1. Einleitung

Motivation, Zielstellung, Vorteile

2 Theoretische Grundlagen

2.1 Serielle Kommunikation/COM-Schnittstelle

2.2 RS232

2.2 Datenbanken – SQL -> NoSQL als Alternative erwähnen

2.3 Datenstacks/Datenfluss

3 Konzeption

3.1 Serverseitig

3.1.1 Datenbankstruktur

3.1.2 Webinterface

3.1.3 Zugangsbeschränkung und Rechteverwaltung

3.2 Clientseitig

3.2.1 Benutzeroberfläche

3.2.2 Serielle Schnittstelle zur Waage

3.2.3 Datenbankschnittstelle

4 Implementierung

4.1 Serverseitig

4.1.1 Datenbankstruktur

4.1.2 Nutzeroberfläche

4.1.3 Zugangsbeschränkung und Rechteverwaltung

4.1.4 Erweiterungen

4.2 Clientseitig

4.2.1 Interface

4.2.2 Serielle Schnittstelle zur Waage

4.2.3 Datenbankschnittstelle

4.2.4 Erweiterungen

5 Produktion

5.1 Testprozeduren

5.2 Datensicherheit

6 Zusammenfassung

2. Zielstellung: Live-Tracking zur Erkennung von Engpässen, statistische Auswertung vom Verbrauch,

Allg. Hinweise:

Genauere Überschriften

Produktion: Verlauf der Entwicklung aus Entwicklungsumgebung -> Testumgebung -> Produktionsumgebung; Überwachung und Backups

# 1 Serielle Kommunikation

Für den modernen Betrieb von industriellen Geräten ist die Kommunikation zwischen Gerät und einer Steuereinheit, meist ein Rechner, oft unumgänglich. Es werden dabei Messwerte oder Statusinformationen des Gerätes an den Rechner und Befehle von dem Rechner auf das Gerät übertragen. Diese Daten bestehen aus einem Binärcode, also der Verkettung von binären Signalen mit definierter Struktur. Der Binärcode repräsentiert dabei häufig Zeichen oder ganze Zeichenketten, die übertragen werden sollen. [HULZEBOSCH, J.]

Die einzelnen binären Signale werden in der Informatik als Bits bezeichnet. Diese Bits können die logischen Zustände 0 oder 1, bzw. high oder low, annehmen. Jeweils 8 Bits werden zu einer geschlossenen Informationseinheit, dem Byte, zusammengefasst. Dadurch ergeben sich für jedes Byte 256 mögliche Zustände zwischen 0 und 255. Nach dem 8-Bit-Code der DIN 66303 kann jedes Byte ein Schriftzeichen, beispielsweise Buchstaben oder Zahlen, aber auch Sonderzeichen, repräsentieren. [DIN66303; HULZEBOSCH, J.]

Die Bytes können entweder seriell oder parallel übertragen werden. Bei der parallelen Übertragung werden alle Bits eines Bytes gleichzeitig übertragen. Voraussetzung hierfür ist eine Verbindung mit mindestens einem Kabel pro übertragenem Bit und weiteren Kabeln für Steuersignale. Weiterhin müssen die Kabel mit zunehmender Übertragungsgeschwindigkeit immer höheren Qualitätsanforderungen entsprechen, um die Datenintegrität sicherzustellen. So müssen beispielsweise die Impendanzen aller Kabel übereinstimmen, um Störungen zu vermeiden, und Längenunterschiede in den Verbindungen können die zeitliche Abstimmung der Signale stören. Weiterhin kommt es durch die langen, parallel liegenden Leitungen und schnell wechselnden Spannungen zu Induktionserscheinungen, die als „crosstalk“ bezeichnet werden. Dieser Effekt senkt durch wechselhafte Störungen die Übertragungsqualität. Gerade bei hohen Übertragungsraten, im Bereich von Gb/s, ist die maximale Übertragungsweite deshalb auf wenige Zentimeter begrenzt. Durch die fest verbaute Anzahl von Kabeln ist auch die Übertragung größerer Datenpakete, zum Beispiel von zusätzlichen Paritätsbits oder Datenpaketen mit mehr als 8 Bits, nachträglich nicht mehr möglich. Die parallele Kommunikation wird deshalb in modernen Systemen vorrangig für die Übertragung von Daten innerhalb eines Rechners oder innerhalb einer Leiterplatte verwendet.

Die zweite Möglichkeit zur Datenübertragung ist die serielle Kommunikation. Dabei werden die Bits eines Bytes sequentiell über ein einzelnes Kabel versendet. Für die einfachste serielle Kommunikation ist dementsprechend nur in einzelnes Datenkabel und ein Kabel für die gemeinsame Masse, als Referenz, nötig. Der technische Aufwand für die Hardware ist also deutlich geringer als für die parallele Übertragung. Andererseits ist die Übertragungsgeschwindigkeit geringer, da die Bits nacheinander gesendet werden. Für normale Anwendungen ist dieser Verlust an Übertragungsgeschwindigkeit kaum relevant, da trotzdem Übertragungsraten von mehreren Gb/s erreicht werden können. Es können auch hybride Methoden verwendet werden, bei denen mehrere serielle Verbindungen parallel Teile der Daten übertragen.

Bei der Übertragung der Bits müssen die logischen Werte, 1 oder 0, durch Spannungslevel dargestellt werden. Die verwendeten Verfahren werden Leitungscodes genannt. Ein einfaches Verfahren ist das bipolare Non-Return-to-Zero (NRZ), bei dem eine 1 durch einen positiven Pegel und eine 0 durch einen negativen Pegel ausgedrückt wird. Der Zustand wird für die gesamte Dauer des Bits gehalten und kehrt nicht auf 0 zurück. Das Spannungsverhalten, abhängig von den zu übertragenden logischen Werten, ist in Abbildung 1 dargestellt. Die verwendeten Spannungslevels sind in den jeweiligen Standards der seriellen Schnittstellen festgelegt.



Abbildung 1: Spannungsverhalten von bipolar NRZ-codierten binären Werten

Die Verwendung von bipolaren Codierungen erbringt den Vorteil, dass sich keine mittlere Gleichspannung in der Leitung aufbaut, wie es bei der Verwendung von unipolaren Codierungen, durch den schnellen Wechsel von mehreren positiven Spannungen, der Fall ist. Diese Störspannung würde die Übertragungsqualität senken und damit die Übertragungsrate limitieren. Ein Nachteil der NRZ-Methode ist, dass bei der Übertragung langer Folgen des gleichen Signals die einzelnen Signale nicht klar voneinander getrennt werden können und somit Unsicherheiten über die Anzahl der gesendeten Signale entstehen. Ein bekannter Standard auf Basis dieses Verfahrens ist der EIA-232, besser bekannt als RS-232. [Appl. Note 83]

Es existieren verschiedene andere Verfahren zur Umsetzung von logischen Werten in Spannungslevel, wie das Manchestercoding oder die Return-to-Zero-Methoden (RZ). Bei dem Manchestercoding werden beide logischen Zustände durch ein positives und ein negatives Signal von jeweils halber Bitdauer repräsentiert. Eine 1 wird als Abfolge von Positiv-negativ und eine 0 als negativ-positiv dargestellt. Dadurch kann die Bildung eines Gleichstrompotentials in den Leitungen verhindert werden und Bits können klar voneinander getrennt werden. Nachteil dieser Methode ist, dass aufgrund der Verwendung von 2 Pegeln pro Bit die Frequenz der Pegelwechsel etwa doppelt so hoch ist wie bei Methoden mit konstanten Pegeln und dementsprechend die Taktraten für die Datenübertragung begrenzt sind. Die Return-to-Zero-Methoden verwenden ebenfalls bis zu zwei Spannungslevels pro Bit, dabei ist der erste Pegel der, für den logischen Zustand, charakteristische Pegel und der zweite Pegel ist 0. Die Spannungslevels werden jeweils für die Hälfte der Übertragungsdauer jedes Bits gehalten. Die Manchester- und RZ-Verfahren werden vorrangig für die möglichst fehlerfreie und kontinuierliche Übertragung von Daten mit geringerer Geschwindigkeit verwendet. [Frenzel, Handbook]

Bei der Übertragung von seriellen Daten ist weiterhin die Art der Verbindung und in diesem Zusammenhang die Wahl eines passenden Übertragungsmedium zu beachten. Es kann in symmetrische und asymmetrische Verbindungen unterschieden werden. Das Prinzip beider Verbindungsarten sind in Abbildung 2 dargestellt.



Abbildung : Prinzip der asymmetrischen (a) und symmetrischen (b) Verbindung als Anwendung in der Tontechnik [http://hbernstaedt.de/Test/Asm\_Asm.jpg]

Die asymmetrischen Verbindungen führen neben der Datenleitung eine weitere Leitung für die gemeinsame Masse mit. Diese Masseleitung wird als Referenz für das Spannungslevel im unbeschalteten Zustand, also als 0 V, verwendet. Die symmetrische Verbindung verwendet ein weiteres Datenkabel mit invertierter Polarität. Als Signal wird bei der symmetrischen Verbindung die Differenz der, auf den beiden Datenkabeln übermittelten, Spannungslevel erfasst. Der in der Abbildung gezeigte Störfall, beispielsweise durch elektromagnetische Felder in der Umgebung, wirkt sich in der asymmetrischen Verbindung direkt auf das Ausgangssignal auf. In der symmetrischen Verbindung tritt die Störung in beiden Datenleitungen auf und kann von dem Empfänger durch die Ermittlung der Signaldifferenz korrigiert werden. Typisch sind das Koaxialkabel als Medium der asymmetrischen Verbindung und der Cannon-Stecker als Träger von symmetrischen Signalen. [Frenzel, Handbook]

NRZ, DIN, RZ

## 1.1 RS-232

RS-232 wurde als Standard für die serielle Direktverbindung zwischen einem Computer oder Terminal (Data Terminal Equipment, DTE) und einem Peripheriegerät (Data Circuit-Terminating Equipment, DCE) entwickelt. Ausgehend von dem Entwickler, Electronic Industry Association, lautet die ursprüngliche Bezeichnung EIA-232. Aufgrund der großen Popularität dieses Standards ist die verbreitetere Bezeichnung RS-232 für „Recommended Standard“. Der offizielle Name für den Standard lautet „Interface between Data Terminal Equipment and Data Circuit-Termination Equipment Employing Serial Binary Data Interchange“ [Appl. Note 83].

Der Standard wurde 1962 erstmalig vorgestellt und war für die Kommunikation von Rechnern mit Modems ausgelegt. Aufgrund der Einfachheit und Robustheit des Standards und der wachsenden Anzahl von PCs mit entsprechender Schnittstelle entwickelte sich dieser Standard schnell zu dem verbreitetsten Standard für serielle Kommunikation und fand unter anderem in einigen der ersten Mäuse, Plotter, Scanner und vor allem den meisten industriellen Geräten Anwendung. Der Standard wurde mehrfach überarbeitet, die aktuellste Revision ist der EIA-232-F von 1997. Nach der Einführung des USB-Standards wurde der RS-232 weitgehend aus privaten Anwendungen verdrängt, stellt aber für die Steuerung der meisten industriellen Geräte weiterhin das verbreitetste Protokoll dar. [Texas Instr. Appl. Note]

In dem Standard wurde als mechanisches Interface der 25-polige D-Sub-Stecker mit definierten Maßen und Pinbelegungen festgelegt. Mit dieser Anzahl Pins können alle in dem Standard festgehaltenen Signale und Funktionen bedient werden. Da für die meisten Anwendungen nicht die volle Ausnutzung aller möglichen Signale nötig ist, wird aus Platzgründen häufig der 9-polige D-Sub-Stecker DB9 verwendet. Die Belegung des DB9-Steckers ist in Abbildung 3 gezeigt. [Texas Instr. Appl. Note]



Abbildung 3: Pinbelegung des 9-poligen D-Sub-Steckers [https://www.db9-pinout.com/db9-pinout/db9-pinout.gif]

Pin 2 und Pin 3 sind die Leitungen zum Senden und Empfangen von Daten, Pin 5 bietet die Möglichkeit, eine gemeinsame Masseleitung aufzubauen. DTR und DSR auf Pin 4 und Pin 6 sind Signale zur Bereitschaft des Computers und des angeschlossenen Gerätes zur Datenübertragung. RTS und CTS sind Signale für die Koordination der Datenübertragung, um Datenverlust durch blockierte Leitungen zu vermieden. Das Protokoll, nach dem diese Signale den Datenfluss steuern, wird als Handshake bezeichnet. Das RI-Signal über Pin 9 war ursprünglich für die Übertragung des Klingel-Signals zwischen Modem und Telefon vorgesehen und ist für moderne Anwendungen kaum noch relevant. [Texas Instr. Appl. Note]

Beide Datenleitungen stellen eine asymmetrische Verbindung zwischen Sender und Empfänger dar, auf der Daten mit bipolarer NRZ-Codierung übertragen werden. Eine logische 0 wird von dem Sender durch eine Spannung zwischen 5 V und 25 V repräsentiert, eine logische 1 durch Spannungen zwischen -5 V und -25 V. Der Empfänger erkennt Signale zwischen 3 V und 25 V bzw. -3 V und -25 V, Spannungen zwischen -3 V und 3 V sind undefiniert. Dementsprechend beträgt der minimale Puffer für Störungen zwischen dem Sender und Empfänger 2 V. Anhand dieses Puffers wurde eine maximale Kapazität des Signalleiters mit 2500 pF festgelegt, was die maximale Leiterlänge auf etwa 15-20 m begrenzt. Als maximale Übertragungsrate wurden in dem Standard 20 kbit/s definiert, allerdings können in Anwendung Datenraten bis etwa 120 kbit/s erreicht werden. Diese Datenraten können allerdings nur mit deutlich kürzeren Datenkabeln und Treibern, die die benötigten Stromstärken für schnellere Spannungswechsel unterstützen, erreicht werden. [Texas Instr. Appl. Note]



Abbildung 4: Rahmenbildung nach RS-232 mit 8 Datenbits und einem Stopbit [Frenzel Handbook]

Um den, bei der NRZ-Codierung auftretenden, Verlust des Datentaktes durch lange Folgen gleicher Signale zu verhindern, werden die Bits bei diesem Standard in definierte Rahmen verteilt. Jeder Rahmen beginnt dabei, wie in Abbildung 4 gezeigt, mit einem Start-Bit, gefolgt von 5 bis 8 Datenbits und maximal 2 Stopbits. Optional kann ein Paritätsbit mitgeführt werden, um die Integrität der übertragenen Datenbits zu überprüfen. Jedes Signal, als Repräsentant eines Bits, wird dabei für eine festgelegte Dauer gesendet. Die Dauer des Signals hängt von der Baudrate der Verbindung ab. Dieser Wert gibt die maximale Anzahl von Signaländerungen pro Sekunde an und ist damit direkt proportional zur Übertragungsrate. Je höher die Baudrate, desto kürzer ist die Dauer jedes Bits. Die Baudrate, die Anzahl der Daten- und Stopbits, das verwendete Handshakeprotokoll sowie die Art des optionalen Paritätsbits sind Parameter, die bei Computer und Peripheriegerät übereinstimmen müssen, um eine Kommunikation zu ermöglichen, und beeinflussen die Übertragungsgeschwindigkeit der Verbindung. [Frenzel Handbook, Texas Instr. Appl. Note]

DCE, DTE, EIA, RS, PC, DTR, DSR, RTS, CTS, RI