

Studiengang: Informationstechnik

Entwicklung eines externen Sensornetzes mit WLAN Kopplung und Visualisierung

STUDIENARBEIT
Im Rahmen der vierten Praxisphase

Abgabedatum Irgendwann 2015

Verfasser
Matrikelnummer
Kurs

Maik Maier, Nicolai Staeger
4050846, 4615051
TINF12B3

Erklärung

Gemäß §5 (3) der „Studien- und Prüfungsordnung DHBW Technik“ vom 22. September 2011.

Ich habe die vorliegende Studienarbeit selbstständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel verwendet.

Karlsruhe, Datum

Unterschrift

Inhaltsverzeichnis

1	Einleitung und Intention	1
1.1	Ziel dieser Arbeit	2
2	Internet of Things	3
2.1	Geschichte	4
2.2	Ziele	4
2.3	Do It Yourself - Trend	5
2.4	Sicherheitsaspekte	5
3	Theoretische Grundlagen	6
3.1	Wireless Sensor Networks	6
3.1.1	Ubiquitäres Rechnen	6
3.1.2	Motivation von Sensornetzen	7
3.1.3	Bestandteile	9
3.1.4	Topologien	9
3.1.5	Schwierigkeiten	11
3.1.6	Adhoc-Netzwerke	11
3.1.7	IEEE 802.15.4	11
3.2	SunSPOT	11
3.2.1	Technische Daten	12
4	Praktische Arbeiten mit SunSPOT	14
4.1	Erste Schritte	14
4.2	Implementierung einer Raumüberwachung	14
4.2.1	Idee	14
4.2.2	Visualisierung der Information	14
5	Zusammenfassung	15
	Literaturverzeichnis	vii

Abbildungsverzeichnis

3.1	Mikroprozessoren-Transistoren im Laufe der Zeit [Wgs]	8
3.2	Peer-To-Peer Netzwerk [Kos09]	9
3.3	Stern Netzwerk [Kos09]	10
3.4	Baum Netzwerk [Kos09]	10
3.5	Vermashtes Netzwerk [Kos09]	11
3.6	Anatomie eines Standard SunSPOT-Sensors [Uni]	12

Tabellenverzeichnis

Listings

Abkürzungsverzeichnis

IoT	Internet of Things
SPOT	Small Programmable Object Technology
I/O	Input/Output
USB	Universal Serial Bus
MHz	Megahertz
CPU	Central Processing Unit
SRAM	Static Random Access Memory
G	G-Kraft - Gewichtskraft
IEEE	Institute of Electrical and Electronics Engineers
USART	Universal Asynchronous Receiver Transmitter
I²C	InterIntegrated Circuit
mA	Milli-Ampere
EEPROM	Electrically Erasable Programmable Read-Only Memory
LED	Licht-emittierende Diode
RGB	Rot, Grün und Blau

Kapitel 1

Einleitung und Intention

Der Computer ist mittlerweile zum festen Bestandteil im alltäglichen Leben geworden. Mit ihm können viele Aufgaben wie Recherchen, komplexe Rechnungen und Kommunikation vereinfacht und schnellstmöglich erledigt werden. Während vor einigen Jahren noch der Desktop-PC die beliebteste Wahl darstellte, geht der Trend mittlerweile in Richtung der mobilen Endgeräte wie z.B. Smartphone, Laptop oder auch Tablet. Menschen wollen sich nicht an einen Ort binden, an dem sie ihren Computer benutzen können und sehnen sich nach dem Wunsch, dass alle Alltagsgegenstände per Smartphone oder Tablet kontrollierbar werden.

Diese Vernetzung aller elektronischen Geräte in einem Haushalt wird als “Internet of Things” (kurz IoT) bezeichnet. Die grundsätzliche Idee besteht darin, dass alle elektronischen Geräte wie z.B. Kühlschrank, Backofen u.a. miteinander kommunizieren können und der Nutzer über sein mobiles Endgerät alle Daten der vernetzten Geräte einsehen und diese auch auf seinen Wunsch hin steuern kann. Nähere Informationen zu IoT folgen im nächsten Kapitel.

Zur beispielhaften Demonstration des Aufbaus eines solchen Netzes elektronischer Geräte beschäftigt sich diese Studienarbeit mit Oracle SunSpot, einem Sensornetzwerk bestehend aus 2 Sensoren und einer Basisstation. Im Folgenden wird die Inbetriebnahme und Programmierung dieser Sensoren vorgenommen und die darin enthaltene Technik erklärt. Ziel der Studienarbeit ist es, mit Hilfe von SunSpot eine rudimentäre Raumüberwachung zu programmieren, indem bewegte Fenster oder Türen bei Abwesenheit des Besitzers der Wohnung erkannt werden, die Basisstation die Werte sammelt und sie an den Besitzer meldet.

1.1 Ziel dieser Arbeit

Hier werden wir das Ziel dieser Arbeit sowie das erwartete Ergebnis niederschreiben. Im Fazit kann hierauf Bezug genommen werden, um rückblickend den Erfolg dieser Arbeit zu messen.

Kapitel 2

Internet of Things

Als im Februar 1946 ENIAC, der erste elektronische sowie programmierbare Universalrechner vorgestellt wurde, wog dieser 27 Tonnen und füllte einen gesamten Raum. Für private Anwendungen waren diese Rechnersysteme nicht geeignet. Mit der voranschreitenden Entwicklung werden Computer immer kleiner und leistungsfähiger. Es erschließen sich immer neue Anwendungen von Computersystemen, die hauptsächlich den Menschen in seinem Alltagsleben unterstützen sollen.

Rund um 1990, als das Internet kommerzialisiert und somit für jeden zugänglich wurde, begann eine rasante Entwicklung neuer Technologien. Bis heute hat es unsere Arbeitsweise sowie unser Privatleben verändert und dieser Trend schreitet ungebremst voran.

Mit dem Web 2.0 wurden Webseiten interaktiv sowie Videos und Bilder im Internet eine Selbstverständlichkeit. Soziale Netzwerke verbinden die Nutzer durch verschiedenste Arten der Kommunikation. Dieses sich immer weiter aufspannende Kommunikations- und Informationsnetz, erreicht nun auch unsere kleinsten elektronischen Geräte.

Das Haus wird durch ein komplexes Sicherheitssystem überwacht, die Tür benötigt nur den Fingerabdruck um sich automatisch zu öffnen, der Fernseher reagiert auf Spracheingaben und in der Zukunft erstellt der Kühlschrank autonom den Einkaufszettel.

All diese Informationen werden über das Internet zu einer zentralen Sammelstelle oder dem Menschlichen Akteur zugespielt, das Internet of Things (IoT) ist entstanden.

2.1 Geschichte

Etwa um 1982 ärgerten sich drei Studenten der School of Computer Science, an der Carnegie Mellon University über den Getränkeautomaten ihres Instituts. An manchen Tagen liefen sie zu dem Automaten und erhielten entweder keine Getränke, oder zu warme, da diese erst kürzlich nachgefüllt wurden. Um diese Problematik zu lösen entwickelten die Studenten John Zsarnay neue Hardware die mit Software von David Nichols und Ivor Durham die Füllstände, sowie die Temperatur der Getränke überwachte [Car98].

Über den damals an der Universität vorhandenen Vorgänger des Internets, das sogenannte Arpanet konnte direkt beim Automaten der aktuelle Status nachgefragt werden. Dieser antwortete zum Beispiel mit:

1	>	EMPTY	EMPTY	1h 3m
2	>	COLD	COLD	1h 4m

Hiermit informierte der Automat darüber, dass kalte Getränke in der mittleren sowie linken Schiene vorhanden seien, die Getränke der rechten Schiene jedoch noch warm seien. Die angegebene Zeit informierte darüber, wie lange die Getränke sich bereits im Automat befanden. Nach drei Stunden nahm der Automat an Getränke seien ausreichend gekühlt.

Bereits 1991 schrieb der Amerikanische Informatiker Mark Weiser eine Vision, wie technische Geräte der Zukunft untereinander vernetzt sein könnten [Wei91]. Den Namen IoT erhielt das ganze jedoch erst 1999.

2.2 Ziele

Hierin kommen all die Ziele des IoTs.

- Umweltsensoren (Temperatur Feuchtigkeit Erschütterung Lautstärke Luftzusammensetzung)
- Lichtsteuerung
- Haushaltshilfen
- Bestandsaufnahme / Nachfuhrkontrolle
- Überwachungsfunktionen

- SSmart Signs Autobahn
- Entertainment
- Haussteuerung
- Prozessüberwachung (Ventile Flussraten usw.)
- Diagnose / Lebensüberwachung usw.

2.3 Do It Yourself - Trend

Erklärung: Trend sachen selbst zu lösen.
Hier Hilft Architecting ab Seite 70.

Vielleicht LEGO-Mindstorms

Arduino

Raspberry

2.4 Sicherheitsaspekte

Hier wollen wir auf Sicherheitsaspekte des IoTs eingehen. Möglicherweise die Absicherung gegen Attacken von außen oder Ähnliches erläutern.

Kapitel 3

Theoretische Grundlagen

In diesem Kapitel erklären wir kurz, dass wir in den folgenden Unterkapiteln Grundlagen erklären. Worum es sich hierbei handelt können Sie den folgenden Kapiteln entnehmen.

3.1 Wireless Sensor Networks

Hier wird alles zu Wireless Sensor Networks erklärt. Bisher steht das erst als Gerüst.

3.1.1 Ubiquitäres Rechnen

1988 verwendete Mark Weiser erstmals den Begriff 'ubiquitous computing' (dt. ubiquitäres Rechnen), um seine Vision nach einem stets verfügbaren Rechensystem, welches dem Nutzer unsichtbar erscheinen soll, zum Ausdruck zu bringen. Der Computer soll sich so in den Alltag integrieren, dass die Menschen ihn gar nicht mehr bemerken. Nach seiner Vorstellung verbessere das ubiquitäre Rechnen die Erfahrungen, die man mit Computern macht, da die Rechner dem Nutzer nahtlos verfügbar gemacht werden, ohne dabei effektiv sichtbar zu sein.

Weiser zufolge sind die besten Technologien diejenigen, die scheinbar verschwinden, tatsächlich jedoch nur in den Hintergrund geraten und unsichtbar werden. Der Mensch soll nicht in der Welt des Computers leben, sondern der Rechner soll sich in die Welt des Menschen integrieren. In Lichtschaltern, Thermostaten, Stereoanlagen und Backöfen werden bereits heute kleine Rechner verbaut, die helfen sollen, den Alltag zu erleichtern und die Idee des 'Internet of

Things' weiter zu verfolgen.

Da Ubiquitäres Rechnen zuverlässig und unsichtbar funktionieren soll, ist die Technologie der unsichtbaren Rechenmodule von großer Bedeutung. Voraussetzungen sind z.B. leistungsstarke Prozessoren, ausreichend Speicherplatz, drahtlose Kommunikation, Sensoren und Aktoren (die z.B. mit der Umwelt und dem Menschen interagieren). Der Mensch muss nicht für alle Anwendungsfälle von ubiquitärem Rechnen direkt eingebunden werden, da die Systeme auch autonom arbeiten können [WB12].

3.1.2 Motivation von Sensornetzen

Sensornetze sind sehr flexibel und können unter anderem dafür eingesetzt werden, um

- Umwelteinflüsse wahrzunehmen ('sensing')
- Umwelteinflüsse zu verarbeiten und zu analysieren ('computing')
- Daten zu übertragen ('transport')
- Netzwerke für verteilte Systeme aufzubauen ('networking')
- Die Umwelt zu beeinflussen und zu verändern ('actuation')

Für viele Anwendungsfälle und Szenarien, in denen mit der Umwelt interagiert wird, soll die Benutzung von drahtlosen Sensornetzen zukünftig ausgebaut und etabliert werden. Der Einsatz von Sensornetzen kann dabei verschiedene Motivationen und Anforderungen haben:

- Direkte Interaktion mit Menschen ist nicht möglich oder nicht erforderlich (z.B. bei Überwachung einer Maschine in der Industrie)
- Der Mensch soll nur im Notfall alarmiert werden (z.B. in Notfällen oder wenn die Sensoren bestimmte Schwellenwerte erreichen)
- Es handelt sich um ein autonomes System, welches nur Selten das Handeln eines Menschen erfordert

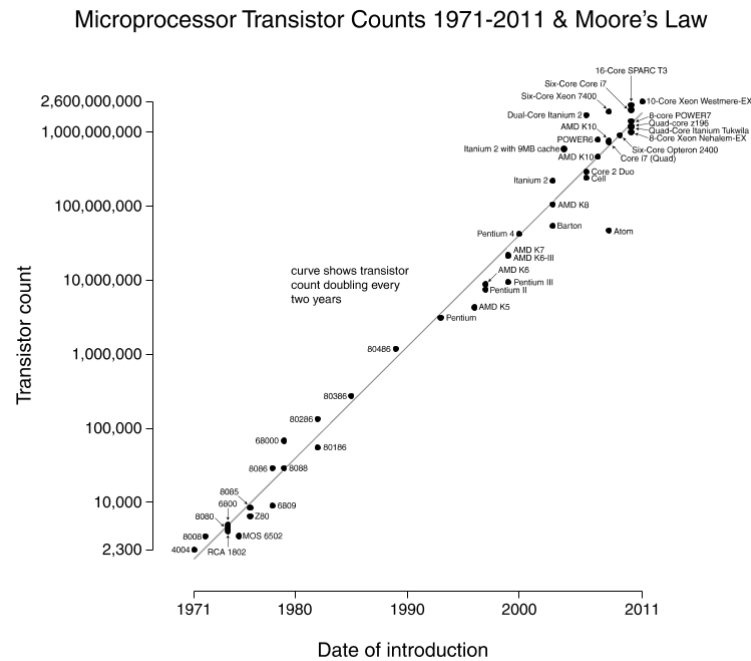


Abbildung 3.1: Mikroprozessoren-Transistoren im Laufe der Zeit [Wgs]

Eine weitere Motivation der Verwirklichung von drahtlosen Sensornetzen ist der Technologiefortschritt, der es möglich macht, immer kleinere Rechengeräte mit mehr Leistung herzustellen und miteinander zu vernetzen. Um diesen Fortschritt zu verdeutlichen, formulierte Gordon Moore 1965 ein Gesetz, welches besagt, dass sich die Anzahl der integrierten Schaltkreise auf einem Mikroprozessor alle 18-24 Monate verdoppelt. Im Gegensatz zu den Anzahl der Schaltkreise steigt die Rechenleistung der Prozessoren allerdings nicht linear an, da immer mehr Schaltkreise für den Cache des Prozessors verwendet werden, was nur geringfügig der Leistungssteigerung dient. Das Ende der Gültigkeit des Mooreschen Gesetzes wurde schon des öfteren wegen unüberwindbarer technischer Grenzen vorausgesagt, diese wurden jedoch bisher alle mit dem Einsatz neuer technischen Mittel und Materialien überwunden. Momentan schätzt der Halbleiterhersteller Intel, welcher 1968 von Moore mitbegründet wurde, dass das Mooresche Gesetz noch mindestens bis 2023 seine Gültigkeit behält. Mittlerweile existieren beim Konzern sogar explizite Pläne, die das Einhalten des Mooreschen Gesetzes sicherstellen sollen. [WB12] [Kah12] [Tuo02].

3.1.3 Bestandteile

3.1.4 Topologien

Bei einem Aufbau eines Sensornetzes stellt sich grundsätzlich die Frage, wie die einzelnen Sensoren miteinander Verbindungen aufbauen und kommunizieren sollen. Ein solche Verbindungsstruktur nennt sich in der Informatik 'Topologie'. Da das Sensornetz insgesamt zuverlässig arbeiten soll, Kosten und Komplexität jedoch gering gehalten werden sollen, wurden speziell für die drahtlosen Sensornetze neue Ansätze im Bereich der Topologie erforscht. Im folgenden sollen 4 Topologie-Alternativen näher erläutert werden.

Peer-to-Peer Netzwerke erlauben es, das jeder Knoten im Netz (in unserem Fall der Sensor) mit jedem anderen Knoten direkt Kontakt aufnehmen kann. Jedes 'Peer-Gerät' ist gleichzeitig Client und Server gegenüber anderen Knoten im Netzwerk.

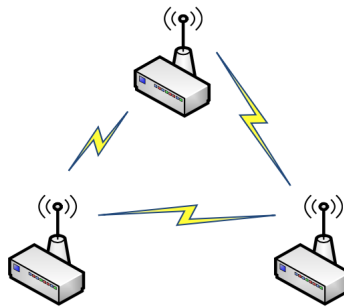


Abbildung 3.2: Peer-To-Peer Netzwerk [Kos09]

Bei der Stern-Topologie sind die Sensoren an ein zentrales Kommunikationsgerät angebunden. In diesem Fall kommunizieren die einzelnen Knoten nicht direkt miteinander. Jegliche Art von Kommunikation wird über das zentrale Gerät (auch Hub genannt) geroutet. Der Hub wird hier als Server betrachtet, wohingegen die Knoten (Sensoren) die Clients darstellen.

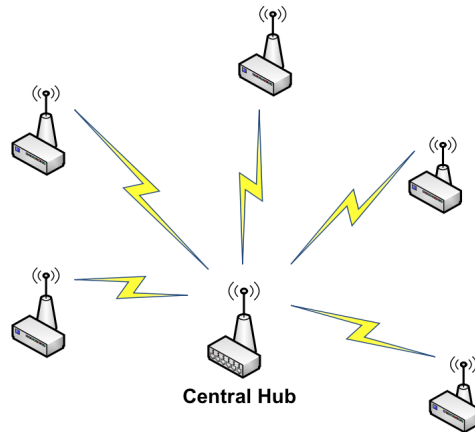


Abbildung 3.3: Stern Netzwerk [Kos09]

Die Baum-Topologie stellt eine Hybridvariante aus Peer-to-Peer und Stern dar. Sie nutzt einen sogenannten 'Root-Knoten' als zentraler Router. Eine Ebene darunter liegen die Hubs, an denen wie in der Stern-Topologie die Sensoren angebunden sind.

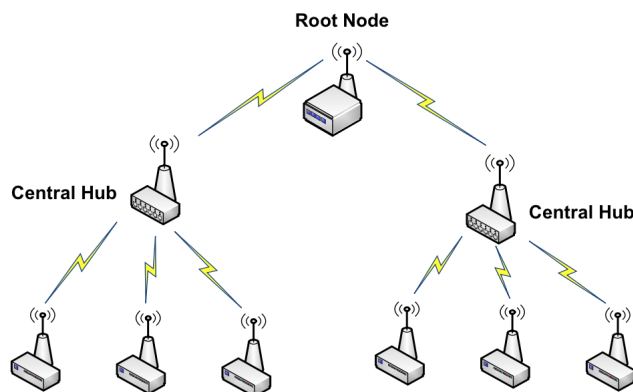


Abbildung 3.4: Baum Netzwerk [Kos09]

Eine weitere Mögliche Variante ist ein vermaschtes Netz. Die Knoten sind untereinander ohne zentralen Hub verbunden und die Daten werden einfach von Knoten zu Knoten weitergesendet, bis sie ihr gewünschtes Ziel erreicht haben [Kos09].

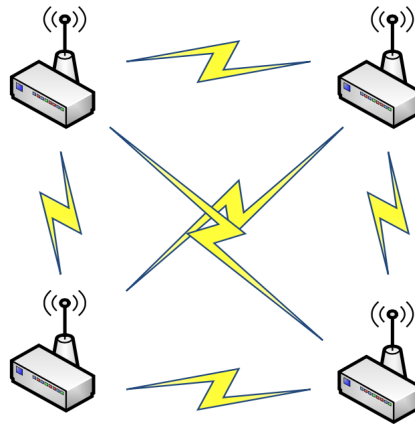


Abbildung 3.5: Vermashtes Netzwerk [Kos09]

3.1.5 Schwierigkeiten

3.1.6 Adhoc-Netzwerke

Inhalt kommt hier auch noch rein.

3.1.7 IEEE 802.15.4

Und hier kommt ebenfalls noch Inhalt rein.

3.2 SunSPOT

Die Firma Oracle besitzt im Rahmen seiner Java-Technologie eine Vormachtstellung im Bereich der Smartphones. Auf der Welt sind schätzungsweise über eine Milliarde Smartphones mit der Java-Technologie lizenziert. [Hor08]

Ziel von Oracle ist es, auch in den zukunftsnahe Technologien mit ihrer Programmiersprache Java auszustatten und diese Produkte zu etablieren.

Ein erster Schritt in diese Richtung ist das von Oracle entwickelte “SunSPOT“-Sensornetzwerk. SunSPOT bedeutet “Sun Small Programmable Object Technology“ und ist eine Plattform für Java-basierte drahtlose Sensornetze. Sie bestätigt den Trend, dass in immer kleiner werdenden Geräten zunehmend leistungsfähigere Technologien eingesetzt werden. Dabei ist wichtig, dass jene Geräte, am Besten drahtlos, miteinander kommunizieren können und jederzeit von überall auf der Welt steuerbar bleiben. Das SunSPOT Starter Paket besteht aus einer

Basisstation und 2 Sensoren.

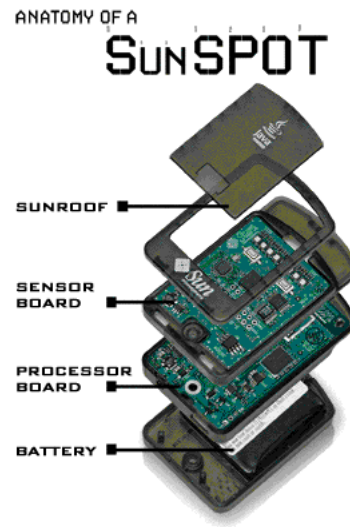


Abbildung 3.6: Anatomie eines Standard SunSPOT-Sensors [Uni]

Die Hardware der SunSPOT-Sensoren ist modular aufgebaut. Das bedeutet, dass man die verfügbaren Boards frei nach Belieben aufeinander stecken und somit verbinden kann. Dabei können maximal bis zu 3 Boards + Stromversorgung miteinander verknüpft werden. [Hor08]

3.2.1 Technische Daten

Das sogenannte eSPOT Prozessor-Board besitzt in der aktuellsten Version eine 400 MHz 32-bit ARM CPU von Atmel, zusammen mit einem Flashspeicher von 8 Megabytes und einem Megabyte SRAM Hauptspeicher. Weiterhin ist es ausgestattet mit einem Radio Transceiver basierend auf IEEE 802.15.4 und einer USB 2.0 - Full Speed Schnittstelle. Der im SunSPOT integrierte Akkumulator hat eine Leistungsfähigkeit von 770mAh. Der maximale Energieverbrauch liegt zwischen 40-100 mA, abhängig von der Nutzung der integrierten LEDs, des Transceivers und anderer angeschlossener Geräte. [Hor08] [Ora10b]

Der SunSPOT-Sensor wird dazu standardmäßig mit dem eDemo Sensor Board ausgeliefert. Dieses Board besitzt in der aktuellen Version einen 2G/4G/8G 3-Achsen-Beschleunigungssensor, einen Lichtsensor, 8 RGB 24bit LEDs, einen Infrarot-Sender & Empfänger, ein kleiner Lautsprecher, 2 Knopfschalter, 4 analoge Eingänge, 4 I/O Pins, diverse weitere I²C- und USART-Interfaces, einen EEPROM und 4

100mA Ausgangspins, mit denen es möglich ist, den SunSPOT-Sensor z.b. an weitere Lautsprecher oder andere Geräte anzuschließen. [Hor08] [Ora10a]

Weitere Boards, welche man nach Bedarf dazustecken kann, sind das eProto-Board, ein Board welches direkte Zugriffe auf das Prozessorboard ermöglicht und einen SD-Kartenslot besitzt, damit man die Daten dauerhaft speichern kann, das eSerial Board zum Verbinden via RS232 und das eFlash SD-Kartenleser Board. [Hor08]

Kapitel 4

Praktische Arbeiten mit SunSPOT

Das ist das normale Todo, inline.
Mit dem `\newline` Befehl erzwingt man eine neue Zeile.

In diesem Kapitel werden wir die praktischen Arbeiten, die wir mit SunSPOTS durchführen erläutern. Dies wird in zwei Unterkapiteln durchgeführt. Man sollte damit das hier besser aussieht auch viel Text haben. Wenn nur eine Zeile unterstrichen ist sieht das nicht ganz so gut aus.

Also das hier ist ein Improvement und ist einfach in einer anderen Farbe.

Das nach dem `\unsure` Befehl folgende wird bis zum Zeilenende rot unterstrichen.

4.1 Erste Schritte

Hier erläutern wir unsre ersten Schritte im Praktischen Teil der Studienarbeit.

4.2 Implementierung einer Raumüberwachung

4.2.1 Idee

Hier kommt die Idee die wir hatten hin.

4.2.2 Visualisierung der Information

Beschreibung der Visualisierung unserer durch das Sensornetz gesammelter Informationen.

Kapitel 5

Zusammenfassung

Hier kommt die Zusammenfassung des Projektes hin. Diese besteht aus Beschreibung, Vorgehensweise und Ergebnis. Insgesamt umfasst sie etwa eine Seite.

Anhang

Beispielerggebnis Bedienungsanleitungen

Manchmal benutzt man Worte wie Hamburgetypes, Raftenducks oder Handgloves, um Schriften zu testen. Manchmal Sätze, die alle Buchstaben des Alphabets enthalten - man nennt diese Sätze »Pangrams«. Sehr bekannt ist dieser: The quick brown fox jumps over the lazy old dog. Oft werden in Typoblindtexte auch fremdsprachige Satzteile eingebaut (AVAIL® and Wefox™ are testing aussi la Kerning), um die Wirkung in anderen Sprachen zu testen. In Lateinisch sieht zum Beispiel fast jede Schrift gut aus. Quod erat demonstrandum. Seit 1975 fehlen in den meisten Testtexten die Zahlen, weswegen nach TypoGb. 204 § ab dem Jahr 2034 Zahlen in 86 der Texte zur Pflicht werden.

Literaturverzeichnis

- [Car98] CARNEGIE MELLON UNIVERSITY COMPUTER SCIENCE DEPARTMENT: *The Only Coke Machine on the Internet*. https://www.cs.cmu.edu/~coke/history_long.txt, June 1998
- [Hor08] HORAN, Bernard: *Sun SPOTs*. <https://www.dropbox.com/sh/12kch3izg7lwdpl/AADjbAi2ukAjtaZY8zVzMpdHa/IoT/sunspot.pdf>, 2008
- [Kah12] KAHLE, Christian: *Intel: Mooresches Gesetz gilt noch mind. 10 Jahre*. <http://winfuture.de/news,72001.html>, September 2012
- [Kos09] KOSMERCHOCK, Steven: *Wireless Sensor Network Topologies*. http://www.k5systems.com/TP0001_v1.pdf, 2009
- [Ora10a] ORACLE CORP.: *SunTM SPOT eDEMO Technical Datasheet Rev 8.0*. <http://www.sunspotworld.com/docs/Yellow/edemo8ds.pdf>, Oktober 2010
- [Ora10b] ORACLE CORP.: *SunTM SPOT Main Board Technical Datasheet Rev 8.0*. <http://www.sunspotworld.com/docs/Yellow/eSPOT8ds.pdf>, Oktober 2010
- [Tuo02] TUOMI, Ilkka: *The lives and deaths of moores law*. <http://firstmonday.org/ojs/index.php/fm/article/view/1000/921>, November 2002
- [Uni] UNIVERSITY OF SOUTHERN CALIFORNIA: *Standardmäßiger Aufbau eines SunSPOT-Sensors*. http://anrg.usc.edu/ee579_2012/Group07/img/spotanatomy.jpg,
- [WB12] WOLF, Lars ; BÜSCHING, Felix: *Wireless Sensor Networks - Introduction and Applications*. <https://www.dropbox.com/sh/12kch3izg7lwdpl/AABu1b-vt8FCuUD2iU905F0ca/IoT/>

RecentTopics_Chapter02_WSN-Introduction-and-Applications.pdf, 2012

[Wei91] WEISER, Mark: *The Computer for the 21st Century*. <http://www.ubiq.com/hypertext/weiser/SciAmDraft3.html>, September 1991

[Wgs] WGSIMON: *Moore'sches Gesetz*. http://commons.wikimedia.org/wiki/User:Wgsimon#mediaviewer/File:Transistor_Count_and_Moore%27s_Law_-_2011.svg,

Notes

■ Erklärung: Trend sachen selbst zu lösen.	
Hier Hilft Architecting ab Seite 70.	5
■ Vielleicht LEGO-Mindstorms	5
■ Arduino	5
■ Raspberry	5
■ Das ist das normale Todo, inline.	
Mit dem <code>\newline</code> Befehl erzwingt man eine neue Zeile.	14
■ Das nach dem <code>\unsure</code> Befehl folgende wird bis zum Zeilenende rot unterstrichen.	14
■ Also das hier ist ein Improvement und ist einfach in einer anderen Farbe.	14