogger, von welcher Sensoreinheit Daten übertragen werden. Für die Zuordnung der Messdaten zu einem bestimmten Sensor benötigen die Sensoreinheiten ein fixes Erkennungsmerkmal, z.B. eine Seriennummer, die mit den Messdaten abgespeichert werden soll.

1.2.3. Steuerung des Betriebs

Die Messstation hat verschiedene Betriebsmodi, die über den Datenlogger angewählt werden können. Der Datenlogger steuert die einzelnen Sensoreinheiten entsprechend an.

1.2.4. Schnittstelle nach Aussen

Über eine Schnittstelle am Datenlogger kann ein Computer angeschlossen werden. Per Kommandozeile wird die Messstation konfiguriert, der Zustand überprüft und der Betriebsmodus gewählt.

1.3. Sensoreinheit

Die Aufgaben der Sensoreinheit umfassen:

- Erfassung von Messdaten.
- Erkennung von Ereignissen.
- Übertragung der Ereignisdaten an den Datenlogger.

following line AVZBE

1.3.1. Messdatenerfassung

Der Sensor zur Erfassung der Daten wird mit einer vordefinierten Abtastrate ausgelesen. Die Abtastrate muss so gewählt werden, dass einzelne Ereignisse erkannt werden können, ohne unnötig viel Messdaten zu generieren.

1.3.2. Ereigniserkennung

Im Mikroprozessor werden die Messdaten fortlaufend analysiert. Überschreitet das gemessene Signal einen gewissen Pegel (Threshold), markiert dies den Beginn eines Ereignisses. Das Ereignis ist beendet, wenn der Signalpegel für eine gewisse Zeit unterhalb des Threshold bleibt. Für jedes Ereignis wird abgespeichert, wann es aufgetreten ist (Timestamp), wie lange es gedauert hat, wie hoch der Signalpegel maximal ausschlug und wie viele Signalspitzen (Peaks) aufgetreten sind. Allenfalls können auch die Höhen aller Peaks übertragen werden.

1.3.3. Datenübertragung

Die Sensoreinheit sendet die Messdaten regelmässig über das Bussystem an den Datenlogger. Nach Bestätigung des Erhalts werden die Messdaten aus dem Speicher der Sensoreinheit gelöscht.

1.4. Bussystem

Das Bussystem verbindet die Einheiten der Messstation miteinander. Die gesamten Messdaten und Steuerkommandos werden über den Bus übertragen. Das Bussystem muss die Datenmenge der angeschlossenen Sensoren bewältigen können, über die geforderte Distanz funktionieren und möglichst robust gegenüber äusseren Einflüssen sein. Der Busmaster hat die Möglichkeit, laufende Übertragungen von Sensoreinheiten zu unterbrechen, um Steuerkommandos zu senden.

2. Hardware-Konzept

2.1. Hardware-Architektur

Anhand der funktionalen Vorgaben für die Messstation werden der Datenlogger, die Sensoreinheit und das Bussytem im folgenden genauer spezifiziert und die Komponenten ausgewählt.

2.1.1. Datenlogger

Der Datenlogger sammelt die Daten der Sensoreinheiten über das Bussytem ein und speichert sie ab. Dafür benötigt er das Bussytem, einen Mikroprozessor, internen Speicher und ein leicht auswechselbares Speichermedium. Ausserdem soll über eine Schnittstelle ein Computer angeschlossen werden können, um den Betrieb der Messstation zu steuern. Der Datenlogger wird in einem wasserdichten Gehäuse untergebracht. Für den Austausch des Speichermediums wäre eine verschraubbare Öffnung denkbar.

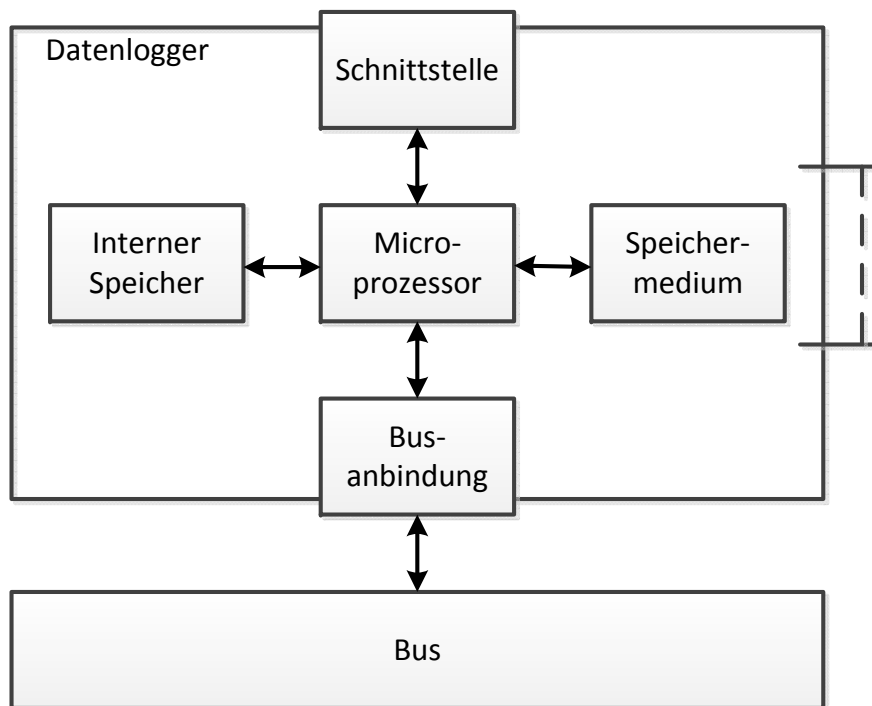


Abbildung 2.1.: Hardwarekonzept des Datenloggers.

2.1.2. Sensoreinheit

Die Sensoreinheit benötigt einen Beschleunigungssensor, um die Einschläge von Geschiebe zu messen. Über einen Analog-Digital-Wandler (ADC) werden die Messsignale digitalisiert. Die gemessenen Signale werden von einem Mikroprozessor verarbeitet, im internen Speicher zwischengespeichert und über das Bussytem an den Datenlogger übertragen.

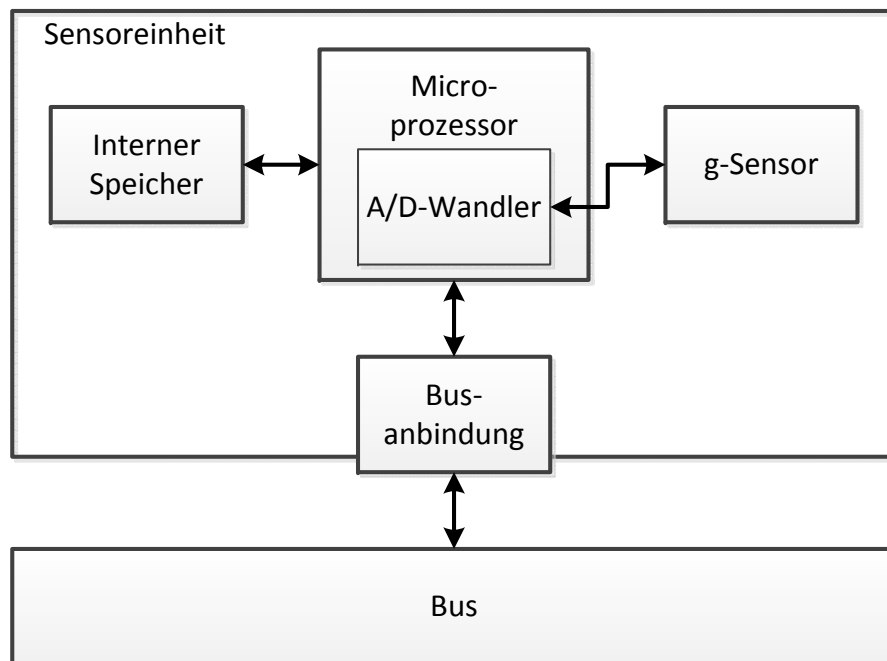


Abbildung 2.2.: Hardwarekonzept der Sensoreinheit.

2.1.3. Bussystem

Das Bussystem muss die Daten und Befehle zwischen Datenlogger und Sensoreinheiten übertragen. Die Reichweite des Bussystems muss genügen, um alle Komponenten der Messinstallation zu verbinden. Die Datenbandbreite muss die Übertragung der Messresultate aller Sensoren erlauben.

2.2. Komponentenauswahl

2.2.1. Mikroprozessor

Bei der Auswahl des Mikroprozessors werden folgende Kriterien berücksichtigt:

- Rechenleistung genügend für allfällige zusätzliche Anforderungen.
- Analog-Digital-Wandler mit genügender Abtastrate und Auflösung.
- Digitaler Signal Prozessor integriert für die Verarbeitung der Messdaten.
- Ein-/Ausgänge für das Bussystem.
- Ein-/Ausgänge für den externen Speicher.
- möglichst geringer Stromverbrauch.

	Bitrate	Distanz	Clients	Besonderheiten
CAN	1 MBit/s 125 kBit/s	40 m 500 m	> 20	+ Collision Detection (CD) umgehen mit Polling durch Master. + Bei synchronem CAN wird CD durch ID gelöst. + CAN Controller sendet Interrupt Request bei erhaltener Nachricht.
SPI	..100 MBit/s	< 1 m	slave select	- Pro Client eine Slave Select Leitung - Daisy Chain \Rightarrow alle MC beschäftigt. - Bei Ausfall eines MC ganzer Bus unterbrochen.
RS485	35 MBit/s 100 kBit/s	10 m 1200 m	>32	- Master am besten in der Mitte des Bus \Rightarrow ungünstig. - Braucht 2..4 Drähte (bei Full Duplex) - braucht pull-up und pull-down Widerstände \Rightarrow mehr Leistungsaufnahme.
Ethernet	100 MBit/s	100 m	> 20	+ Stromversorgung bei Power over Ethernet (PoE) integriert. - kein Bus sondern allenfalls Daisychain. - bei Daisychain kein PoE möglich.
Feldbus				ist eine Familie von Bussen, z.B. CAN-Bus
I2C	0.4..5 Mbit/s	wenige Meter	< 20	nur für kurze Distanzen, Bitrate nimmt rasch ab.

Tabelle 2.1.: Entscheidungsmatrix für die Auswahl des Bussystems.

2.2.2. Bus-System

Anhand folgender Kriterien wurde ein Bussystem ausgewählt:

- Übertragungsbandbreite genügend für fortlaufende Übertragung von Rohdaten einer Sensoreinheit.
- Reichweite mindestens 20 Meter.
- Robust gegenüber äusseren Einflüssen.
- Mindestens zwanzig Busteilnehmer möglich.

In Tabelle 2.1 sind die Eigenschaften diverser Bussysteme aufgeführt.

Kommentare SPI und I2C sind nur für kurze Distanzen geeignet und sind deshalb keine Option. Die Verwendung von Ethernet zur Datenübertragung würde zwei Schnittstellen auf jeder Sensoreinheit voraussetzen, um die Sensoren hintereinander zusammenzuhängen (Daisychain). Jedes Paket müsste vom Microcontroller weitergeleitet werden, wenn es für einen anderen Empfänger bestimmt ist. Dies führte zu einer zusätzlichen Belastung der Microcontroller. Stromversorgung über Ethernet ist mit PowerOverEthernet (PoE) zwar möglich, erfordert aber spezielle Geräte zur Speisung über den Stecker des Datenkabels. Dies verunmöglicht eine Daisychain mit PoE, neben dem Datenkabel wäre noch ein Kabel für die Stromversorgung notwendig.

Vergleich CAN-Bus und RS485 CAN und RS: Stecker nicht definiert => wasserdichte Stecker einfach zu finden.

Kriterienliste
RS485/CAN ein
Lit-Referenz auf
Paper von IXXA

Entscheidung CAN-Bus erfüllt alle Kriterien und erlaubt es, den Busmaster am Ende des Bus zu platzieren. Dies ist ein Vorteil gegenüber RS485, wo der Master in der Mitte platziert werden sollte. CAN-Bus bietet bereits Kollisionserkennung und Fehlererkennung, während dies bei RS485 in der Software gelöst werden muss. Für CAN-Bus sind Bus-Treiber (Transceiver) erhältlich, die mit hohen Spannungen umgehen können, was das Bussystem robuster gegenüber Umwelteinflüssen macht. Die Grösse der Datenpakete ist bei CAN-Bus auf 8 Byte begrenzt, bei RS485 werden die Datenpakete über die Software frei definiert, was ein klarer Vorteil von RS485 darstellt. Insgesamt überwiegen die Vorteile von CAN-Bus klar.

2.2.3. Speichermedium

Kriterien Das externe Speichermedium soll möglichst klein sein, wenig Stromverbrauch haben und einfach auswechselbar sein. Bei Inaktivität sollte das Medium wenn möglich keinen Strom verbrauchen. Für einen mehrwöchigen unabhängigen Betrieb einer Messstation muss genügend Speicherkapazität bereitgestellt werden.

Datenmenge Pro Sensor werden bei hohem Geschiebeaufkommen maximal hundert Ereignisse pro Sekunde erwartet. Ein solches Geschiebeaufkommen stellt jedoch die Ausnahme dar. Ein Ereignis benötigt je nach verlangtem Detailgrad und Dauer des Ereignisses 10..90 Byte Speicherplatz. Für den normalen Betriebsmodus werden 50 Byte/Ereignis gerechnet, bei 5 Ereignissen pro Sekunde. Damit ergibt sich eine Datenrate von 250 Byte/s, die es pro Sensor abzuspeichern gilt. Mit zehn Sensoren im Einsatz müssen 2.5 kByte/s gespeichert werden.

Unabhängige Betriebsdauer Pro Gigabyte Speicherplatz können 111 Stunden Daten für zehn Sensoren gespeichert werden. Bei hohem Geschiebeaufkommen mit zwanzig mal mehr Ereignissen bleiben immer noch 5 Stunden Aufzeichnungszeit pro Gigabyte. Begnügt man sich mit weniger Details, reichen fallen pro Sensor in zehn Sekunden rund 400 Byte Daten an. Bei dieser Datenrate reicht ein Gigabyte für rund 700 Stunden. Auch bei hohem Geschiebeaufkommen kann die Anlage mehrere Tage an Daten speichern.

Kapazität Heute sind Speichermedien mit Kapazitäten bis über 128 GB erhältlich, so dass die Detailrate kein entscheidendes Kriterium mehr darstellt.

Datentransfer Für den Transfer der Daten aus dem Datenlogger auf einen Computer gibt es grundsätzlich zwei Varianten. Entweder man liest die Daten über eine Schnittstelle auf den Computer aus, oder man tauscht das Speichermedium aus. Das Auslesen via Schnittstelle benötigt zusätzlich Strom, das Wechseln des Speichermediums setzt einen mehr oder weniger komfortablen und trotzdem wasserdichten Zugang zum Medium voraus. Da heute Speichermedien mit kleinem Platzbedarf erhältlich sind, könnte ein solcher Zugang recht einfach mit einem Schraubverschluss realisiert werden.

Vergleich In Tabelle 2.2 werden verschiedene Speichermedien miteinander verglichen. In der Spalte 'Breite' ist aufgelistet, wie gross eine Öffnung mindestens sein muss, um das Speichermedium wechseln zu können. 'Pins' gibt an, wie viele Leitungen für den Anschluss des Mediums am Microcontroller nötig sind. Der Stromverbrauch in Klammern ist für den Standby-Modus des Speichermediums.

Literatur-Referenz
Tabelle 2.2

	Breite	Pins	Stromverbrauch	Bemerkungen
SD-Card	24 mm	9	20..100 mA (0.2 mA)	4 bit breiter serieller Bus
CompactFlash	43 mm	50	max. 70 mA (k.A.)	paralleler Bus
USB-Stick	min. 12 mm	4	typ. 70 mA (k.A.)	

Tabelle 2.2.: Entscheidungsmatrix zur Auswahl des Speichermediums.

Entscheid Für einen verschraubbaren Verschluss ist die CompactFlash-Karte zu breit, das Gehäuse würde dadurch sehr gross werden. Die SD-Karte und der USB-Stick sind vergleichbar in der Grösse. Von der SD-Karte sind auch kleinere Varianten erhältlich. Eine Öffnung für den Austausch des Speichermediums kann eine gewisse Grösse ohnehin nicht unterschreiten, damit hineingegriffen werden kann. Da die SD-Karte im Standby den geringeren Stromverbrauch hat, wird der Datenlogger mit einem SD-Kartenleser ausgestattet.

2.2.4. Sensor

swahl beschrei-

2.2.5. Schnittstelle

cellenauswahl
oen

2.3. Komponenten

2.3.1. Cortex M4 Mikroprozessor

Flash Speicher

SDRAM

2.3.2. Beschleunigungs-Sensor

2.3.3. CAN Bus

CAN Transceiver

Texten

2.3.4. SD Karte

Texten

2.3.5. UART Schnittstelle

Texten

2.4. Datenlogger

Texten

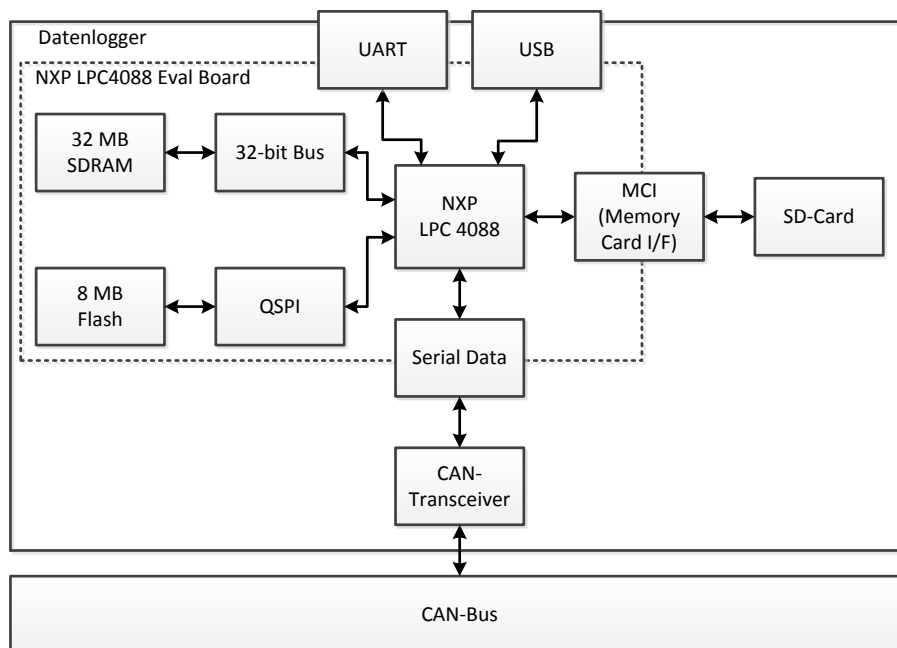


Abbildung 2.3.: Schematischer Hardware-Aufbau des Datenloggers.

2.5. Sensoreinheit

Texten

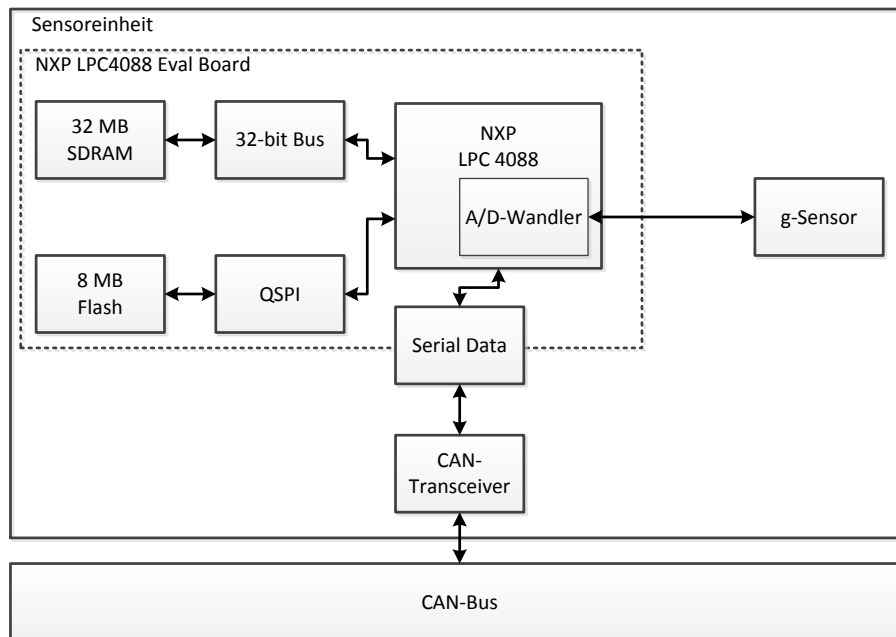


Abbildung 2.4.: Schematischer Hardware-Aufbau der Sensoreinheit.

3. Software-Konzept

3.1. Software-Stack

Glossar: Impact,
Mikroprozessor,
Wandler, Pin, Se
Threshold/Grenz

3.1.1. Überblick

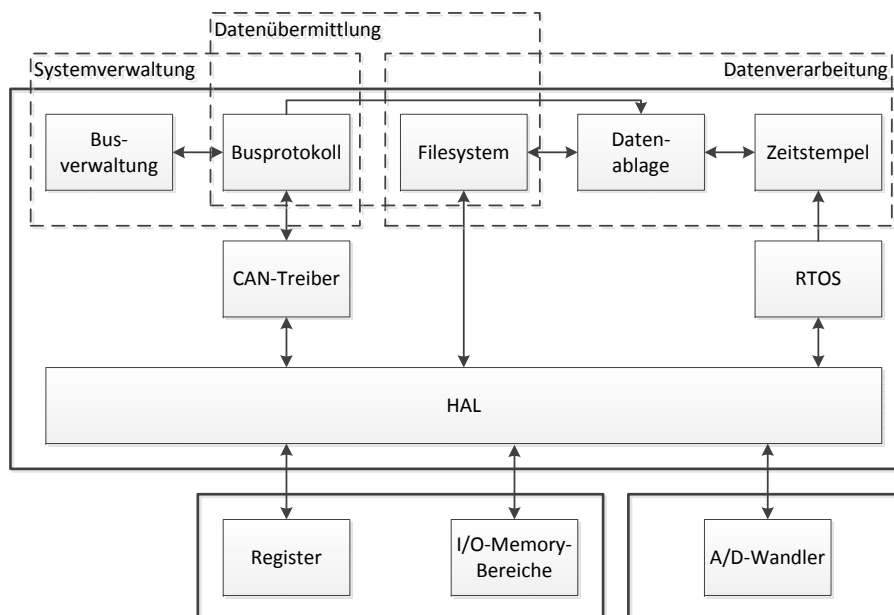


Abbildung 3.1.: Softwarestack des Datenloggers.

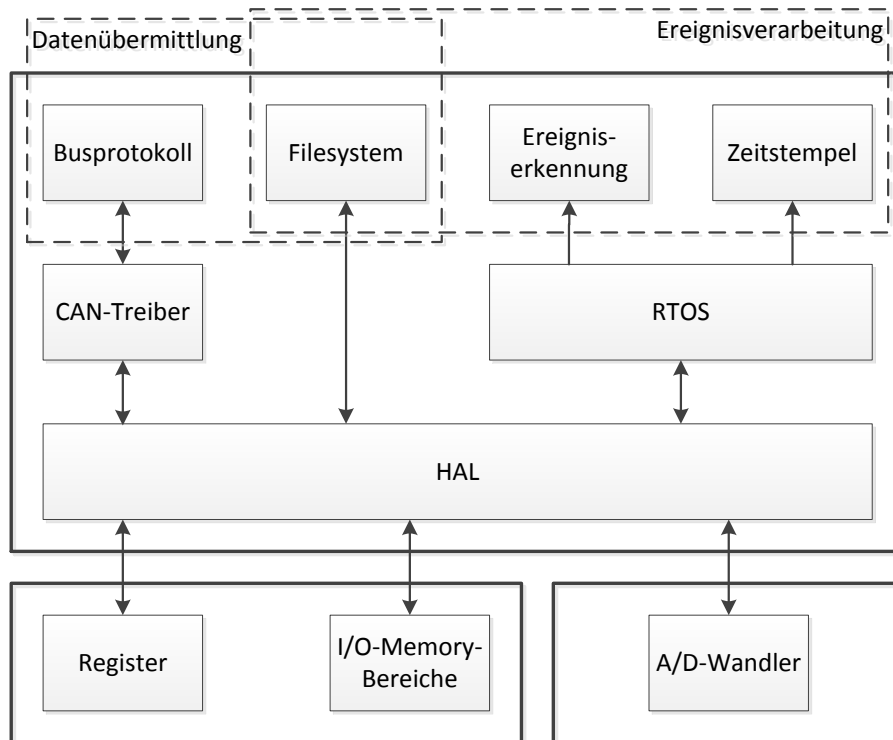


Abbildung 3.2.: Softwarestack der Sensoreinheit.

3.1.2. Messdatenerfassung

Der NXP LPC4088 Mikroprozessor verfügt über einen 12-bit A/D-Wandler, der über einen Multiplexer auf acht Pins messen kann. Auf dem verwendeten Quickstart-Board stehen 6 Pins für A/D-Wandlung zur Verfügung. Für die geplante Anwendung reicht ein A/D-Eingang, da der Beschleunigungs-Sensor die Beschleunigung nur auf einer Achse misst. Der A/D-Wandler des NXP LPC4088 wird mit einer Abtastrate von 10 kHz betrieben. Falls höhere Abtastraten nötig sind, kann der A/D-Wandler mit bis zu 400 kHz betrieben werden.

3.1.3. Ereigniserkennung

Vom WSL wurde die Ereigniserkennung bisher mittels Hilbert-Transformation gelöst. Die Hilbert-Transformation liefert die umhüllende Kurve des gemessenen Signals. Überschreitet die Umhüllende den Threshold, markiert dies den Start eines neuen Ereignisses. Fällt die Umhüllende unter den Threshold, ist das Ereignis beendet. Um den Rechenaufwand der Hilbert-Transformation zu umgehen, lösen wir die Ereigniserkennung einfacher.

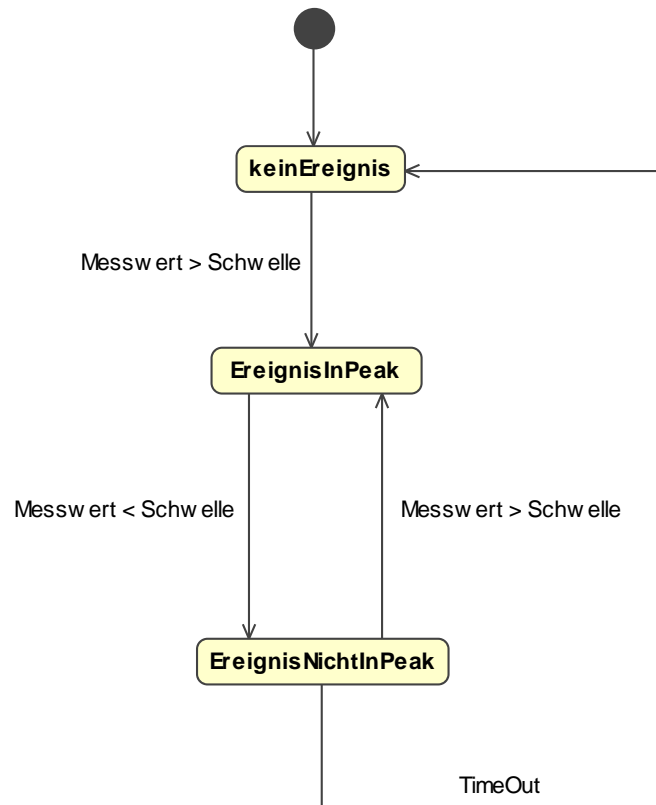


Abbildung 3.3.: Zustandsmaschine der Ereigniserkennung.

3.1.4. Timestamp

3.1.5. Verwaltung der Messstation

Ereigniserkennung
schreiben, welche
meter können ko
riert werden

Zusammenhänge
Wandlung und E
erkennung und
tragung beschrei

Verschiedene Be
modi mit Grafike
schreiben

Berechnungen, in
chem Modus wie
gemessen werden
und wie lange ei
mit dem vorhand
Speicher die Res
zwischenspeicher
Allenfalls ein Sys
erwähnen, das a
tisch zwischen v
denen Modi hin-
herschalten kann
allerdings heikel)
viele Sensoren kö
welchem Modus
zeitig am System
ben werden, bei
Ereignisrate ist S

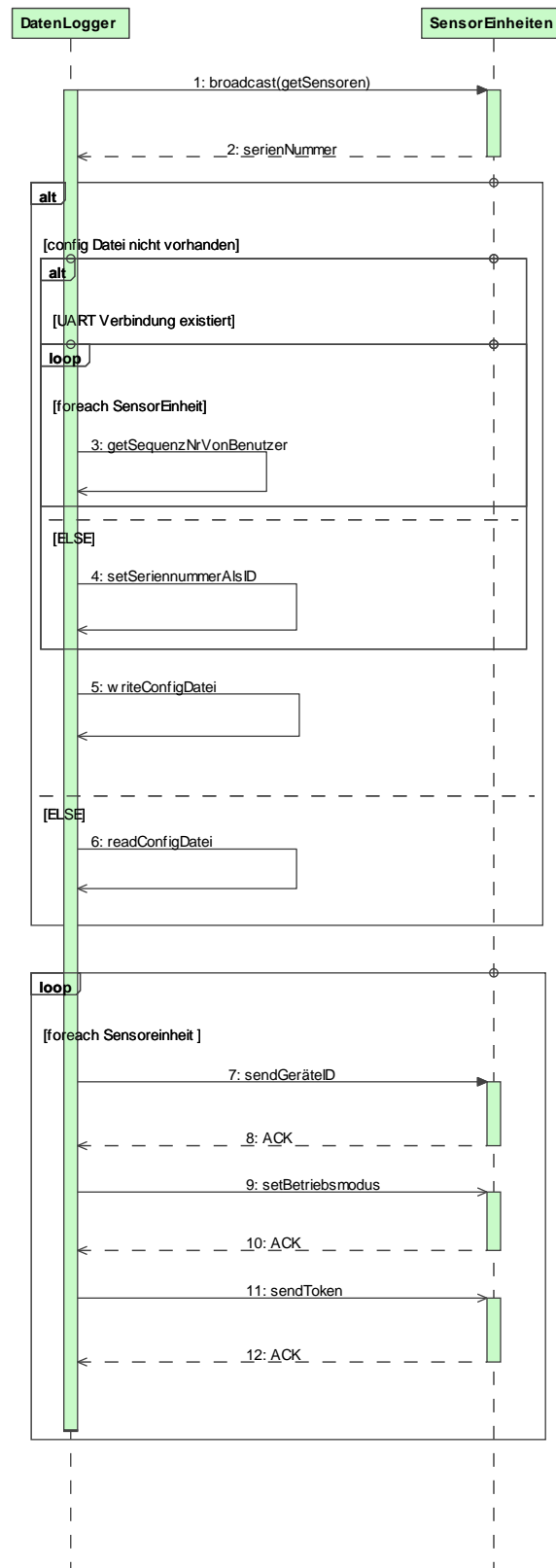


Abbildung 3.4.: Sequenzdiagramm des Startupvorgangs der Messstation.

3.1.6. Busprotokoll

3.1.7. Filesystem

3.1.8. UART-Kommandozeile

3.2. Funktionalität

3.3. Konfiguration

Busprotokoll aus
teln. Darstellung
HW-Konzept Ric
naue Beschreibu
Nachrichtentype
mestamp der ein
Peaks bezieht sic
Offset vom Begi
Impacts.

Kommunikations
Bushandler

Interrupt-System
Bushandlers auff

Texten

Frage: wird für j
Sensor ein eigene
geführt? Kann m
Files offen lassen
ist das keine gut
Was ist besser, j
mal das File-End
suchen um neues
hängen?

Dokumentation
Kommandos, wir
ter für die Bedie
anleitung gebrau

Texten

Hier eine Art Be
nungsanleitung z
figuration geben
ches Kommando
was zur Folge? (
Datenerfassung
startet, werden a
andere Sensoren
viert etc.

4. Verzeichnisse

Literaturverzeichnis

- [1] B. Klaus and P. Horn, Robot Vision. Cambridge, MA: MIT Press, 1986.
- [2] L. Stein, ">Random patterns,"> in Computers and You, J. S. Brake, Ed. New York: Wiley, 1994, pp. 55-70.

Abbildungsverzeichnis

1.1. Eine Messstation mit einem Datenlogger, der mehrere Sensoreinheiten im Bach steuert.	5
2.1. Hardwarekonzept des Datenloggers.	8
2.2. Hardwarekonzept der Sensoreinheit.	9
2.3. Schematischer Hardware-Aufbau des Datenloggers.	13
2.4. Schematischer Hardware-Aufbau der Sensoreinheit.	14
3.1. Softwarestack des Datenloggers.	15
3.2. Softwarestack der Sensoreinheit.	16
3.3. Zustandsmaschine der Ereigniserkennung.	17
3.4. Sequenzdiagramm des Startupvorgangs der Messstation.	18

Tabellenverzeichnis

2.1. Entscheidungsmatrix für die Auswahl des Bussystems.	10
2.2. Entscheidungsmatrix zur Auswahl des Speichermediums.	12

Wenn es klappt, müsste das Glossar hier erscheinen:

Glossar

AVZBE Abkürzzzzzzzungsverzeichnisbeispieleintrag 5, 6

Beispielglossareintrag Ein Eintrag im Glossar, der nur für den Zweck der Demonstration der Anwendung des Glossars angelegt wurde. 5

und zwar oberhalb dieser Zeile.

Listings

A. Anhang

A.1. Projektmanagement

- Offizielle Aufgabenstellung, Projektauftrag
- (Zeitplan)
- (Besprechungsprotokolle oder Journals)

A.2. Weiteres

- CD mit dem vollständigen Bericht als pdf-File inklusive Film- und Fotomaterial
- (Schaltpläne und Ablaufschemata)
- (Spezifikationen u. Datenblätter der verwendeten Messgeräte und/oder Komponenten)
- (Berechnungen, Messwerte, Simulationsresultate)
- (Stoffdaten)
- (Fehlerrechnungen mit Messunsicherheiten)
- (Grafische Darstellungen, Fotos)
- (Datenträger mit weiteren Daten (z.B. Software-Komponenten) inkl. Verzeichnis der auf diesem Datenträger abgelegten Dateien)
- (Softwarecode)