

Technische Umsetzung eines Telebanking-Belegdruckers für Senioren/-innen

DIPLOMARBEIT

zur Erlangung des akademischen Grades

Diplom-Ingenieur

im Rahmen des Studiums

Medizinische Informatik

eingereicht von

Dipl.-Ing. Clemens Alexander Eisserer

Matrikelnummer 0625701

an der Fakultät für Informatik

der Technischen Universität Wien

Betreuung: Ao.Univ.Prof. Dipl.-Ing. Dr.techn. Wolfgang Zagler

Mitwirkung: Projektass. Dipl.-Ing. Christian Beck

Wien, 29. August 2016

Clemens Alexander Eisserer

Wolfgang Zagler

Erklärung zur Verfassung der Arbeit

Dipl.-Ing. Clemens Alexander Eisserer
Triesterstraße 68/4, 2620 Neunkirchen

Hiermit erkläre ich, dass ich diese Arbeit selbstständig verfasst habe, dass ich die verwendeten Quellen und Hilfsmittel vollständig angegeben habe und dass ich die Stellen der Arbeit – einschließlich Tabellen, Karten und Abbildungen –, die anderen Werken oder dem Internet im Wortlaut oder dem Sinn nach entnommen sind, auf jeden Fall unter Angabe der Quelle als Entlehnung kenntlich gemacht habe.

Wien, 29. August 2016

Clemens Alexander Eisserer

Danksagung

Vielen Dank an Prof. Dr. Wolfgang Zagler und Dipl.-Ing. Christian Beck für die professionelle Betreuung, die konstruktiven Ideen und Verbesserungsvorschläge sowie das aufgebrachte Verständnis für berufsbedingte terminliche Einschränkungen meinerseits.

Ebenfalls großen Dank möchte ich meinem Arbeitgeber, dem Austrian Institute of Technology (AIT), für die offene und fördernde Atmosphäre sowie die Flexibilität in Hinblick auf Arbeitszeitgestaltung aussprechen.

Mein größte Anerkennung gebührt meiner Familie für die Unterstützung während des Studiums. Ohne diese Unterstützung wäre mir der Abschluss meines Zweitstudiums nicht möglich gewesen.

Kurzfassung

E-Banking verzeichnet seit der breiten Markteinführung stetig steigende Akzeptanz und Nutzung [19], da es zahlreiche Vorteile, wie beispielsweise Ortsunabhängigkeit, ständige Verfügbarkeit sowie oftmals niedrigere Transaktionskosten bietet. Allerdings lässt sich gerade bei älteren Menschen eine deutlich geringere Inanspruchnahme von E-Banking-Diensten als bei jüngeren Altersschichten feststellen [18]. Dies kann unter anderem auf technische Einstiegsbarrieren, kritische Haltung gegenüber neuen Technologien, zu komplizierten User-Interfaces sowie Sicherheitsbedenken zurückgeführt werden [22]. In Hinblick auf Sicherheitsbedenken ist die digitale Archivierung von Transaktionsdaten, wie beispielsweise Buchungsbestätigungen, ein Hindernis, da in digitale Bestätigungen oftmals das Vertrauen fehlt [13].

Aufbauend auf einer Designstudie von Pavlovic [17] wurde ein einfach verwendbarer und wartungsfreier Belegdrucker konstruiert, mit welchem einerseits diese Hemmnisse abgebaut aber auch neue Einsatzgebiete wie Gutschein- oder Rezeptdruck erschlossen werden sollen. Das Design dieses Belegdruckers unterscheidet sich grundlegend von herkömmlichen PC-Druckern, da der Belegdrucker, bis auf das zu bedruckende Papier, ohne Verbrauchsmaterialien wie Tintenpatronen auskommt und vollständig wartungsfrei ist. Zusätzlich wird das Papier von oben nach unten bedruckt, sodass Druckergebnisse schnellstmöglich sichtbar werden. Es werden nur Einzelblätter bedruckt, wobei der Datentransfer und die Stromversorgung über den Universal Serial Bus erfolgen.

In einem ersten Schritt wurde, aufbauend auf der Designstudie von Pavlovic [17], ein technisches Grundkonzept erstellt, für welches passende Komponenten wie Thermodruckwerk, Controllerplatine sowie Mikrocontrollerboard recherchiert wurden. Bei der darauf folgenden Integration wurden die Datenverbindungen zwischen den einzelnen Komponenten hergestellt und diese mittels Microcontroller-Firmware angesprochen, wobei anschließend der mechanische Aufbau sowie die Konstruktion eines geeigneten Gehäuses erfolgte. Den Abschluss der Arbeiten bildete die Implementierung eines Windows-Druckertreibers, der die Aufbereitung der Druckdaten und die Kommunikation über den Universal Serial Bus sowie die Anzeige von Statusdaten übernimmt.

Abstract

Since the introduction of E-Banking services on the market ever growing customer acceptance can be observed. This can be attributed to various benefits such as lower transaction fees, location independence and high availability. However, for elderly people a lower-than-average acceptance of E-Banking services can be observed [18]. This can be attributed to technical entry barriers, sceptical attitude toward new technologies, too complicated user interfaces as well as security concerns. Regarding security concerns, the low confidence in digital archiving of confirmations is a crucial factor lowering the acceptance of E-Banking services [13].

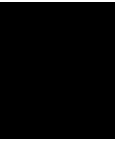
The aim of this thesis is the technical design and construction of a maintenance-free receipt printer, based on a design study created by Pavlovic [17]. Not only shall the receipt printer help to reduce barriers for E-Banking services but it can also be used to print vouchers and receipts. The design of the printer differs fundamentally from available consumer printers, because except for paper, it does not consume additional supplies and it is completely maintenance-free. Furthermore the paper is printed from top to bottom so that the printed content is visible to the user as soon as possible. The printer only consumes single sheets of paper and is connected to a PC via the Universal Serial Bus.

In the first phase of development the technical components such as the thermal printing mechanism, the print controller as well as a microcontroller board have been selected. After appropriate components had been selected, they were connected so that they can be controlled by firmware running on an Arduino Leonardo. On the PC-side a Windows printer driver was implemented, allowing standard Windows software to print on the receipt printer. After the electronic and software part of this thesis were completed, a proper housing consisting of Polystyrol was hand-crafted, allowing the useage of the printer in actual studies.

Inhaltsverzeichnis

Kurzfassung	vii
Abstract	ix
Inhaltsverzeichnis	1
1 Einleitung	3
1.1 Problemstellung	3
1.2 Forschungshintergründe	4
1.3 Methodisches Vorgehen	4
1.4 State of the Art	5
1.5 Aufbau der Diplomarbeit	6
2 E-Banking für Senioren/-innen	9
2.1 Demographische Entwicklung	9
2.2 Veränderungen der individuellen Fähigkeiten	12
2.3 Informations- und Kommunikationstechnologie für ältere Menschen	15
2.4 Banking Services	16
2.5 Fazit	19
3 Designstudie E-Banking Belegdrucker	21
3.1 Problemstellung	21
3.2 Anforderungen	24
3.3 Design und Gestaltung des Prototyps von Pavlovic	25
3.4 Zusammenfassung der Studienergebnisse von Pavlovic	26
4 Software-Architektur	29
4.1 Windows Druckertreiber	30
4.2 Firmware	43

5	Hardware	49
5.1	Thermodruckwerk	50
5.2	Steuerungsprozessor Seiko Instruments PTV10P01-E	54
5.3	Interface-Board Seiko Instruments IFV101-01B-E	58
5.4	Arduino Leonardo	58
5.5	Stromversorgung	62
5.6	Ausführung des Papiereinzugs	68
5.7	Schaltungsaufbau	73
5.8	Mechanischer Aufbau	80
5.9	Gehäuse	82
6	Fazit und Ausblick	85
6.1	Fazit	85
6.2	Ausblick	85
7	Appendix - Installationsanleitung Belegdruckertreiber	87
7.1	Download und Installation von Redmon	87
7.2	Installation des Belegdrucker UniDRV-Minitreibers	88
7.3	Anlegen des Belegdruckers	92
7.4	Konfiguration des Port-Redirectors	94
	Abbildungsverzeichnis	99
	Tabellenverzeichnis	101
	Literaturverzeichnis	107



Einleitung

1.1 Problemstellung

E-Banking verzeichnet seit der Verfügbarkeit entsprechender Technologien und deren breiter Markteinführung stetig steigende Akzeptanz und Nutzung [19]. Gründe dafür sind neben einer gestiegenen Technologieakzeptanz in der Gesellschaft zahlreiche Vorteile wie beispielsweise Ortsunabhängigkeit, ständige Verfügbarkeit sowie oftmals niedrigere Transaktionskosten. Der stetige Fortschritt der *Informations- und Kommunikationstechnologien* (ITK) hat sich positiv auf die Benutzerfreundlichkeit und die Verfügbarkeit entsprechender Lösungen ausgewirkt, was zusätzlich zur Verbreitung und gesteigerter Inanspruchnahme entsprechender Angebote beigetragen hat. War beispielsweise früher die Verwendung eines stationären *Personal Computers* (PC) mit Internetanbindung über Modem/Telefonleitung eine grundlegende Voraussetzung, um E-Banking-Dienste nutzen zu können, so haben sich durch die Entwicklungen der letzten Jahre, speziell im Umfeld der mobilen Breitband-Internetanbindung, umfassende Komfortsteigerungen bei E-Banking ergeben. So ist es heute durchaus üblich, seine Bankgeschäfte mit der Smartphone-Anwendung der Hausbank durchzuführen - oftmals in Kombination mit weiteren Aktivitäten welche Zahlungsverkehr implizieren, wie dem Ausführen einer Online-Bestellung. Zusätzlich sanken die technologischen Eintrittsbarrieren, da Smartphones mit Internetzugang dieser Tage eine sehr hohe Marktdurchdringung aufweisen und gerade unter jüngeren Menschen als Selbstverständlichkeit gelten [18].

Im Hinblick auf in ihrer Mobilität eingeschränkte Menschen, ist besonders die Ortsunabhängigkeit eine vorteilhafte Eigenschaft, da diese es ermöglicht, Bankgeschäfte ohne

Aufsuchen einer Filiale bequem von zu Hause aus erledigen zu können. Allerdings lässt sich gerade bei älteren Menschen, welche den Umgang mit Computern sowie die Nutzung von Internetdiensten nicht gewohnt sind, eine deutlich geringere Inanspruchnahme von E-Banking-Diensten als bei jüngeren Altersschichten feststellen [16]. Dies kann unter anderem auf technische Einstiegsbarrieren, kritische Haltung gegenüber neuen Technologien, Sicherheitsbedenken sowie zu komplizierten User-Interfaces aktueller E-Banking-Lösungen zurückgeführt werden [18].

Im Hinblick auf Sicherheitsbedenken, ist die digitale Archivierung von Transaktionsdaten, wie beispielsweise Buchungsbestätigungen, ein Hindernis. Bei traditionellen Bankgeschäften am Schalter wird üblicherweise eine Bestätigung in Papierform ausgehändigt, welche in physischer Form aufbewahrt werden kann. In digitale Bestätigungen, wie diese bei E-Banking üblich sind, fehlt oftmals das Vertrauen [22]. Dies ist auch einer der Gründe, weshalb Schecks als Zahlungsmittel in manchen Ländern bei älteren Menschen noch intensiv genutzt werden [12].

1.2 Forschungshintergründe

Mittels eines im Rahmen dieser Diplomarbeit entwickelten, einfach verwendbaren und wartungsfreien Belegdruckers sollen diese Hemmnisse abgebaut werden, in dem den Benutzern/-innen die Möglichkeit gegeben wird, Bestätigungen wie gewohnt in Papierform auszudrucken. Das Design dieses Belegdruckers unterscheidet sich allerdings grundlegend von herkömmlichen Druckern, da der Belegdrucker, bis auf das zu bedruckende Papier, ohne Verbrauchsmaterialien wie Tintenpatronen oder Tonerpatronen auskommen muss. Weiters muss der Drucker komplett wartungsfrei zu betreiben sein, ohne der Notwendigkeit beispielsweise periodisch den Druckkopf reinigen zu müssen. Um den Druckablauf für den/die Anwender/-in klar ersichtlich ablaufen zu lassen, wird das Papier von oben nach unten bedruckt, sodass Druckergebnisse schnellstmöglich sichtbar sind und die Druckrichtung der Leserichtung entspricht. Das zu bedruckende Format soll 1/3 A4 betragen (was in etwa dem Format von Zahlscheinen entspricht), wobei immer nur ein Einzelblatt bedruckt wird.

1.3 Methodisches Vorgehen

Dazu wird in einem ersten Schritt, aufbauend auf der Designstudie von Pavlovic [17], ein technisches Grundkonzept erstellt. Anhand dieses Konzepts wird anschließend eine Recherche nach geeigneten Komponenten wie Thermodruckerwerk, Controllerplatine sowie

Mikrocontrollerboard durchgeführt. Da Thermodrucker, wie sie bei diesem Drucker Verwendung finden, üblicherweise eine sehr hohe Stromaufnahme aufweisen (bis zu 60W), die nur teilweise über den *Universal Serial Bus* (USB) (2,5W) abgedeckt werden kann, muss zusätzlich eine Pufferschaltung mittels Akkumulatoren und Ladungspumpe realisiert werden.

Bei der darauf folgenden Integration werden die Datenverbindungen zwischen den einzelnen Komponenten hergestellt. Anschließend wird mittels Mikrocontroller-Board eine rudimentäre PC-Schnittstelle über USB realisiert, die das Testen des Aufbaus erlaubt und im Laufe der weiteren Entwicklung inkrementell vervollständigt wird. Unabhängig davon wird ein Einzelblatt-Papiereinzug konstruiert, welcher ebenfalls von dem Mikrocontroller-Board angesteuert wird.

Den Abschluss der Arbeiten bildet die Implementierung eines Windows-Druckertreibers, der die Aufbereitung der Druckdaten und die Kommunikation über USB und die Anzeige von Status-Daten übernimmt.

1.4 State of the Art

Ein ähnliches Konzept wurde von Orrandre et al. in [11] verwirklicht, wobei der verwendete Drucker dabei allerdings nur ein Teil eines Gesamtkonzepts namens „Chat Machine“ darstellt. Die sogenannte „Chat Machine“ ermöglicht älteren Menschen, handschriftlich Emails zu verfassen, zu versenden und zu empfangen. Ausführlich wird dabei die Bedeutung der papiergebundenen Kommunikation für ältere Menschen betont. Neben einem monochromen Thermodrucker für handschriftliche Ausdrücke verfügt die „Chat Machine“ ebenfalls über einen Fotodrucker, welcher ebenfalls auf Thermodruckerbasis funktioniert. Allerdings lag bei der Entwicklung der „Chat Machine“ der Fokus auf der Integration vorhandener Produkte. Dies bedingt, dass beispielsweise die Druckbreite des Monochromdruckwerks sehr eingeschränkt ist und dieses auf Endlospapier druckt. Auch das spezielle Fotopapier befindet sich in einer Cartridge, welche im Gerät verbaut ist. Die „Chat Machine“ ist somit aus Benutzersicht nicht wartungsfrei und benötigt regelmäßig die Intervention von mit der Wartung vertrauten Personen. Dieser Aspekt der „Chat Machine“ könnte durch den, in dieser Diplomarbeit vorgestellten, wartungsfreien Einzelblatt-Thermodrucker dahingehend verbessert werden, als dass nur noch das geeignete Thermopapier im Einzelblattformat bei der Verwendung vorrätig sein müsste. Als Nebeneffekt besitzt der, in dieser Arbeit beschriebene, Drucker ein breiteres Druckwerk, wodurch Ausdrücke, welche Handschrift enthalten, größer und somit besser

lesbar ausfallen würden.

Der Hersteller Brother^{1,2} bietet eine Serie von Druckern für Geschäftsreisende auf Basis des Thermodruckverfahrens an, wobei diese Drucker A4 Einzelblätter verarbeiten können. Auch wenn bis auf das Thermopapier keine Verbrauchsmaterialien benötigt werden und diese Serie von Druckern daher als weitgehend wartungsfrei bezeichnet werden kann, wird die Anforderung an einen einfachen Bedienungsablauf nicht erfüllt. Dies rührt von der Tatsache her, dass diese Geräte über keinen Papiereinzug im herkömmlichen Sinn verfügen und daher jedes Blatt manuell eingespannt und anschließend fixiert werden muss, was neben dem an sich recht komplexen Ablauf auch hohe Anforderungen an die Feinmotorik der Benutzer/-innen stellt.

In der Diplomarbeit von Pavlovic [17] wurde ein Prototyp des Gehäusedesigns vorgestellt, auf welchem im Rahmen dieser Diplomarbeit aufgebaut wurde. Weiters wurde darin die Akzeptanz anhand eines Mockups mit einer Gruppe von Senioren/-innen getestet, wobei deren Feedback in das finale Design eingeflossen ist. In dieser Arbeit wird auf die dabei gewonnenen Erkenntnisse aufgebaut. Dabei dient die, von Pavlovic entworfene, Design-Studie als Grundlage für den Prototyp, welcher im Rahmen dieser Diplomarbeit entwickelt wurde.

1.5 Aufbau der Diplomarbeit

Zu Beginn der Diplomarbeit wird in Kapitel 2 kurz auf die demographische Entwicklung eingegangen, um die aktuell hohe und sogar noch steigende Relevanz von assistiven Technologien in der Zukunft zu veranschaulichen. Darauf folgend wird in Abschnitt 2.2 auf die Veränderungen der individuellen Fähigkeiten mit zunehmendem Alter thematisiert, um ein besseres Verständnis für die Anforderungen an unterstützende Technologien zu schaffen. Abschließend wird in Abschnitt 2.3 auf die aktuelle Nutzung von Informations- und Telekommunikationstechnologie durch Senioren/-innen eingegangen, wobei zusammenfassend die Gründe für die (Nicht-) Akzeptanz einzelner Technologien beleuchtet werden. Akzeptanzbeispiele flossen als Ausgangsbasis in die Entwicklung des Belegdruckers ein, indem versucht wurde, Fallstricke bei der Bedienung im Voraus zu vermeiden sowie einen klaren Zusatznutzen durch das Gerät zu schaffen.

Kapitel 3 behandelt schon dediziert die Konzeptionierung des Belegdruckers, wobei zu Beginn, in Abschnitt 3.1, auf die Problemstellung eingegangen wird. Zusätzlich

¹<http://www.brother.at/drucker/mobile-drucker/pj-623> (letzter Aufruf: 2016-06-09)

²<http://www.brother.at/drucker/mobile-drucker/pj-762> (letzter Aufruf: 2016-06-09)

werden weitere Verwendungsmöglichkeiten des Belegdruckers erläutert, welche neben dem Haupteinsatzgebiet, dem Drucken von Kontoauszügen und Belegen, einen Mehrwert für den/die Benutzer/-in bieten können und so zu einer gesteigerten Akzeptanz beitragen. Anschließend werden die besonderen Anforderungen an das Design beleuchtet und in Abschnitt 3.3 wird schließlich die Designstudie von Pavlovic vorgestellt, welche im Rahmen einer vorherigen Diplomarbeit entstanden ist.

Kapitel 4 und 5 behandeln im Detail die praktische Umsetzung eines Belegdrucker-Prototypen, wobei Kapitel 4 auf den Aufbau und die Architektur der einzelnen Softwarebestandteile eingeht. Im Detail wird dem/der Leser/-in die Entwicklung des Windows Druckertreibers und die Aufgaben und das Design der Embedded-Software, welche auf einer Arduino-Mikrokontroller-Platine läuft, nähergebracht, wobei ebenfalls auf die technischen Hintergründe eingegangen wird.

Kapitel 5 hingegen beleuchtet die Umsetzung des Belegdrucker-Prototyps aus Hardware-Sicht. So wird im Detail auf die verwendeten Drucker-Komponenten wie Druckwerk, Steuerplatine, Arduino und Stromversorgung eingegangen, um anschließend ein tiefgreifendes Verständnis für die technische Umsetzung der Hardwarekonstruktion vermitteln zu können. Mit ausführlichen Schaltplänen inklusive Erklärung sowie Informationen zum Design des Gehäuses und dessen mechanischer Konstruktion schließt das Kapitel und ermöglicht so dem/der geneigten Leser/-in einen Nachbau des im Rahmen der Diplomarbeit entstandenen Belegdruckers.

Den Abschluss bildet das Kapitel 6, welches in zusammenfassender Form die Errungenschaften und Erkenntnisse in Kurzform aufbereitet. Weiters werden Möglichkeiten zur Weiterentwicklung, im Speziellen aber auch zur Kostensenkung, aufgezeigt sowie ein Ausblick auf zusätzliche mögliche Einsatzgebiete des Druckers präsentiert.

E-Banking für Senioren/-innen

Ziel dieses Kapitel ist es, dem/der Leser/-in die Rahmenbedingungen für diese Arbeit sowie deren Hintergründe zu vermitteln. Es beginnt mit einer detaillierten Darstellung der demographischen Entwicklung in Abschnitt 2.1 und geht dabei auf den Hintergrund dieser Entwicklung, wie beispielsweise den „Baby-Boom“ der 1950er-1960er Jahre, ein. Hauptziel dieses Abschnittes ist es, neben dem Bereitstellen von Hintergrundinformation, dem/der Leser/-in ein Verständnis für die Wichtigkeit von assistiven Technologien anhand der zu erwartenden demographischen Entwicklung aufzuzeigen.

Abschnitt 2.2 behandelt die sich verändernden Fähigkeiten von Senioren/-innen, auf welche beim Design von Technologie für diese Zielgruppe Rücksicht genommen werden muss, wobei auf die Veränderungen der wichtigsten Fähigkeiten wie sensorische, kognitive und motorische Fähigkeiten eingegangen wird.

Die Akzeptanz und Nutzung von Telekommunikationstechnologie wird in Abschnitt 2.3 kurz beleuchtet. Anschließend wird in Abschnitt 2.4 auf die verschiedenen, aktuell verfügbaren, Möglichkeiten eingegangen, um Bankgeschäfte elektronisch zu erledigen. Zusätzlich werden die Herausforderungen beschrieben, die diese Technologien für Senioren/-innen bergen können. Weiters wird auf die Akzeptanz dieser Technologien eingegangen.

2.1 Demographische Entwicklung

Ausgehend von Prognosen, wie beispielsweise der Veröffentlichung „Demografische Entwicklung im 20. Jahrhundert“ im Familienbericht des Österreichischen Bundesministeriums für Wirtschaft, Familie und Jugend, kann davon ausgegangen werden, dass die

Gruppe der über 65-jährigen, wie in Abbildung 2.1.1 illustriert, bis 2050 kontinuierlich steigen wird [4].

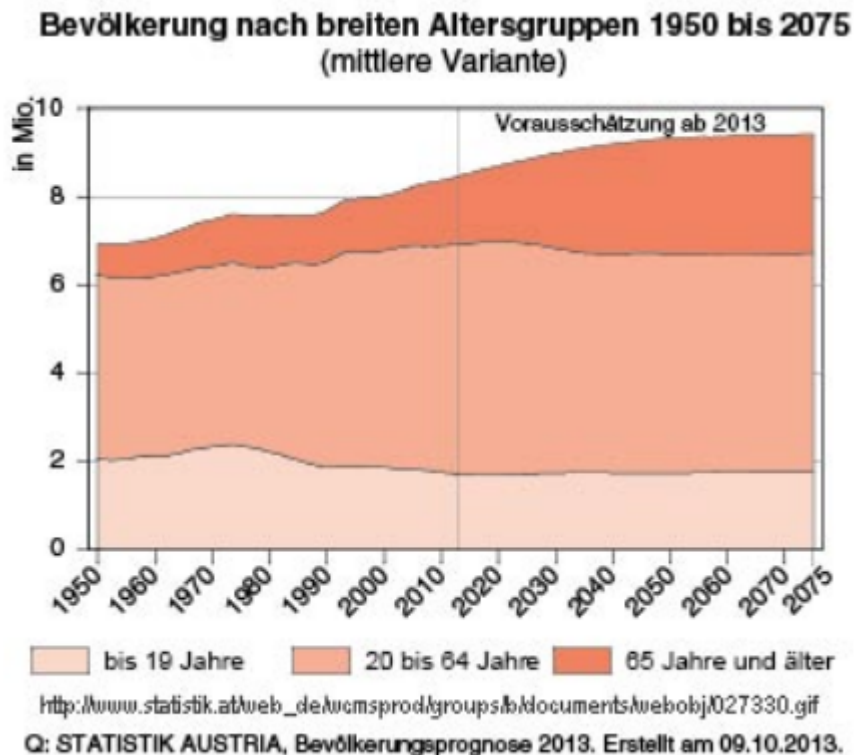


Abbildung 2.1.1: Bevölkerungsentwicklung bis 2075 [4]

Einer der Gründe für die prognostizierte Entwicklung hat ihren Ursprung im „Baby-Boom“ der 1950- und 1960er-Jahre, welcher sowohl in den USA und in etwas abgeschwächter Form auch in Europa einsetzte und durch sinkendes Alter bei Heirat und Erstgeburt kurzzeitig noch ausgeprägter auftrat. In [4] wird diese Zeit auch als „Zweiter Demographischer Übergang“ bezeichnet. Diese Phase währte allerdings nur einige Jahre, denn bereits Mitte der 1960er-Jahre sanken die Geburtenzahlen in Westeuropa wieder. Diese Entwicklung setzte sich bis Mitte der 1990er-Jahre fort, als pro Frau nur mehr eine Kinderzahl von 1,5 erreicht wurde, wobei die Kinderzahl seit dem wieder leicht im Steigen begriffen ist. Eine Kinderzahl von 2 Kindern pro Frau gilt dabei als notwendig, damit eine Population ein stabiles Gleichgewicht erreicht, also Sterbe- und Geburtenzahlen in etwa gleich hoch sind. Die Generation, welche aus dem „Baby-Boom“ hervorgegangen ist, stellt daher in der Altersstruktur eine gewichtige Gruppe dar, da folgende Generationen wieder weniger ausgeprägt waren.

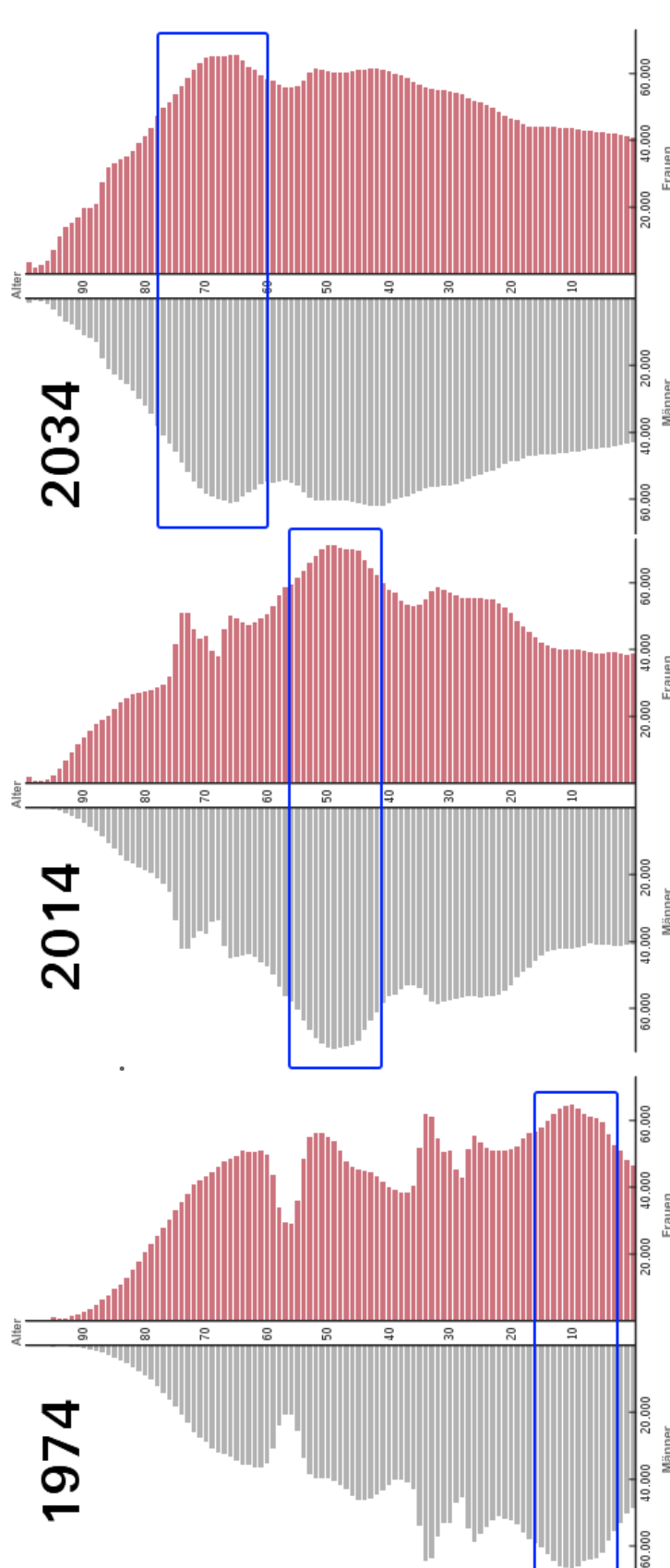


Abbildung 2.1.2: Demographische Auswirkungen des „Babybooms“ [4]
 Auswirkungen des „Baby-Booms“ (blaues Rechteck) der 1950er-1960er Jahre auf die
 Entwicklung der Bevölkerungsstruktur

Laut [4] hat diese Bevölkerungsentwicklung direkte Konsequenzen auf den Altersaufbau der Bevölkerung. Der Anteil der Gruppe der unter 20-jährigen lag 2008 bei 21,7% und dürfte 2030 auf 19,7% zurückgehen, wobei bis 2050 von einem weiteren Rückgang auf 18,9% ausgegangen wird. Im Gegensatz dazu wird sich das Wachstum der älteren Bevölkerungsschichten in den kommenden Jahrzehnten weiter fortsetzen. Während aktuell die Anzahl der Menschen über 65 Jahren 17,1% der Gesamtbevölkerung der EU beträgt, wird diese Altersgruppe 2030 bereits 23,6% der Gesamtbevölkerung betragen und bis 2050 auf 29% steigen. Anschließend wird sich das Wachstum dieser Altersgruppe laut Prognose kaum noch fortsetzen, da die bereits angesprochene Generation des „Baby-Booms“ der 1950er-1960er Jahre dann verstorben sein wird. Abbildung 2.1.2 veranschaulicht diese Entwicklung anhand der Alterspyramide.

Fazit: Durch die stetige Zunahme von Menschen der Altersgruppe der über 65-jährigen steigt auch die Bedeutung an assistiven Technologien, welche es diesen Menschen ermöglicht, möglichst lange ein selbstständiges Leben mit hoher Lebensqualität führen zu können.

2.2 Veränderungen der individuellen Fähigkeiten

Mit zunehmendem Alter verändern sich die individuellen Fähigkeiten eines Menschen, welche grob in kognitive, körperliche sowie sensorische Fähigkeiten unterteilt werden. Da dieser Prozess von Mensch zu Mensch unterschiedlich verläuft, ergibt sich ein sehr heterogenes Bild bezüglich der Fähigkeiten älterer Menschen. Während man bei einem langsamen, stetigen Verlust von Fähigkeiten von biologischer Alterung spricht, verläuft der krankheitsbedingte Verlust von Fähigkeiten typischerweise schneller. Der Fähigkeitsverlust mit zunehmendem Alter steht im Gegensatz zum Fähigkeitsgewinn in den ersten Lebensjahren.

Wie sich die Abnahme von Fähigkeiten auf den Umgang mit Informations- und Kommunikationstechnologie auswirkt, wird in den folgenden Abschnitten näher erläutert.

2.2.1 Motorische Fähigkeiten

Die wichtigsten Fähigkeiten, welche mit zunehmendem Alter abnehmen, sind Stärke der Muskeln, der Gleichgewichtssinn und die Beweglichkeit. Besonders die Beweglichkeit der Gliedmaßen ist für das Design von User-Interfaces ausschlaggebend. Wie in [3] näher erläutert, ist die Beweglichkeit des Zeigefingers einer der wichtigsten Parameter diesbezüglich. Dieser dient auch als Maßstab zur Bewertung der motorischen Fähigkeiten, da wie in Abbildung 2.2.1 dargestellt, dessen Beweglichkeit mit zunehmendem Alter abnimmt. Aber auch Krankheiten, wie beispielsweise die Parkinson-Krankheit, können die Feinmotorik der betroffenen Personen negativ beeinflussen.

Age	Gender	Mean	P5	P95
55-59	M	51	38	65
	W	53	35	73
60-64	M	50	34	67
	W	55	39	74
65-69	M	52	36	67
	W	53	36	70
70-74	M	54	34	73
	W	53	40	68
75-79	M	49	35	64
	W	53	41	68
80+	M	54	41	64
	W	53	37	68

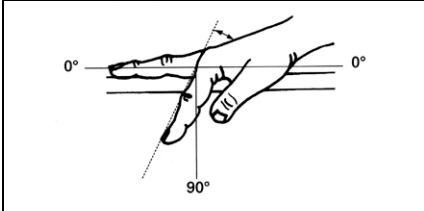


Abbildung 2.2.1: Movement Rank je nach Altersgruppe und Geschlecht [3]
 Der Movement Rank (erreichbarer Winkel zwischen Zeigefinger und der horizontalen Achse) dient als Maß, um die Beweglichkeit der Finger mit zunehmendem Alter zu beurteilen

2.2.2 Sensorische Fähigkeiten

Zusätzlich zu den körperlichen Fähigkeiten nehmen auch die sensorischen Fähigkeiten mit zunehmendem Alter ab. Die für das Design von ITK-Technologie wichtigsten sensorischen Fähigkeiten sind die visuellen, die auditiven sowie die taktilen Sinne.

Der Rückgang der visuellen Sinne lässt sich unter anderem an folgenden Punkten beobachten:

- Rückgang der Tiefenwahrnehmung
- Rückgang der Hell-Dunkel-Anpassung
- Einschränkung des Bereichs, in welchem scharf gesehen werden kann
- Weniger Kontrast-Sensitivität
- Geringere Licht-Sensitivität, daher benötigen ältere Menschen bis zu 3x mehr Licht als junge Erwachsene
- Rückgang der visuellen Sehschärfe um nahezu den Faktor 4 verglichen mit jungen Erwachsenen

Bei den auditiven Sinnen können folgende Effekte auftreten:

- Probleme bei der Sprachverständlichkeit
- Rückgang der Fähigkeit zur Ortung von Schallquellen
- Höhere Lärmempfindlichkeit

Mit zunehmendem Alter tritt zusätzlich der Effekt auf, dass hohe Frequenzen nicht mehr wahrgenommen werden können. Dieser Effekt setzt bereits in jungen Jahren ein [3], in denen der betroffene Frequenzbereich allerdings so hoch ist, dass der Verlust noch keine merkliche Einschränkung darstellt. Mit fortschreitender Alterung des Gehörs sind allerdings immer tiefere Frequenzen betroffen, so dass beispielsweise Probleme mit dem Verstehen von Sprache die Folge sein können.

Zusätzlich zu den auditiven und visuellen Fähigkeiten, nehmen auch die taktilen Fähigkeiten ab. So fällt es älteren Menschen oft schwer, Rauheit von unter 0.05mm festzustellen, wobei 10% der älteren Menschen sogar Schwierigkeiten haben, eine Rauheit von 0.1mm zu erfassen [3].

2.2.3 Kognitive Fähigkeiten

Laut [3] lässt sich die Fähigkeit zur Aufmerksamkeit auf folgende 4 Kategorien klassifizieren:

- Selektive Aufmerksamkeit: Fähigkeit des Menschen, relevante Informationen aus einem Informationsstrom zu filtern. Ältere Menschen lassen sich leichter ablenken und besitzen daher geringere selektive Aufmerksamkeit.
- Die Fähigkeit über einen längeren Zeitraum aufmerksam zu sein. Diese Fähigkeit bleibt auch bei älteren Menschen nahezu vollständig erhalten.
- Die Fähigkeit „Multitasking“ zu betreiben, also die Aufmerksamkeit zwischen zwei oder mehr Aufgaben gleichzeitig aufzuteilen, nimmt mit zunehmendem Alter kontinuierlich ab.
- Auch die Fähigkeit, mit der Aufmerksamkeit zwischen mehreren Ereignissen zu wechseln, ist bei älteren Menschen weniger ausgeprägt. In [3] wird vermutet, dass dies mit der geringen Gedächtnisleistung älterer Menschen in Verbindung steht.

2.2.4 Fazit

Die fortschreitende Abnahme der Fähigkeiten stellt Herausforderungen an das Design von ITK-Equipment für Senioren/-innen, welches diese besonderen Anforderungen berücksichtigen muss. Allerdings wird in [3] auch darauf hingewiesen, dass durch das kontinuierliche Training der beschriebenen Fähigkeiten durch Verwendung von ITK oder mittels Lernspielen der Abbau der Fähigkeiten verlangsamt werden kann. Beim Design von auf Senioren/-innen ausgerichteter ITK muss allerdings bedacht werden, dass Senioren/-innen, auch wenn diese oft in Altersklassen eingeteilt werden, keine homogenen Gruppen darstellen, sondern natürlich individuell ausgeprägte Fähigkeiten besitzen sowie einen unterschiedlichen Erfahrungsschatz aufweisen und differierende Erwartungen an eine neu entwickelte Technologie oder ein neues Produkt haben [23].

2.3 Informations- und Kommunikationstechnologie für ältere Menschen

Laut [14], ist eine der größten Hürden für die Akzeptanz neuer Technologien ein Mangel an empfundenem Mehrnutzen und nicht, wie oft vermutet, Skepsis oder Voreingenommenheit gegenüber Technologie im Allgemeinen.

Technologie, welcher der Kommunikation dient, erfährt die höchste Akzeptanz, da Senioren/-innen diese nutzen, um ihr Soziales Kapital zu erhöhen. Zu einem gewissen Grad rührt die hohe Akzeptanz eventuell auch von der Tatsache her, dass Senioren/-innen der heutigen Zeit schon mit Echtzeitkommunikation über Telefon und Faxgeräte aufgewachsen sind, diese Form der Kommunikation also vertraut ist. Eine Vermutung des Autors ist, dass moderne Kommunikationsformen wie E-Mail oder Videotelefonie als technisch verbesserte Implementierung dieser bereits bekannten Kommunikationsmittel angesehen werden und daher der Zusatznutzen schnell erkannt werden kann, was wiederum zur erwähnten hohen Akzeptanz führt.

Laut [14] sind die größten Hürden für Senioren/-innen bei der Verwendung von neuen Technologien auf die mangelnde Rücksichtnahme auf ihre veränderten Fähigkeiten zurückzuführen, auf welche in Abschnitt 2.2 genauer eingegangen wurde. Dies betrifft zum Beispiel:

- Zu kleine Knöpfe, kein taktiles Feedback („Klick bei erfolgter Betätigung“)
- Unhandliche, kleine Geräte
- Zu kleiner Text, welcher trotz Brillengläsern für Senioren/-innen kaum lesbar ist
- Zu komplizierte Menüstrukturen

2.4 Banking Services

Seit einigen Jahren werden Bankgeschäfte, die früher üblicherweise am Bankschalter abgewickelt wurden, zunehmend über elektronische Geräte, die von den Kunden selbst bedient werden, abgewickelt. Während Bankomat-Karten und Geräte schon länger zum Alltag der meisten Bank-Kunden/-innen gehören, so nahm in den vergangenen Jahren die Möglichkeiten zu, mit eigenen Geräten Bankgeschäfte über das Internet abzuwickeln. So bieten neuerdings nahezu alle Banken Anwendungen für Smartphones an, während E-Banking mittels Browser am PC schon etwas länger verbreitet ist.

In den folgenden Abschnitten soll auf die Eigenschaften und Eignung dieser elektronischen Geräte für Senioren/-innen eingegangen werden. Weiters wird versucht, die Probleme aufzuzeigen, die für mangelnde Akzeptanz in dieser Zielgruppe verantwortlich sein könnten.

2.4.1 Bankomat

Bankomaten sind Geräte, die das Abwickeln von typischen Bankgeschäften rund um die Uhr ermöglichen. Oftmals sind sie direkt bei Banken, teilweise auch in geschützten Vorräumen, installiert. Es gibt aber auch Bankomaten, welche beispielsweise in Supermärkten aufgestellt sind.

Die Hauptprobleme bei der Verwendung von Bankomaten sind für Senioren/-innen unter anderem:

- Verlust der Feinmotorik, welche für die Bedienung von Funktionsknöpfen beziehungsweise eines eventuell vorhandenen Touch-Screens wichtig ist.
- Eingeschränktes Sehvermögen, daher kann es zu Problemen beim Lesen der Anweisungen am Bildschirm kommen (Kontrast, Blendung).
- Eingeschränkte kognitive Leistungsfähigkeit, wodurch komplexere Menüstrukturen und Optionen Verwirrung stiften können (zeitkritische Bedienungsabläufe).
- Je nach Bankomat-Typ unterschiedlicher Funktionsumfang und wechselnde User-Interfaces.

Aus persönlicher Erfahrung des Autors kann zusätzlich der Zeitdruck bei der Verwendung von Automaten eine beträchtliche Einstiegshürde darstellen. Üblicherweise sind Bankomat-Geräte gut frequentiert und die Dauer einer Transaktion beschränkt sich auf wenige Minuten. Zum Erlernen der Menüstruktur und zum Verstehen des Ablaufs ist allerdings, besonders unter Rücksichtnahme auf eingeschränkte Fähigkeiten von Senioren/-innen, wesentlich mehr Zeit erforderlich. Daher könnte die zeitliche Komponente ebenfalls eine Barriere für ältere Menschen darstellen, die sie daran hindert, sich mit dieser Technologie zu beschäftigen.

Auf die negativen Auswirkungen von Stress beim Umgang mit Technologie wird detailliert in [15] eingegangen, wobei an dieser Stelle nicht gesondert erwähnt werden muss, dass sich Stress sowohl auf den Selbstwert des/der Users/-in als auch auf die Akzeptanz der technischen Lösung negativ auswirkt. Auch in [6] wird auf die Notwendigkeit der Vermittlung eines positiven Selbstbildes für die Akzeptanz einer neuen Technologie hingewiesen.

2.4.2 Smartphones

Obwohl über 80% der Senioren/-innen ein Mobiltelefon benutzen, verwendet nur ein geringer Prozentsatz das Mobiltelefon, um darauf Internet-Dienste in Anspruch zu nehmen [3].

Die Größe der Mobiltelefone ist dabei ein Hindernis, oftmals sind Bedienelemente bei Smartphones so klein ausgeführt, dass es sogar jungen Menschen schwer fallen kann, beispielsweise die Bildschirmtastatur zu benutzen. Andererseits sind Mobiltelefone, welche explizit für die Bedürfnisse von Senioren/-innen konzipiert wurden, oftmals stark in deren Funktionalität beschnitten und eignen sich beispielsweise oftmals nur eingeschränkt um damit im Internet zu surfen.

2.4.3 E-Banking

Mittlerweile bieten nahezu alle Banken E-Banking-Dienste über Internet an, welche ein breites Spektrum des Leistungsangebotes der Bank abdecken. E-Banking hat allerdings auch hohe Anforderungen an die Ausstattung der Benutzer/-innen, so muss eine Internet-Anbindung bestehen und der/die Benutzer/-in muss im Besitz eines Internet-fähigen Gerätes sein.

Zusätzlich zu den, im Abschnitt 2.4.1 aufgezählten, Schwierigkeiten bei der Bedienung von Bankomaten, stellt E-Banking höhere Anforderungen an die kognitiven Fähigkeiten der Benutzer/-innen. So sind die Möglichkeiten oftmals umfangreicher, was im Gegenzug zu komplizierter Menüstruktur führt. Während der Bedienungsablauf beim Bankomat auf einige wenige Grundfunktionen beschränkt ist, versuchen Banken oftmals in deren E-Banking Oberflächen zusätzliche Dienste wie beispielsweise Info-Portale oder Wertpapier-Verwaltungen zu integrieren, die das Interface für unbedarfte Anwender/-innen überladen erscheinen lassen. Im Vergleich zum Bankomat haben User/-innen keinen Zeitdruck beim Erlernen der Abläufe, auch das User-Interface bleibt konsistent.

Als Vorteil ist zu nennen, dass durch die Benutzung des eigenen Geräts für E-Banking zielgruppenorientierte Anpassungen am User-Interface möglich sind, wie beispielsweise ein seniorenfreundliches Touch-Interface, das in dieser Form bei Bankomaten nicht möglich ist.

Sicherheitsaspekte

Sicherheit ist ein wesentlicher Einflussfaktor für die Akzeptanz von E-Banking Lösungen, gehört doch E-Banking zu einer der sensibelsten Aktivitäten, die Benutzer/-innen über das

Internet durchführen können [13]. Neben technischen Aspekten sollte bei der Verwendung von E-Banking nicht auf die Notwendigkeit eines ausgeprägten Sicherheitsbewusstseins der Anwender/-innen vergessen werden. So ist beispielsweise der Umgang mit sensiblen Zugangsdaten wie einer *Personal Identification Number* (PIN) oder einer *Transaction Authentication Number* (TAN) eine Grundvoraussetzung für die sichere Nutzung von E-Banking-Systemen. Für eine ausführliche Abhandlung bezüglich der Sicherheitsaspekte von E-Banking sei auf die Studie von Mannan und Oorschot [13] verwiesen, welche sowohl die technischen, als auch die humanen Faktoren (wie beispielsweise Angriffe über „Social Engineering“ anstatt technischer Sicherheitslücken [10]) beleuchtet.

2.5 Fazit

Zusammenfassend kann festgehalten werden, dass die Möglichkeiten, welche die moderne ITK bietet, die Lebensqualität und den Aktionsradius von älteren Menschen signifikant steigern kann. Wichtig ist es allerdings, dass der sich daraus ergebenden Zusatznutzen für die Zielgruppe klar ersichtlich ist und die Technologie an die Fähigkeiten der Zielgruppe angepasst ist. Oft kann die Akzeptanz zusätzlich erhöht werden, wenn man den Senioren/-innen bei gewohnten Abläufen entgegenkommt und diese mit Hilfe der ITK direkt nachbildet. Ein Beispiel dafür sind Überweisungsbestätigungen oder Gutscheine in Papierform, die das Vertrauen durch Beibehaltung gewohnter physischer Ablaufelemente stärken [24]. Mit der Gestaltung eines Belegdruckers soll daher im Rahmen dieser Diplomarbeit die technische Basis für die Beibehaltung beziehungsweise Abbildung gewisser gewohnter und bewährter Abläufe auf Papierform geschaffen werden.

Designstudie E-Banking Belegdrucker

Bei der Gestaltung des Belegdruckers wurde auf die Erkenntnisse der Diplomarbeit von Goran Pavlovic [17] zurückgegriffen, welcher im Rahmen seiner Diplomarbeit prototypisch ein Gehäuse für den Belegdrucker nach den Methoden des Partizipatory Designs [5] entworfen und mittels 3D-Drucker auch gefertigt hat.

In Abschnitt 3.2 wird auf die Anforderungen an das Drucker-Design sowie die technische Ausführung eingegangen, welche größtenteils schon bei Beginn der Diplomarbeit von Pavlovic feststanden. Zusätzlich ergaben sich während der Arbeit von Pavlovic weitere Einsatzzwecke, welche in das Konzept des, im Rahmen dieser Diplomarbeit erstellten, Prototyps Eingang gefunden haben. Auf das daraus resultierende Design wird in Abschnitt 3.3 näher eingegangen und mit einigen Bildern veranschaulicht.

3.1 Problemstellung

Für viele Senioren/-innen ist die ausschließlich digitale Abwicklung von Bankgeschäften mit Sicherheitsbedenken verknüpft, wobei sich diese Bedenken nicht nur gegen die Möglichkeit eines „Hacks“ richten, sondern auch gegen die Vertrauenswürdigkeit der Bankinstitute [3]. Dieses Vertrauen in die Bankinstitute wurde durch die Geschehnisse der Finanzkrise, bei welcher selbst große Finanzinstitute in finanzielle Bedrängnis kamen, noch weiter verstärkt.

Während bei Bankgeschäften am Bankschalter in den meisten Fällen eine Art Quittung oder ein Beleg ausgestellt wird, mit welchem sich die durchgeführte Transaktion belegen lässt, so findet diese Archivierung bei über das Internet abgewickelten Bankgeschäften ausschließlich in digitaler Form statt. Während es Papierbelege erlauben, in physisch greifbarer Form beispielsweise in Ordnern archiviert zu werden, stellen Belege in digitaler Form ein gewisses „Mysterium“ dar, da oftmals aufgrund eingeschränkter Computerkenntnisse auch nicht bekannt ist, wie diese auf dem eigenen Computer archiviert werden können, beziehungsweise mittels eines normalen Druckers in Papierform überführt werden können.

Theoretisch besteht natürlich die Möglichkeit, Belege mittels handelsüblicher Consumer-Drucker, wie beispielsweise Tintenstrahldrucker oder Laserdrucker, auszudrucken. Dabei muss allerdings bedacht werden, dass klassische Drucker einen hohen technischen Anspruch an den/die Benutzer/-in stellen, da regelmäßig nicht-triviale Wartungsarbeiten, wie zum Beispiel das Wechseln der Tintenpatronen oder das Befüllen der Papierlade, notwendig sind. Durch die komplexe Ausführung essentieller Bestandteile wie des Papiereinzugs oder des Druckkopfs kann es auch zu einer Vielzahl von sporadisch auftretenden Betriebsstörungen, wie das Verstopfen des Druckkopfes bei Tintenstrahldruckern, oder, aufgrund der aufwändigen Papierführung über eine Vielzahl von Umlenkrollen, einem Papierstau bei Laserdruckern kommen. Andererseits stellt die breite Funktionalität dieser Consumer-Produkte, die mit reduzierter Zuverlässigkeit und höherem Wartungsaufwand erkaufte wird, oftmals keinen Mehrwert für den geplanten Einsatzzweck dar. Für den Ausdruck von Überweisungsbestätigungen, Rezepten oder Gutscheinen ist beispielsweise Schwarz/Weiß-Druck vollkommen ausreichend. Auch die Möglichkeit, viele Ausdrücke sequentiell durchführen zu können, ohne manuell ein Blatt Papier einlegen zu müssen, stellt keinen großen Mehrwert dar, da zur persönlichen Archivierung üblicherweise die Belege manuell ausgewählt, Stück-für-Stück gedruckt und anschließend einzeln abgelegt werden.

Neben den hardware-technischen Schwierigkeiten bei der Verwendung von bereits im Handel verfügbaren Druckern zum Archivieren von Banktransaktions-Bestätigungen, kommen zusätzlich Herausforderungen beim Initiieren eines Druckauftrages innerhalb der E-Banking Software. So resultiert ein direkter Ausdruck eines Browser-Fensters nur selten in dem gewünschten Seiten-Layout oder der erwarteten Formatierung, da *Hypertext Markup Language* (HTML) nicht konzipiert wurde, um das *What You See is What You Get* (WYSIWYG) - Paradigma zu unterstützen. Erwähnenswert ist dabei, dass mit *Cascading Style Sheets 3* (CSS3) Möglichkeiten geschaffen wurden, um mittels zusätzlicher Layout-

Direktiven konsistentere Druckergebnisse zu erhalten. Trotzdem setzen viele Banken auf ein WYSIWYG-taugliches Zwischenformat wie das *Portable Document Format* (PDF). Obwohl dieses Format aus technischer Sicht ideal geeignet ist, konsistente Ausdrücke zu erstellen, ergeben sich durch dessen Verwendung zusätzliche Usability-Probleme. Da es sich bei PDF-Dokumenten aus Sicht der Browser um „fremde“ Dokumente handelt, werden diese entweder mittels eines separaten Dokumentenbetrachters geöffnet, oder sofern der Browser selbst oder mittels eines Plugins die Darstellung von PDF-Dokumenten unterstützt, in einem separaten Tab/Reiter geöffnet. Dieser zusätzliche Schritt von Indirektion unterbricht den Arbeitsfluss und stellt höhere Anforderungen an die kognitiven Fähigkeiten sowie das Technikverständnis der Senioren/-innen.

Fazit: Aufgrund der im vorangegangenen Absatz ausführlich beschriebenen Probleme bei der Verwendung von Consumer-Druckern sowie Browser-basierten Standard-Software-Lösungen für E-Banking, wurde die Entwicklung einer kombinierten und aufeinander abgestimmten Hard- und Softwarelösung als Ziel dieser Diplomarbeit festgelegt. Durch die selbst entworfene Hardware, auf welcher der Schwerpunkt dieser Diplomarbeit liegt, soll die Notwendigkeit, periodisch Wartungsarbeiten durchführen zu müssen, vollständig eliminiert werden. Die Usability-Probleme bei der Verwendung von Browser-basierten Standard-Software-Lösungen für E-Banking werden in einer separaten Arbeit am Institut behandelt.

3.1.1 Weitere Verwendungsmöglichkeiten des Belegdruckers

In [17] wurden weitere Einsatzmöglichkeiten eines Online-Banking-Belegdruckers ausgearbeitet, welche es neben dem geplanten Haupteinsatzzweck (dem Drucken von Kontoauszügen und Belegen) erlauben, zusätzlichen Mehrwert für die Zielgruppe zu generieren.

Zu den weiteren Einsatzmöglichkeiten zählen zum Beispiel das Ausdrucken von:

- Fahrkarten
- Gutscheinen
- Eintrittskarten
- Flug- und Reisebuchungsbestätigungen
- Bestellbestätigungen und Abholscheinen
- Rezepten für Medikamente

Alle diese Einsatzmöglichkeiten sind mit dem geplanten Papierformat, welches in etwa dem Format von Zahlscheinen entspricht (1/3 A4), problemlos durchführbar.

Ein weiterer Einsatzzweck, welcher als Vorschlag während der Evaluierungsphase in [17] mit Personen aus der Zielgruppe eingebracht wurde, ist die Integration einer Scan-Funktion in den Belegdrucker. Dabei wäre es möglich, den Papiereinzug des Druckwerks zu nutzen, um einen konstanten Vorschub des zu scannenden Dokuments zu erreichen. Zusätzlich müsste beim Design des Gehäuses Platz für eine Scan-Zeile sowie eine Beleuchtung eingeplant werden. Diese Anforderungen wurden beim Design und bei der technischen Realisierung des Belegdruckers soweit wie möglich bedacht und eingeplant, die Scan-Funktion selbst wurde allerdings, da sie nicht im Fokus dieser Arbeit steht, nicht innerhalb dieser Diplomarbeit realisiert.

3.2 Anforderungen

Die folgenden Anforderungen wurden bereits im Voraus an den Belegdrucker gestellt oder haben sich im Rahmen der Evaluierungsphase des Design-Mockups in der Diplomarbeit von Pavlovic [17] ergeben:

- Es werden ausschließlich Einzelblätter verarbeitet, da das Wechseln von Papierrollen bei Endlosdruck der Zielgruppe der Senioren/-innen nicht zumutbar ist
- Der Belegdrucker muss komplett wartungsfrei arbeiten, es dürfen keine Verschleißteile wie Tonerkassetten verwendet werden, die eines regelmäßigen Austausches bedürfen
- Das zu verarbeitende Papierformat soll die Breite von DIN A6 und die Höhe von DIN A4 aufweisen, was in etwa dem Format von Zahlscheinen entspricht
- Papiereinzug und -auswurf sollen an der gleichen Öffnung stattfinden
- Das Einlegen und Herausnehmen des Einzelblattes muss ohne besondere Vorkenntnisse und auch mit eingeschränkten motorischen Fähigkeiten möglich sein
- Die bedruckte Zeile sollte sofort sichtbar sein, um dem/der Benutzer/-in sofort optisches Feedback zum Druckvorgang geben zu können
- Dadurch bedingt sollte die Druckrichtung von oben nach unten erfolgen. Daher muss das Papier zuerst vollständig eingezogen und anschließend im Laufe des Druckvorgangs Zeile für Zeile aus dem Drucker befördert werden

- Während des Druckvorganges sollen dem/der Benutzer/-in am Bildschirm Statusinformationen zum Druckvorgang angezeigt werden
- Das Gehäuse soll ein formschönes und ansprechendes Design aufweisen, da dies die Akzeptanz bei der Zielgruppe tendenziell positiv beeinflusst [27]

Um diesen Anforderungen gerecht zu werden, haben wir uns, wie in Kapitel 5 ausführlich beschrieben, für die Verwendung eines Thermo-Druckwerkes, welches ursprünglich für den industriellen Einsatz in Ticketdruckern oder Automaten konzipiert wurde, entschieden. Dieses Kapitel geht auch im Detail darauf ein, wie diese Anforderungen auf Hardware-Seite umgesetzt wurden.

Die Verwendung eines Industrie-Druckwerkes machte die Steuerung dieses Druckwerks über einen eigens entwickelten Druckertreiber von der PC-Seite aus notwendig, welcher zusätzlich Flexibilität im Hinblick auf die speziellen Anforderung wie anwendungsspezifische Bildschirm-Meldungen und Einzelblatt-Einzug erlaubt. Die Realisierung der speziellen Anforderungen auf Treiberseite ist im Detail in Kapitel 4 beschrieben.

3.3 Design und Gestaltung des Prototyps von Pavlovic

Das Design des Gehäuses wurde grundsätzlich an den Prototypen von Pavlovic angelehnt. Dieses Design wurde ausführlich mit einer Gruppe von Senioren/-innen evaluiert und ging aus mehreren vorgeschlagenen Modellen als das praktikabelste und das am besten akzeptierte Modell hervor.

Der Prototyp lag als 3D-Modell im Format des Programms „SketchUp Pro“ vor, und wurde mittels eines 3D Druckers (MakerBot Replicator 2) in verkleinerter Form ausgedruckt. Das 3D Modell ist in gerenderter Form in Abbildung 3.3.1 dargestellt.

Das 3D Modell des Onlinebanking-Belegdruckers weist Abmessungen von etwa 12,5 x 15 x 5,5 cm auf, wobei die Breite wesentlich eingeschränkt wurde, um Probleme beim 3D-Druck zu vermeiden.

Des Weiteren wurden, im Rahmen der Arbeit von Pavlovic, Exemplare der Prototypen mit Schaumkarton verstärkt und mit verschiedenen Designs versehen, wie in Abbildung 3.3.2 dargestellt.

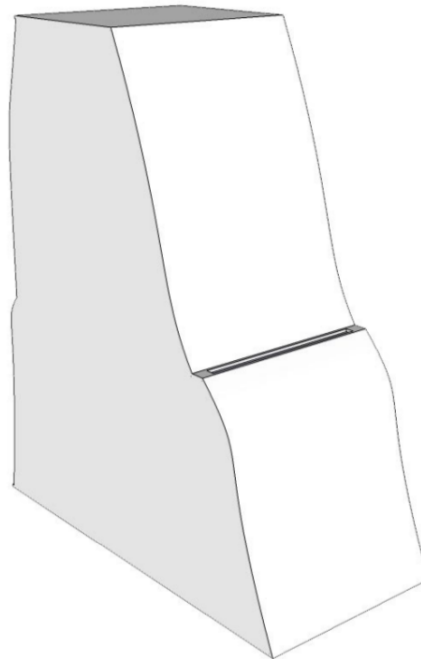


Abbildung 3.3.1: 3D Modell des Drucker-Gehäuses nach Pavlovic

3.4 Zusammenfassung der Studienergebnisse von Pavlovic

Um Rückmeldungen zum und Erfahrungen mit dem von ihm erdachten Design zu sammeln, führte Pavlovic in [17] ein etwa achtzig-minütiges Gruppeninterview mit fünf Teilnehmern/-innen durch, wobei zwei der Studienteilnehmer/-innen weiblich und drei männlich waren. Bei dem Interview wurden in [26] beschriebene Vorgehensweisen angewandt. Neben Fragen, welche sich konkret auf das Design des Belegdruckers bezogen, wurde auch versucht, die grundsätzliche Einstellung der Teilnehmer/-innen zu Online-Banking sowie den Status der Nutzung zu ergründen.

3.4.1 Technik-Nutzung durch die Studienteilnehmer/-innen:

Hier sollen daher kurz die wichtigsten Erkenntnisse dieser Studie zusammengefasst werden:

Mobiltelefone

Anhand der hohen Verbreitung von Mobiltelefonen (80% der Studienteilnehmer/-innen waren im Besitz eines Mobiltelefons) lässt sich der Schluss ziehen, dass Senioren/-innen bei einem klar erkennbaren Vorteil durchaus gewillt sind, sich mit neuen Technologien



Abbildung 3.3.2: Mit Schaumkarton verstärkte Gehäuse-Prototypen

auseinanderzusetzen. Da das Mobiltelefon als logische Weiterentwicklung des Festnetztelefons verstanden werden kann, ist der Zusatznutzen sofort erkenn- und begreifbar, wobei zusätzlich die Möglichkeit, in Notsituationen Unterstützung anzufordern, als wichtiger Pluspunkt gegenüber einem Festnetztelefon gewertet wurde - ein wichtiger Beweggrund, der auch in [25] angesprochen wird. Allerdings gab auch ein nicht vernachlässigbarer Anteil der Studienteilnehmer/-innen an, dass die kleinen Tasten sowie die kleine Schrift die Benutzung erschweren. Diese Schwierigkeiten waren auch der Grund, weshalb eine/einer der fünf Studienteilnehmer/-innen gänzlich auf die Verwendung von Mobiltelefonen verzichtete, beziehungsweise verzichten musste.

Personal Computers

PCs hatten bei den Studienteilnehmern/-innen mit 80% dieselbe Verbreitung wie Mobiltelefone, wobei sich im Einsatz der PCs große Unterschiede zeigten. Hohe Akzeptanz erfuhr die Nutzung des PCs zur Informationsbeschaffung, wobei auch hier von Schwierigkeiten mit schlechter Lesbarkeit berichtet wurde.

3.4.2 Erkenntnisse und Rückmeldungen zur Designstudie des Belegdruckers

Die Funktionsweise des Belegdruckers wurden den Studienteilnehmern/-innen anhand einer 3D-Animation veranschaulicht, die mit Anime Studio Pro erstellt wurde. Neben der

virtuellen Animation wurde den Studienteilnehmern/-innen nach Ablauf der Animation auch das fertige Druckresultat überreicht, um die Illusion eines tatsächlich erfolgten Ausdrucks zu vermitteln.

Für das Anstoßen des Ausdrucks wurden den Studienteilnehmern/-innen verschiedene Aufgaben aufgetragen, wie beispielsweise:

- Ausdrucken von Zahlungsanweisungen beim Online-Banking. Nachdem mit den Studienteilnehmern/-innen verschiedenste Arbeitsabläufe mittels des Online-Banking Portals der Bank Austria durchgeführt wurden, wurden die Studienteilnehmer/-innen gebeten, Zahlungsanweisungen auszudrucken.
- Ausdrucken einer Einkaufsliste

Das Drucken der Einkaufsliste wurde aufgrund attestierter zu geringer Flexibilität eher weniger positiv aufgenommen, wobei sich das Bild der Studienteilnehmer/-innen durch Gedankenspiele und Anregungen bezüglich zusätzlicher Möglichkeiten, die sich mit dem Belegdrucker ergeben würden, sich zumindest teilweise verbesserte.

Bei der Verwendung des Belegdruckers zum Drucken von Online-Banking-Bestätigungen, gab es sehr unterschiedliche Meinungen, welche primär davon geprägt waren, welchen Wert der/die Studienteilnehmer/-in Papierbelegen beimaß.

Zusammenfassend zieht Pavlovic folgendes Resümee aus der Studie:

„Die Einstellungen zu dem vorgestellten Onlinebanking-Belegdruckermodell sind zu einem großen Teil sehr positiv ausgefallen. Natürlich lagen die Schwierigkeiten bei zwei Befragten in den bereits bekannten Abläufen und der Frage, wozu brauchte ich nun sowas, ich habe ja bereits einen Drucker und der druckt mir alles, da ja heutzutage tatsächlich die meisten Bestätigungen, Fahrkarten, Buchungen etc. als Bestätigung in einem A4 Drucker gedruckt werden können. Doch der Bezug des angestrebten Onlinebanking-Belegdruckers auf die Bankangelegenheiten sowie weiteren Möglichkeiten des üblichen Ausdrucks von Gutscheinen, Eintrittskarten, Fahrkarten etc. hebt diesen Drucker von den anderen wesentlich ab.“

Software-Architektur

Mit einer detaillierten Beschreibung des Windows-Drucker-Subsystems beginnt dieses Kapitel, um einen Einblick in die Zusammenhänge innerhalb dieses komplexen Systems geben zu können. Abschnitt 4.1 beginnt mit einem groben Überblick, ehe auf das genaue Zusammenspiel der einzelnen Subsystem-Komponenten eingegangen wird.

Weiters wird in Abschnitt 4.1.3 schließlich der für den Belegdrucker gewählte Realisierungsweg im Detail erläutert, wobei die Realisierung den Umstand ausnützt, dass für viele Komponenten des Windows-Drucker-Subsystems bereits Basisimplementierungen existieren, die für das gewählte Design ausreichend flexibel und leistungsfähig sind. Daher ist oftmals die Konfiguration über Textdateien ausreichend und es musste beispielsweise kein Kernspace-Treiber entwickelt werden.

Den Abschluss des Kapitels bildet Abschnitt 4.2, welcher von der Software handelt, die auf dem Arduino-Leonardo Mikrocontroller-Board ausgeführt wird und schlussendlich die vom Windows-Druckertreiber generierten Daten empfängt und an die IFV-Controllerplatine weiterreicht. Die dabei verwendete State-Machine zur Unterscheidung der unterschiedlichen Kommandos, die vom PC an das Board gesendet werden sowie die Details zur Bild-datenübertragung werden dabei ebenso behandelt, wie die Fehlerbehandlung, welche notwendig ist, um dem Druckertreiber über den aktuellen Status des Druckwerks zu informieren.

4.1 Windows Druckertreiber

Um eine universelle Verwendung des Belegdruckers zu ermöglichen, wurde ein Windows Druckertreiber entwickelt. Dieser ermöglicht es jeder beliebigen Windows-Anwendung, über die normale Windows-Printing-API Ausdrücke anzustoßen. Es sind daher keine Anpassungen der Anwendungen notwendig, um den Belegdrucker ansprechen zu können.

Das Windows Druck-Subsystem besteht, ähnlich den Druck-Subsystemen der Unix-Welt wie LPR(ng) oder dem Common Unix Printing System (CUPS), aus einer Vielzahl von zueinander prinzipiell unabhängigen Komponenten, welche jeweils eine klar definierte Aufgabe im Ablauf eines Druckprozesses ausführen. Daher wird in diesem Abschnitt, bevor auf die spezialisierte Umsetzung des Belegdruckertreibers eingegangen wird, die Architektur des Windows Druck-Subsystems genauer erläutert. Dies dient unter anderem dazu, Design-Entscheidungen, welche im Verlauf des Entwicklungsprozesses getroffen wurden, anhand einer Erklärung der verschiedenen Implementierungsmöglichkeiten verständlicher zu machen. Da das Windows Druck-Subsystem mit Windows 2000 eine größere Überarbeitung erfahren hat, beschränkt sich die Beschreibung auf die Architektur welche seit Windows 2000 verwendet wird. Aufgrund des geringen Marktanteils von noch älteren Windows-Versionen (Windows 9x und NT \leq 4.0) und der Entscheidung von Microsoft, keine Updates mehr bezüglich kritischer Sicherheitslücken für diese Systeme zu veröffentlichen, hatte die Unterstützung älterer Windows-Versionen keine Priorität.

4.1.1 Überblick über das Windows Druck-Subsystem

Die zwei Hauptkomponenten des Windows Druck-Subsystems sind der „Print Spooler“ sowie verschiedene Druckertreiber. Aufgabe des Spoolers ist die Verwaltung der einzelnen Drucker-Jobs und das Management der Drucker-Warteschlange. Aufgabe der Druckertreiber ist es, die geräteunabhängigen Zeichenoperationen, welche von den Anwendungen über das *Graphics Device Interface* (GDI) oder über die *Windows Presentation Foundation* (WPF) generiert werden, in ein Format zu konvertieren, welches der anzusprechende Drucker verarbeiten kann. Im einfachsten Fall kann dies beispielsweise ein *Device Independent Bitmap* (DIB) sein, es ist jedoch auch möglich, Druckern Vektor-Grafik sowie Text direkt zu senden, sofern dies vom jeweiligen Drucker unterstützt wird. Weiters stellen die Druckertreiber Konfigurationsmöglichkeiten für Optionen wie beispielsweise Papierformat, Druckqualität oder Papierfach-Auswahl zur Verfügung.

Üblicherweise erfolgt die Verarbeitung eines lokalen Druckauftrags in folgenden Schritten:

- Eine Windows-Anwendung öffnet den Drucker, wobei verschiedene Optionen über ein User-Interface ausgewählt werden können, welches vom Druckertreiber implementiert wird
- Die Anwendung zeichnet über das GDI oder WPF Text und Grafiken, die gedruckt werden sollen
- Die Windows-GDI/WPF Implementierung zeichnet die von der Anwendung durchgeführten Zeichenbefehle auf
- Der Druckertreiber wandelt die aufgezeichneten Zeichenbefehle in ein für den Drucker verständliches Format um
- Der Drucker-Spooler sendet die vom Druckertreiber generierten Daten an den Drucker

Da viele Eigenschaften der Drucker oftmals ident sind, ist es nicht notwendig, für die Unterstützung eines neuen Druckers sämtliche vorher beschriebenen Komponenten vollständig neu zu implementieren. Windows enthält unterschiedliche Implementierungen, welche oftmals über Plugin-Mechanismen an die Eigenheiten eines spezifischen Druckers angepasst werden können. Daher ist nur für die Unterstützung von Spezifika des Druckers dediziert programmierter Programmcode notwendig, ansonsten kann auf eine große gemeinsame Code- und Funktionsbasis zurückgegriffen werden, die von Windows bereitgestellt wird.

4.1.2 Aufbau von Windows Druckertreibern

Die Aufgaben eines Windows Druckertreiber sind:

- Umwandeln der geräteunabhängigen GDI-Zeichenbefehle in ein, dem Drucker verständliches, Format
- Weiterreichen der dabei entstandenen Daten an den Drucker-Spooler
- Zur Verfügung stellen eines Konfigurations-Interfaces, welches zum Beispiel die Auswahl des Papierformats, die Anzahl der zu erstellenden Kopien oder die Druckqualität erlaubt.

Wie bereits erwähnt, werden mit Windows bereits unterschiedliche Druckertreiber ausgeliefert, welche durch Plugins und Konfigurationsdateien an die Spezifika eines Druckers

angepasst werden. So existiert neben dem Microsoft PostScript Treiber, welcher für Drucker konzipiert ist, die direkt Vektorgrafiken im PostScript-Format verarbeiten können, auch ein Treiber für Plotter, welche die „Hewlett-Packard Graphics Language“ (HPGL/2) unterstützen. Für die Unterstützung des Belegdruckers wurde der *Microsoft Universal Printing Driver* (UniDRV) als Ausgangsbasis benutzt, auf welchen im Folgenden näher eingegangen werden soll.

Windows unterscheidet zwischen GDI und *Open XML Paper Specification* (XPS) basierenden Druckertreibern, wobei der Hauptunterschied das verwendete Zwischenformat ist, welches der Druckertreiber zu verarbeiten hat. XPS ist ein neuerer Standard, vergleichbar mit dem, besonders in der Unix-Welt weit verbreiteten, PostScript. Besonders für den Fall, dass der zu unterstützende Drucker Vektorgrafiken direkt verarbeiten kann, bietet XPS den Vorteil, dass die, an den Drucker zu übertragende, Datenmenge geringer ausfällt.

Der Windows Universal Printing Treiber Der Windows Universal Printing Treiber ist ein generischer Druckertreiber, welcher für die Unterstützung von Druckern konzipiert ist, die selbst keine oder nur sehr eingeschränkte Raster-Fähigkeiten aufweisen und daher auf den Druckertreiber angewiesen sind, um Zeichenbefehle in ein Raster-Bild umzuwandeln. Umgangssprachlich werden Drucker, welche auf UniDrv-basierende Treiber aufbauen auch als „GDI Drucker“ bezeichnet, weil in diesem Fall üblicherweise die Konvertierung von Zeichenbefehlen zu Rastergrafik vom GDI durchgeführt wird. Da allerdings auch Drucker über das GDI angesprochen werden, die nicht auf UniDrv aufbauen, wird im weiteren Verlauf von der Verwendung der Bezeichnung „GDI Drucker“ abgesehen. Auch ist es mittels eigener Rendering Plugins möglich, UniDrv zu nutzen, ohne das GDI für das Umwandeln in Rastergrafik zu verwenden. Typischerweise wird UniDrv für die Unterstützung von Druckern im Consumer-Bereich verwendet und hat daher eine sehr hohe Verbreitung. UniDrv ist ein vollständiger Druckertreiber und kann ohne dem Schreiben von zusätzlichem / druckerspezifischem Code verwendet werden, erlaubt allerdings auch durch Plugin-Schnittstellen die Einbindung von Treiber-eigenem Code.

In Abbildung 4.1.1 sind die einzelnen Komponenten von UniDrv dargestellt. Die Konfiguration erfolgt dabei über sogenannte GPD-Dateien, welche auch als „Minitreiber“ bezeichnet werden. Diese Minitreiber beschreiben den zu unterstützenden Drucker näher, indem sie beispielsweise die unterstützten Papierformate, die möglichen Qualitätseinstellungen sowie Optionen zum Papiereinzug enthalten. Mittels logischer Ausdrücke, lassen sich sogar Regeln spezifizieren, welche Optionen miteinander kombinierbar sind,

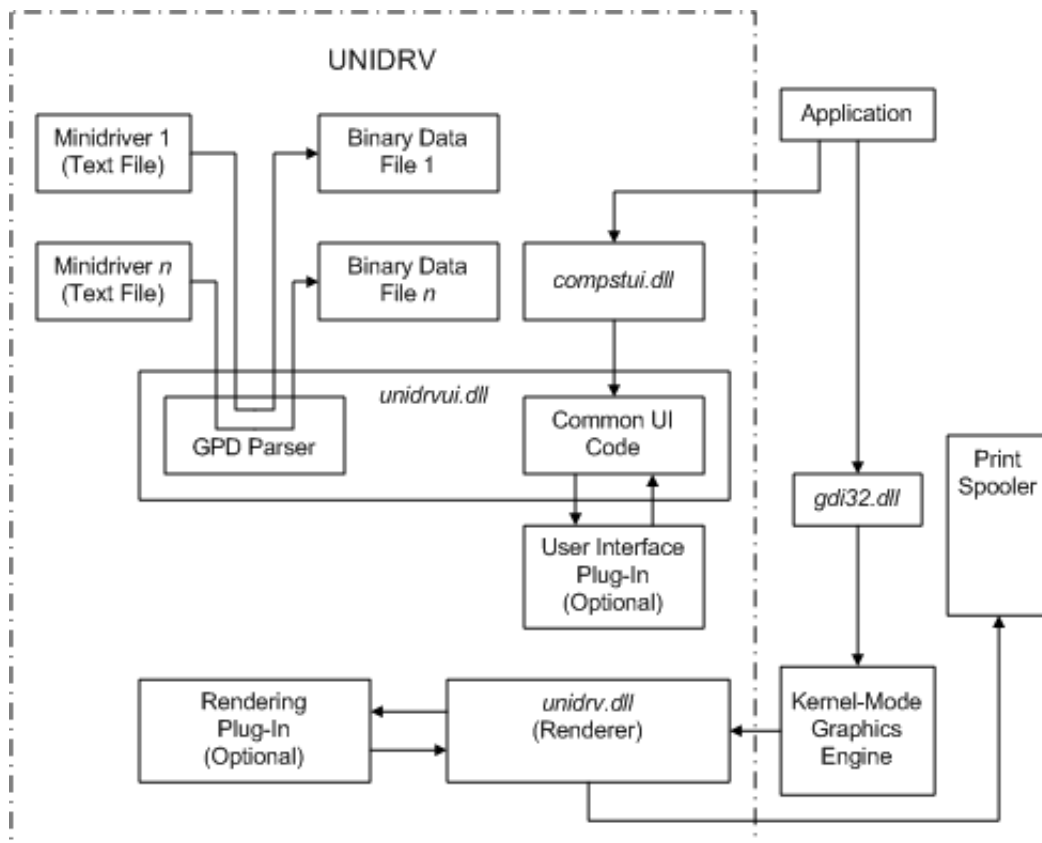


Abbildung 4.1.1: Komponenten des Microsoft Universal Printing Driver
 Quelle: [https://msdn.microsoft.com/en-us/library/windows/hardware/ff562889\(v=vs.85\).aspx](https://msdn.microsoft.com/en-us/library/windows/hardware/ff562889(v=vs.85).aspx)
 aufgerufen 2016-07-06

beziehungsweise sich gegenseitig ausschließen. Anhand dieser Optionen wird unter anderem das Ausgabeformat beeinflusst, das an den Drucker gesendet wird. Weiters stellt UniDrv Dialogelemente zur Konfiguration zur Verfügung, die anhand der, durch die GPD-Dateien festgelegten Parameter, aufgebaut werden. Durch optionale *Graphical User Interface* (GUI) Plugins ist es möglich, diese Dialogelemente um Funktionalitäten zu erweitern, welche durch UniDrv nicht abgedeckt werden. Wie bereits beschrieben, wandelt der UniDrv-Treiber standardmäßig die GDI-Zeichenoperationen in eine Rastergrafik um. Dies erfolgt durch die Renderer-Komponente, wobei auch hier die Möglichkeit besteht, das Rendering, über ein Rendering-Plugin, gerätespezifisch zu implementieren. Nachdem die Daten in einem, mit dem Drucker kompatiblen, Format vorliegen, werden sie an den Print-Spooler zur weiteren Verarbeitung weitergeleitet. Wie aus dem Diagramm ersichtlich, erfolgt nur die Aufbereitung in ein druckerverständliches Format in UniDrv, die Kommunikation mit dem Gerät allerdings nicht.

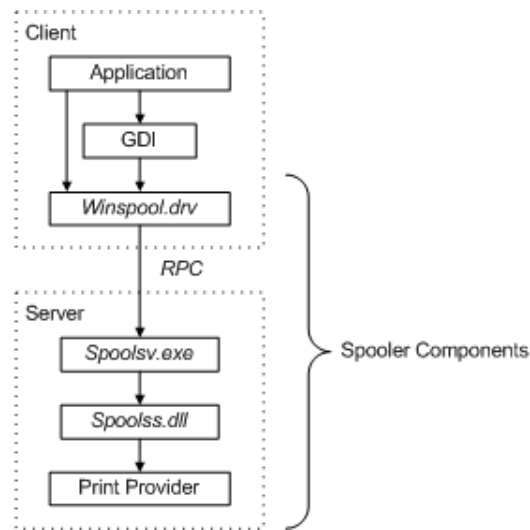


Abbildung 4.1.2: Komponenten des Windows Print Spoolers

Quelle: [https://msdn.microsoft.com/en-us/library/windows/hardware/ff556447\(v=vs.85\).aspx](https://msdn.microsoft.com/en-us/library/windows/hardware/ff556447(v=vs.85).aspx) aufgerufen 2016-07-06

Der Windows Drucker Spooler Die Aufgabe des Windows Print Spoolers ist einerseits, die laufenden Druckaufträge in Druckerwarteschlangen zu verwalten und andererseits zu entscheiden, wohin zu druckende Daten gesandt werden sollen. Das Ziel kann dabei entweder eine Schnittstelle am lokalen Rechner sein oder ein entfernter Print-Server. Wie in Abbildung 4.1.2 dargestellt, basiert das Windows Print Spooler System auf einer Client/Server-Architektur, die über *Remote Procedure Calls* (RPC) implementiert ist. Wird auf einem lokalen Drucker gedruckt, so werden Client und Server auf derselben Maschine ausgeführt.

Die Kommunikation mit dem Drucker, also das Ansprechen der eigentlichen Schnittstelle und das Generieren von vom Drucker interpretierbaren Datenpaketen erfolgt, wie in Abbildung 4.1.4 ersichtlich, durch sogenannte Language- und Port-Monitors. Language-Monitors sind dabei für die Umsetzung von Befehlen des Druck-Subsystems einerseits (z.B. Druckkopf reinigen, Druckauftrag starten) sowie für die Übermittlung von Status-Information des Druckers an das Druck-Subsystem andererseits zuständig. Eine Default-Implementierung, welche die *Printer Job Language* (PJP) umsetzt, wird mit dem *Windows Driver Kit* (WDK) als Beispielimplementierung mitgeliefert, es ist jedoch wie im Architekturdiagramm 4.1.4 ersichtlich auch möglich, davon unabhängige eigene Language Monitor Treiber als *Dynamic Link Library* (DLL) zu registrieren.

Port Monitors übernehmen schließlich die eigentliche Datenübertragung, indem sie das Bindeglied zwischen den Hardware-Schnittstellen, auf welche über den Kernel-Mode

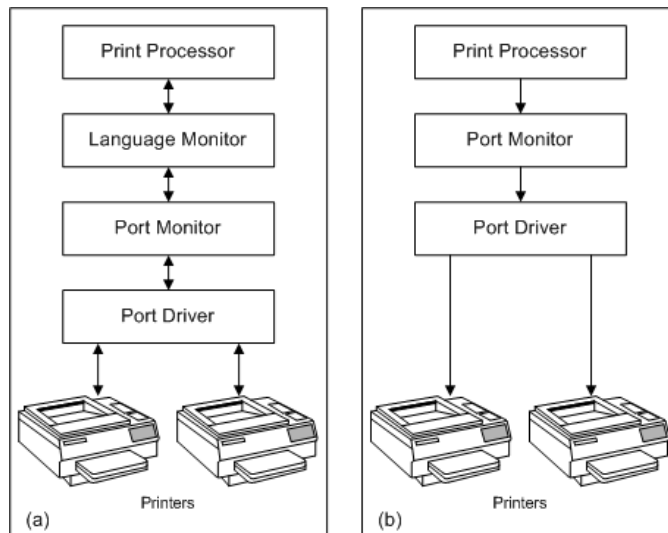


Abbildung 4.1.3: Datenfluss vom Print-Processor bis hin zum Drucker

Quelle: [https://msdn.microsoft.com/en-us/library/windows/hardware/ff556447\(v=vs.85\).aspx](https://msdn.microsoft.com/en-us/library/windows/hardware/ff556447(v=vs.85).aspx) aufgerufen 2016-07-06

Port Driver Stack zugegriffen wird und dem Language-Monitor, welcher die Daten in ein für den Drucker verständliches Format aufbereitet, bilden. Die Hauptaufgabe eines Port Monitors ist daher die Übertragung der Daten. Wie in Abbildung 4.1.3 ersichtlich, ist die Verwendung eines Language Monitors dabei optional. Sofern der Drucker die vom Print-Processor generierten Daten direkt verarbeiten kann, ist auch eine direkte Weiterleitung der Daten über den Port Monitor zulässig.

Weiters ist die Kombination von Language- und Port-Monitor in einer einzigen DLL zulässig. Dabei müssen allerdings die User-Interface-Teile, welche den Language-Port Teil betreffen, in eine separate DLL ausgelagert werden. Generell wird empfohlen, von DLLs, welche direkt vom Windows Druck-Subsystem eingebunden werden, keine Dialogelemente zu öffnen. Windows stellt für diesen Zweck extra vorgefertigte Elemente zur Verfügung, um zum Beispiel den Druckfortschritt grafisch darzustellen oder Fehlermeldungen auszugeben, die mittels eines RPC-Mechanismus angesprochen werden müssen.

Ein Port-Monitor kann mittels eines *Setup Information File* (INF) im System registriert werden. Die Registrierung kann dabei entweder über die API des Windows Druck-Subsystems (Funktion `AddMonitor`), oder über den Dialog zum Hinzufügen von Hardware vorgenommen werden. Nachdem der Port registriert wurde, steht er als neuer Typ in der Combobox „Port hinzufügen“ des „Drucker hinzufügen“ Dialogs zur Verfügung, wie in Abbildung 4.1.5 ersichtlich.

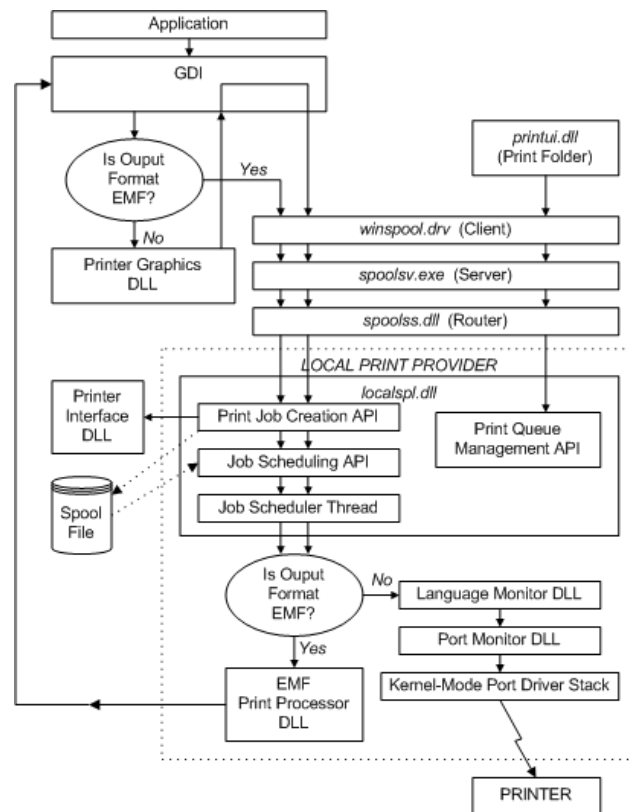


Abbildung 4.1.4: Spooling-Ablauf bei Druck auf einem lokalen Drucker
 Quelle: [https://msdn.microsoft.com/en-us/library/windows/hardware/ff556486\(v=vs.85\).aspx](https://msdn.microsoft.com/en-us/library/windows/hardware/ff556486(v=vs.85).aspx) aufgerufen 2016-07-06

4.1.3 Realisierung und Design des Belegdrucker-Treibers

In diesem Abschnitt wird auf die spezifische Implementierung des Treibers für den Belegdrucker und auf seine einzelnen Komponenten eingegangen.

Bei der Realisierung des Belegdrucker-Treibers gab es folgende Ziele:

- Möglichst einfache Implementierung, um die Wartbarkeit auch durch nachfolgende Entwickler/-innen zu ermöglichen
- Soweit sinnvoll hohe Portabilität, um eventuell später weitere Plattformen, wie Linux oder OSX, unterstützten zu können
- Um die Kompatibilität mit einer breiten Basis unterschiedlicher Windows-Versionen sicherzustellen, möglichst eingeschränkte Verwendung von spezifischen Schnittstellen des Windows-Drucker-Subsystems

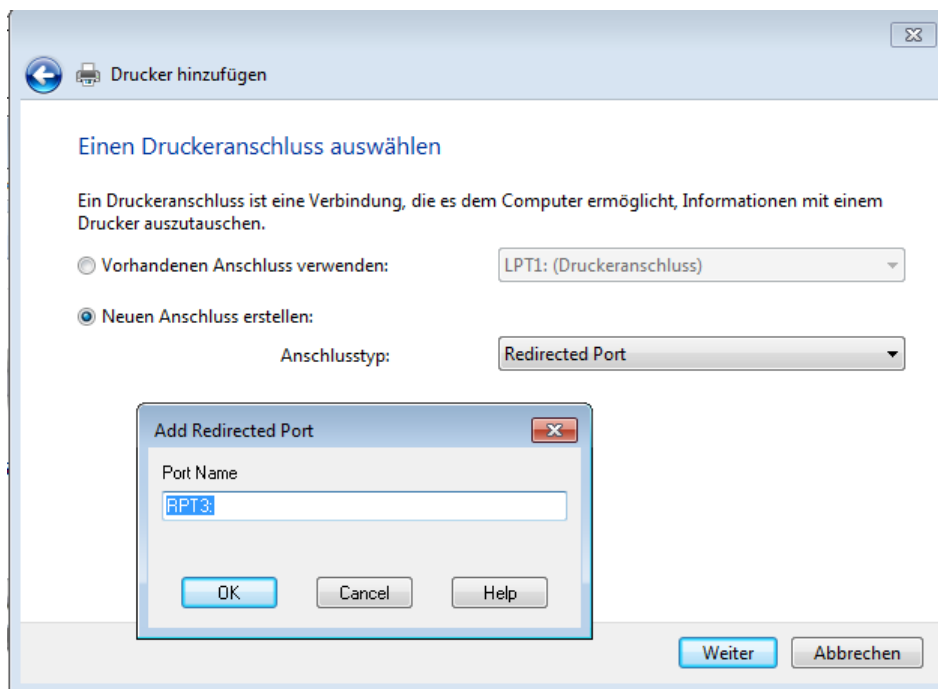


Abbildung 4.1.5: Dialog zum Anlegen eines Druckeranschlusses

In der ersten Entwicklungsphase wurde für Tests der Arduino-Software auf die serielle Schnittstelle mit einem stand-alone Programm zugegriffen, wobei der Zugriff in diesem Fall über die portable File-API erfolgt. Die Schnittstelle wird dabei implizit und für die Anwendung vollständig transparent von Windows über den Kernel-Mode Port Driver Stack angesprochen. Im weiteren Entwicklungsverlauf bewährte sich dieses Konzept aufgrund der einfachen Struktur, der guten Debugging-Eigenschaften sowie des portablen Konzepts so gut, dass entschieden wurde, falls möglich, statt einer in den Print-Prozess eingebundenen Dynamic Link Library (DLL), weiterhin einen separaten und entkoppelten Prozess für die Datenübertragung zu benutzen. Ein weiterer Entscheidungsgrund für einen separaten Prozess war, dass die Entwicklung hauptsächlich unter dem Betriebssystem Linux erfolgte, was bei Erstellung einer Port Monitor DLL durch Abhängigkeiten zum Windows Druck-Subsystem nicht mehr möglich gewesen wäre.

Generic Printer Description (GPD) Die Parametrierung von UniDRV, erfolgt wie in Abschnitt 4.1.2 erläutert, über sogenannte *Generic Printer Description* (GPD) -Dateien. Die für den Belegdrucker erstellte GPD-Datei ist mittels der Listings 4.1 und 4.2 abgebildet.

Im Folgenden wird nur auf die verwendeten Parameter eingegangen, eine vollständige Parameterliste ist auf der Dokumentationsseite¹ des *Microsoft Developer Network* (MSDN) verfügbar. Folgende Parameter wurden verwendet:

Algorithmus 4.1 Erster Teil des General Printer Description File, mit welchem UniDrv für den Belegdrucker parametrisiert wird

```
1 *GPDSpecVersion: "1.0"
2 *Include: "StdNames.gpd"
3 *ResourceDLL: "unires.dll"
4 *GPDFileVersion: "1.0"
5 *ModelName: "Belegdrucker"
6 *MasterUnits: PAIR(200, 200)
7 *PrinterType: PAGE
8 *Feature: Orientation
9 {
10     *rcNameID: =ORIENTATION_DISPLAY
11     *DefaultOption: PORTRAIT
12     *Option: PORTRAIT
13     {
14         *rcNameID: =PORTRAIT_DISPLAY
15     }
16 }
17 *Feature: InputBin
18 {
19     *rcNameID: =PAPER_SOURCE_DISPLAY
20     *DefaultOption: MANUAL
21     *Option: MANUAL
22     {
23         *rcNameID: =MANUAL_FEED_DISPLAY
24     }
25 }
26 *Feature: Resolution
27 {
28     *rcNameID: =RESOLUTION_DISPLAY
29     *DefaultOption: Standard
30     *Option: Standard
31     {
32         *Name: "200 x 200" =DOTS_PER_INCH
33         *DPI: PAIR(200, 200)
34         *TextDPI: PAIR(200, 200)
35         *SpotDiameter: 100
36         *Command: CmdSendBlockData { *Cmd: "" }
37     }
38 }
```

-
- **GPDSpecVersion** : "1.0": Gibt die Version der GPD-Spezifikation an, nach welcher die GPD-Datei erstellt wurde und muss zwingend als erste Eigenschaft in der GPD-Datei, noch vor dem ersten Kommentar, enthalten sein. Die GPDSpecVersion „1.0“ wird ab Windows 2000 unterstützt und unterstützt alle Funktionen, die der Belegdrucker benötigt.

¹<https://msdn.microsoft.com/en-us/library/windows/hardware/ff551750%28v=vs.85%29.aspx>

- `Include : " StdNames . gpd "`: Inkludiert die Datei `StdNames.gpd`, welche vordefinierte Typen besitzt. In neueren GDP-Versionen ist die `Include`-Direktive durch die Einführung eines Präprozessors obsolet. Da dies allerdings die Rückwärtskompatibilität negativ beeinflussen würde, verwendet der Belegdrucker-Treiber weiterhin die alte `Include`-Direktive.
- `ResourceDLL : " unires . dll "`: `UniDrv` Minitreiber können auf sogenannte „Ressource DLLs“ zurückgreifen, um zusätzliche Daten, wie Font-Dateien oder Icons, mit dem Treiber auszuliefern, wobei das `DLL`-Format in diesem Fall nur als Container-Format für die eigentlichen Ressourcen fungiert (ähnlich einer `Zip`-Datei). Da der Belegdrucker-Treiber keine eigene Ressourcen-DLL verwendet, wird die `UniDRV` DLL verwendet, in welcher beispielsweise Namen für Papierformate in lokalisierter Form enthalten sind.
- `ModelName : " Belegdrucker "`: Selbstredend spezifiziert das Property „`ModelName`“ den Modellnamen des Belegdruckers
- `MasterUnits : PAIR (200 , 200)`: Die `Master-Units` stellen die „Universal“-Einheit in vertikaler sowie horizontaler Richtung dar, in welcher anschließend die Druckereigenschaften beschrieben werden. Gebildet werden sie aus dem kleinsten gemeinsamen Vielfachen der unterstützten Auflösungen in *Dots Per Inch* (DPI). Der Belegdrucker druckt über eine Breite von 105mm (4,13386 inches) mit einer Auflösung von 832 Punkten, was einer Auflösung von 201.26 DPI entspricht, welche zur Vereinfachung auf 200 PPI abgerundet wurde.
- `PrinterType : PAGE`: Diese Eigenschaft wurde von Microsoft nicht dokumentiert. Einzig die wählbaren Werte „`TTY`“, „`Serial`“ und „`Page`“ sind bekannt, wobei der schlussendlich gewählte Wert „`Page`“ durch Experimentieren ermittelt wurde.
- `Feature : Orientation`: Mit dem Feature „`Orientation`“ können die möglichen Papier-Orientierungen spezifiziert werden. Der Belegdrucker unterstützt nur den `Portrait`-Modus, welcher daher auch als `Default`-Wert angegeben ist.
- `Feature : Resolution`: Das Feature `Resolution` beschreibt die möglichen Auflösungen, welche der Drucker unterstützt und welche dem/der Benutzer/-in im Druckdialog zur Auswahl angezeigt werden. Für den Belegdrucker wurde nur die native Auflösung des Druckkopfs von etwa 200x200 DPI implementiert und ist daher die einzig auswählbare Option. Der Unterparameter „`SpotDiameter`“ bezieht sich auf die Größe eines Druckpunktes relativ zur Auflösung, wobei es sich um eine prozentuale

Angabe handelt. Ein Wert von 100% bedeutet, dass ein Druckpunkt genau so groß ist wie die Auflösung des Druckers. Zusätzlich muss zwingend ein Parameter „Command“ angegeben werden, welcher UniDrv veranlasst, beim Start eines neuen Blocks eine Kommando-Sequenz in den Byte-Stream einzufügen. Da für den Belegdrucker die Kommunikation mit dem Gerät über den Data Transfer Process erfolgt, wird dieser Unter-Parameter, da zwingend erforderlich, zwar angegeben, er bleibt jedoch leer - womit keine extra Kommando-Bytes in den Datenstrom eingefügt werden.

Algorithmus 4.2 Zweiter Teil des General Printer Description File, mit welchem UniDrv für den Belegdrucker parametrisiert wird

```
1 *Feature: PaperSize
2 {
3     *rcNameID: =PAPER_SIZE_DISPLAY
4     *DefaultOption: LETTER
5     *Option: LETTER
6     {
7         *rcNameID: =RCID_DMPAPER_SYSTEM_NAME
8         *PrintableArea: PAIR(825, 1600)
9         *PrintableOrigin: PAIR(28, 28)
10    }
11 }
12
13 *Feature: ColorMode
14 {
15     *rcNameID: =COLOR_PRINTING_MODE_DISPLAY
16     *DefaultOption: MONO
17
18     *Option: MONO
19     {
20         *rcNameID: =MONO_DISPLAY
21         *DrvBPP: 1
22         *DevNumOfPlanes: 0
23         *DevBPP: 0
24         *IPCallbackID: 1
25         *Color?: FALSE
26     }
27 }
28
29 *Command: CmdCR: ""
30 *Command: CmdLF: ""
31 *Command: CmdFF: ""
32
33 *RotateCoordinate?: FALSE
34 *RotateFont?: FALSE
35 *RotateRaster?: FALSE
```

- Feature : PaperSize: Mit dem Feature „PaperSize“ werden die unterstützten Papierformate in Master-Units angegeben, welche anschließend dem/der Benutzer/-in im Druckdialog zur Auswahl angeboten werden. Das Thermopapier des Belegdruckers ist 112mm breit, wobei 105mm bedruckt werden können - woraus folgt, dass auf jeder Seite ein 3,5mm breiter Rand eingehalten werden muss. Dies entspricht einem bedruckbaren Bereich von etwa 825x1600 Master-Units (200 Master-Units pro Inch), wobei der bedruckbare Bereich jeweils 28 Master-Units in vertikaler und horizontaler Position versetzt ist.
- Feature : ColorMode: Das Feature „ColorMode“ spezifiziert die unterstützten Farbtiefen des Druckers. Das Thermo-Druckwerk des Belegdruckers unterstützt nur 1-Bit Farbtiefe (Schwarz/Weiß).
- Command : CmdCR/CmdLF/CmdFF: Mit diesen Kommando-Folgen wird UniDrv instruiert, bei gewissen Steuerzeichen keine besonderen Kommandos in den Druckerdatenstrom einzufügen.
- RotateCoordinate / RotateFont / RotateRaster: Gibt an, dass der Drucker selbst keine Funktionen zur gedrehten Darstellung von Text bzw. Grafikdruck hat. Diese Parametrierung wurde bewusst, aufgrund der einfacheren Implementierung, gewählt, obwohl der Druck-Prozessor der Controllerplatine in der Lage ist, sowohl Grafiken als auch Text, gedreht zu drucken, da die zusätzliche Rechenzeit auf aktuellen PC-Prozessoren vernachlässigbar ist.

Windows INF-Datei Neben der GPD-Datei besteht der Windows Druckertreiber zusätzlich noch aus einer INF-Datei.

Diese Datei enthält folgende zusätzliche Meta-Informationen:

- Betriebssystem-Version für welche der Treiber geeignet ist
- Architekturen, welche von dem Treiber unterstützt werden (beispielsweise x86, amd64, ia64)
- Um welche Art von Treiber es sich handelt, in diesem Fall um die Klasse „Printer“
- Das Erstellungsdatum sowie die Versionsnummer des Treibers
- Den/die Herausgeber/-in / Ersteller/-in des Treibers

- Datei-Listing, das angibt, welche Dateien zu diesem Treiber gehören. Dieses Listing wird bei der Installation des Treibers durch Windows ausgewertet, um festzulegen, welche Dateien in das System-Verzeichnis kopiert werden sollen

Da es sich bei dem Belegdrucker um kein Plug&Play-Gerät handelt, muss Windows die Geräteinformationen aus der INF-Datei beziehen. Diese kann entweder manuell bei der Drucker-Installation mittels Dialog ausgewählt werden, oder automatisiert über die Windows-API registriert werden - was beispielsweise bei automatisierten Installationsroutinen Sinn ergeben würde.

Redirection Port Monitor Um mittels eines separaten Prozesses die Kommunikation mit dem Drucker durchführen zu können, wird eine Port Monitor Implementierung benötigt, welche die weiterzuleitenden Daten aus dem Kontext beziehungsweise Prozess des Local Print Provider heraus an eine separate Anwendung umleitet, da eine derartige Funktionalität ohne Umwege über Dateien von Microsoft standardmäßig nicht vorgesehen ist.

Eine derartige Implementierung wird von RedMon, also dem Redirection Port Monitor, bereitgestellt ². Bei RedMon handelt es sich um ein Open-Source Projekt, das unter *GNU General Public License* (GPL) verfügbar ist, und einen Port Monitor für Windows 2000 - Windows 7 zur Verfügung stellt. RedMon ermöglicht es, virtuelle Drucker-Ports anzulegen und die Daten, welche über diesen Port transportiert werden, an die Standardeingabe eines beliebigen Programms, das pro Datenübertragung separat gestartet wird, weiterzuleiten. Üblicherweise wird diese Funktionalität gemeinsam mit GhostScript genutzt, um bei Druckern, welche nativ keine Unterstützung für die PostScript-Druckersprache aufweisen, eine PostScript-Emulation realisieren zu können.

Data Transfer Process Die von RedMon abgefangenen Daten werden an den *Data Transfer Process* (DTP) weitergeleitet, einer Stand-Alone-Anwendung, die aus den ursprünglichen Funktions- und Schnittstellentests hervorgegangen ist. Der DTP übernimmt dabei folgende Aufgaben:

- Übertragung der von RedMon empfangenen Daten auf das USB -> Seriell Interface des Arduino
- Implementierung einer eigens entworfenen Flusskontrolle

²<http://pages.cs.wisc.edu/~ghost/redmon/> - Accessed 2015-05-23

- Grafische Anzeige von Druckfortschritt, Statusmeldungen und Anweisungen

Der DTP wurde in der Sprache C++ unter der Verwendung der, für viele Plattformen verfügbaren, Klassenbibliothek QT erstellt. Da QT sowohl den Zugriff auf die serielle Schnittstelle als auch die Elemente der grafischen Oberfläche zur Verfügung stellt, lässt sich der Quellcode des DTP unverändert sowohl für Windows als auch Linux und OSX kompilieren und verwenden, wobei einzig die Anbindung an das betriebsystemspezifische Drucker-Subsystem zusätzlich erfolgen müsste.

Als Teil von QT wird eine eigene, integrierte Entwicklungsumgebung mit dem Namen „QT Creator“ mitgeliefert. Diese bietet neben einem Editor zum Editieren des Quellcodes auch von Microsoft Visual Studio gewohnte Funktionen wie ein grafisches Frontend zur Fehlersuche und ist wie Visual Studio projektbasiert. Für Windows gibt es neben Versionen, welche die kostenpflichtigen Microsoft-C++-Compiler verwenden auch eine Version, die die *Minimalist GNU for Windows* (MinGW) Portierung der *GNU Compiler Collection* (GCC) mitliefert.

QT selbst ist unter der *GNU Lesser General Public License 3* (LGPL3) Lizenz verfügbar und kann somit kostenlos sowohl für kommerzielle als auch nichtkommerzielle Projekte eingesetzt werden, ebenso ist die Entwicklungsumgebung QT Creator kostenlos erhältlich. Daher konnte durch die Verwendung der QT-Bibliothek sowie der IDE „QT Creator“ eine portable Softwarelösung erstellt werden, welche im Gegensatz zu einer auf Microsoft Visual Studio basierenden Lösung keinerlei kostenpflichtige Tools voraussetzt.

4.2 Firmware

Die Embedded-Software welche am Arduino ausgeführt wird, übernimmt unter anderem folgende Aufgaben:

Initialisierung der Peripheriegeräte

Direkt nach dem Power-up des Arduino Leonardo wird die Funktion `setup()` aufgerufen, in welcher folgende Initialisierungsschritte ausgeführt werden:

- Die jeweiligen Eingangs- und Ausgangspins werden als solche konfiguriert
- Die Pins `!feed` und `!bfeed` der IFV101-01B-E Controller-Platine werden anschließend auf „high“ gesetzt, da diese low-aktiv sind und direkt nach dem Power-Up keine Feed-Aktion gewünscht wird

- Der Relaisausgang zur Steuerung der Stromversorgung des Leistungsteils (Heizelemente des Druckkopfs) wird mit low initialisiert. Damit wird sichergestellt, dass aus dem NIMH-Akku nur Leistung entnommen wird, wenn wirklich gedruckt wird
- Konfiguration des PWM-Verhältnisses des Servos, welcher für den Papiereinzug verwendet wird
- Stellen des Servos in seine Ausgangsposition
- Initialisieren der seriellen Schnittstelle sowohl zwischen Arduino und PC („Serial“) als auch zwischen IFV101-01B-E Controller-Platine und Arduino („Serial1“) mit 115200 baud/s, keine Parität sowie 1 Stopp-Bit (8N1)

State-Machine

Nach der Initialisierung wartet der Arduino auf Steuersignale vom PC. Dies wurde mittels einer Endlosschleife realisiert, welche eine simple State-Machine implementiert.

Ein Befehl ist als einzelnes *American Standard Code for Information Interchange* (ASCII)-Zeichen charakterisiert, welches zum Arduino gesendet wird und diesen, je nach gesendetem Zeichen, in einen gewissen Status versetzt. Unbekannte Zeichen werden ohne Konsequenzen verworfen. Folgende Befehle sind definiert:

- 'R': Setzt den Relais-Ausgang auf „high“. Somit wird das Druckwerk sowie der Servo mit dem NIMH-Akkumulator verbunden. Nach einer Pause, welche notwendig ist, um das Schließen des Relais abzuwarten, wird der Servo erneut konfiguriert (PWM und Ausgangsstellung). Anschließend wird der Servo aktiviert und der !feed-Pin der Controllerplatine auf „low“ gesetzt, um den Fehlerstatus des Controllers durch die bisher fehlende Versorgungsspannung des Druckwerkes zu resetieren. Nachdem das Druckwerk eine Feed-Operation durchgeführt hat, wird der Servo wieder in die Ausgangsstellung gefahren und die Controllerplatine ist bereit, Druckbefehle entgegenzunehmen.
- 'r': Setzt den Relais-Ausgang auf „low“. Somit wird das Druckwerk und der Servo nicht weiter mit Spannung versorgt.
- 'S': Der Servo wird auf jene Position gestellt, in welcher die Kraft der Federn im Druckwerk kompensiert wird und ein Drehen der Einzugwalze auch ohne Papier möglich ist.

- 's': Der Servo wird in die Ausgangsposition gestellt, in welcher er keine Kraft auf das Druckwerk ausübt.
- 'f': Backward-Feed des Druckwerks. Dabei wird das Papier entgegen der Druckrichtung bewegt beziehungsweise eingezogen. Die Anzahl der Zeilen um welche das Papier bewegt wird, wird als 2 Byte Parameter übergeben.
- 'F': Forward-Feed: Dabei wird das Papier in Druckrichtung bewegt beziehungsweise eingezogen. Die Anzahl der Zeilen, um welche das Papier bewegt wird, wird ident wie beim Backward-Feed als 2 Byte Parameter übergeben.
- 'E': Aktiviert so lange den Papiereinzug entgegen der Druckrichtung (Digitalwert 0 auf Pin !BFEED der IFV101-01B-E Controller-Platine), bis die Lichtschranke vor dem Papiereinzug wieder geöffnet wird. Dieses Steuerkommando wird als Teil des Papiereinzug ausgeführt, bei welchem das Papier so lange in den Drucker befördert werden soll, bis sich nur mehr ein kleiner, für den Transport notwendiger Teil, im Druckwerk befindet.
- 'I': Daten werden vom PC übertragen und an die Controller-Platine weitergereicht. Dies ist mit Abstand der komplexeste Befehl, auf welchen im folgenden Absatz genauer eingegangen wird.
- 'T': Gibt einen Ton mit konfigurierbarer Frequenz und Dauer auf dem im Drucker verbauten Lautsprecher aus.

Übertragung Bilddaten

Das Kommando 'I' initiiert einen Transfer von Daten vom PC direkt zur Controllerplatine. Auch wenn das Kommando aktuell nur für den Bildtransfer genutzt wird (daher der Character 'I' für „Image“), können damit beliebige Datentransfers vom PC zur Controllerplatine des Druckwerks durchgeführt werden. Der Arduino arbeitet in diesem Modus eigentlich nur als Umsetzer zwischen der USB-Schnittstelle des PCs und der seriellen Schnittstelle mit TTL-Pegeln der Controllerplatine. Hintergrund dieser Design-Entscheidung war es, durch Auslagerung des Schrittes der Protokollgenerierung auf den PC, größtmögliche Flexibilität auch hinsichtlich Erweiterbarkeit im Betrieb zu erreichen und den Entwicklungsanteil auf der Arduino-Plattform, mit ihren beschränkten Debugging-Möglichkeiten und sehr beschränkten Hardware-Ressourcen, so gering wie möglich zu halten.

Nach dem Kommando-Character wird zuerst die Länge der an die Controllerplatine zu übertragenden Daten als 3-Byte Integer-Wert übermittelt. Anschließend werden die Daten in Paketen zu je 8 Byte von der USB-Schnittstelle gelesen und über die serielle Schnittstelle an die Controllerplatine übertragen, wobei am Arduino eine manuelle Flusskontrolle implementiert wurde. Die Flusskontrolle ist notwendig, da das Thermodruckwerk keine konstante Druckgeschwindigkeit aufweist. Die Controller-CPU würde eine per Software realisierte Flusskontrolle unterstützen - würde die Controllerplatine also mittels Pegelwandler direkt mit der seriellen Schnittstelle eines PC verbunden werden, wären keine Anpassungen notwendig und die Platine würde die Daten so schnell entgegennehmen, wie das Druckwerk die Daten verarbeiten kann. Da allerdings das USB-Interface des Arduino, beziehungsweise dessen API für die seriellen Schnittstellen, keine Flusskontrolle unterstützt, muss stattdessen die Flusskontrolle mittels eines eigens entworfenen Protokolls abgewickelt werden.

Algorithmus 4.3 Implementierung der Flusskontrolle des DTP

```
1 void waitForReadyArduino(int srFD) {
2     uint8_t ackStatus = 0;
3     int bytesRead = 0;
4
5     struct pollfd fds;
6     fds.fd = srFD;
7     fds.revents = POLLIN;
8
9     // Loop as long as there are no ack-bytes left
10    // and the last buffered one was a ready-byte
11    while((bytesRead = read(srFD, &ackStatus, 1)) > 0 || ackStatus != '+') {
12        poll(&fds, 1, 200);
13    }
14 }
```

Die Controllerplatine ist mit einem 256 Byte großen Eingangspuffer ausgestattet. Ist dieser mit mehr als 223 Byte gefüllt, so wird der BUSY-Ausgang der Controllerplatine auf „high“ gesetzt. Anhand dieses BUSY-Ausgangs wird von der Arduino-Software festgestellt, ob die Controller-Platine bereit ist, Daten zu empfangen. Wenn sich eine Status-Änderung ergibt, so wird diese dem PC über den seriellen Rückkanal mitgeteilt. Wenn sich in der Zwischenzeit mehrere Status-Änderungen im Empfangspuffer der seriellen Schnittstelle angesammelt haben, so wird nur das letzte Signal ausgewertet.

Um nicht durch Polling unnötig CPU-Zyklen zu verschwenden, wird unter Linux, für den Fall, dass auf das „Ready“-Signal des Arduino gewartet werden muss, auf die Verfügbarkeit von neuen Daten mittels der „poll“-Funktion gewartet. Diese versetzt den aktuellen Prozess / Thread in den Ruhezustand, bis neue Daten von der seriellen

Schnittstelle gelesen werden können. Eine vereinfachte Version, bei welcher ein gelesenes '+'-Zeichen für „ready“ steht, ist in Listing 4.3 enthalten, um dieses Verhalten und dessen Implementierung zu veranschaulichen. Sind über 223 Byte im Empfangspuffer und wird somit das BUSY-Signal der Controllerplatine „high“, so sendet der Arduino kein ready-Signal mehr zum PC, welcher daraufhin den Datentransfer pausiert. Sobald der Empfangspuffer unter 191 Byte enthält, wird das BUSY-Signal wieder auf „low“ gesetzt, der Arduino sendet wieder ready-Signale und der Datentransfer wird wieder aufgenommen. Der Sendebeginn vor einer vollständigen Entleerung des Empfangspuffers stellt sicher, dass das Druckwerk immer mit ausreichend Daten versorgt wird und der Druckprozess nicht pausieren muss, um auf neue Daten zu warten.

Bei den Bilddaten selbst handelt es sich um unkodierte Bilddaten mit 1-bit Farbtiefe, welche als sequentieller Datenstrom der Controllerplatine übergeben werden. Das Format, mit welchem der Controller die Daten überträgt, ist in Abbildung 5.2.2 näher erläutert.

Auswertung der Status-Bits und Fehlererkennung

Das Controllerboard IFV101-01B überträgt im Fehlerfall Status-Informationen im langsamsten unterstützten Modus 2400 bps an den Arduino. Da dazu allerdings die serielle Schnittstelle des Arduino on-the-fly umkonfiguriert werden müsste, sind diese Daten nur schwer auszulesen. Eine weitere Möglichkeit besteht darin, 4 diskret vorhandene Ausgangsleitungen auszuwerten, wobei jene Fehlerzustände damit kodiert werden, welche in Tabelle 4.1 aufgelistet sind. Einzig der Fehlerzustand „kein Papier“ kann nicht auftreten, da die Fotodiode zur Erkennung des Papierstatus im Druckwerk überbrückt wurde. Wäre die Fotodiode aktiv, so wäre es nicht möglich, den Schrittmotor für den Einzelblatteinzug zu nutzen, da das Druckwerk sofort in den Fehlermodus „kein Papier“ wechseln würde und anschließend sämtliche Feed-Befehle ignorieren würde.

Status	!PERR	ST0	ST1	ST2
Initialisierungsfehler	0	0	0	0
Hardwarefehler	0	1	0	0
Kein Papier*	0	1	1	1
Positionierungsfehler Abdeckung	0	0	1	1
Vp (Versorgungssp. Druckkopf) Fehler	0	1	0	1
Druckkopftemperatur Fehler	0	0	0	1
„Return-Waiting“ Status	1	1	0	1
„Print-Ready“-Status	1	0	0	1

Tabelle 4.1: Mögliche Fehlerzustände der IFV101-01B Steuerungsplatine

Diese 4 diskreten Ausgangsleitungen werden von der Arduino-Software laufend ausgelesen und ständig in Form des Ready-Status-Bytes an den PC übertragen, die Fehlerbehandlung wurde auf diesem Weg in die bereits vorhandene Flusskontrolle integriert. Dies ist notwendig, um Fehlerzustände im Druckertreiber dem/der Benutzer/-in ansprechend erklären und Ratschläge zur Behebung derselben geben zu können.

Für den Datentransfer vom emulierten USB-Serial-Interface (über welches der Arduino mit dem PC verbunden ist) zum IFV101-01B-E Interface Board wird die Serial-Klasse aus dem Arduino Software Stack verwendet. Um eine höhere Effizienz zu erreichen, werden immer Blöcke mit einer Länge von bis zu 32 Bytes kopiert. Dadurch kann der Overhead des Arduino Software Stacks durch Übertragung der Daten in Batches minimiert werden. Ein direkter Datentransfer vom Interface-Board zum PC ist nicht vorgesehen, hier sendet die Arduino-Software selbstständig Datenpakete, welche den Status des Druckwerks und der Steuerplatine beinhalten sowie zur Flusskontrolle dienen.



Hardware

In diesem Kapitel wird der Hardwareaufbau detailliert beschrieben. Zu Beginn wird von Abschnitt 5.1 bis Abschnitt 5.4 auf die verwendeten Hauptkomponenten wie auf das Thermodruckwerk, die Controllerplatine sowie das Arduino-Leonardo Mikrocontroller-Board näher eingegangen. Dabei wird auch erläutert, weshalb jene Komponenten für diese Arbeit ausgewählt wurden sowie ausreichend Detailwissen über die Komponenten vermittelt, um einen späteren Nachbau des Prototyps zu ermöglichen und die Hintergründe zum gewählten Schaltungsdesign verstehen zu können.

Anschließend wird in Abschnitt 5.5 auf die Thematik der Stromversorgung eingegangen. Weiters werden darin Alternativen zum gewählten Lösungsweg der Verwendung eines Pufferakkumulators zum Abdecken der Stromspitzen des Druckwerks aufgezeigt.

In Abschnitt 5.6 folgt die Beschreibung der Problemstellung, das Druckwerk so zu adaptieren, dass statt Endlospapier auf Rollen, Einzelblätter eingezogen werden können. Besonders herausfordernd an dieser Anforderung ist die Tatsache, dass beim Einzugsvorgang das Papier entgegen der üblichen Papierlaufrichtung transportiert werden muss, was die Konstruktion einer Umlenkvorrichtung notwendig machte. Auch wird die Aufhebung der Papier-Einspannung mittels Servomotor näher erklärt, welche notwendig ist, um das Einzelblatt zwischen die Führungsrollen des Druckwerks zu transportieren.

Den Abschluss bildet der Abschnitt 5.7, in welchem die Verkabelung der einzelnen Komponenten beschrieben ist. Weiters ist in diesem Abschnitt das schematische Schaltungsdesign der für die Stromversorgung notwendigen Leiterkarte abgebildet.

5.1 Thermodruckwerk

Trotz eingehender Recherche konnte kein Thermodruckwerk ausfindig gemacht werden, welches Unterstützung für Einzelblatteinzug bietet. Sämtliche erhältlichen Thermodruckwerke sind für den industriellen Einsatz mit Thermopapier-Rollen ausgelegt. Um Einzelblätter zu erhalten, ist bei diesen Druckwerken die Verwendung eines Cutters vorgesehen - das Endlospapier wird also erst nach dem Druckvorgang auf die passende Länge zugeschnitten. Ein großer Nachteil bei diesem Konzept ist allerdings die Notwendigkeit, bei einem Papierrollenwechsel aufwändig das neue Papier ins Druckwerk einfädeln zu müssen, da normalerweise der Papierwechsel von kundigem Fachpersonal oder einem/einer dedizierten Service-Techniker/-in durchgeführt wird. Ein komplizierter Wechsel von Verbrauchsmaterial steht allerdings in Konflikt mit der Anforderung, dass der Drucker wartungsfrei und einfach in der Handhabung sein muss.

Daher wurde bei der Auswahl des Thermodruckwerks darauf Wert gelegt, dass dieses für Umbaumaßnahmen auf Einzelblatteinzug gut geeignet ist, wobei auf folgende Eigenschaften geachtet wurde:

- Vorschub und Rückzug von Papier ist über ein externes Signal steuerbar. Diese Funktion ist notwendig, um das Einzelblatt zuerst vollständig einzuziehen, um es anschließend von oben nach unten bedrucken zu können.
- Unterstützung für möglichst dicke Thermopapier-Sorten. Da Thermopapier von Rollen üblicherweise relativ dünn ist, ist ein Großteil der erhältlichen Druckwerke nicht für die Verarbeitung von dickeren Papiersorten geeignet. Dies ist allerdings notwendig, damit ein Einzelblatt die für den Einzugvorgang notwendige Steifigkeit aufweisen kann.
- Ein gerader Papierpfad, damit sich das Thermopapier-Einzelblatt beim Transport durch das Druckwerk nicht verformt oder verbogen wird. Außerdem soll damit das Auftreten von Biegekräften mit dickeren Papiersorten vermieden werden.
- Das Druckwerk sollte von einer etablierten Herstellerfirma stammen oder im Idealfall von mehreren Herstellerfirmen in zueinander kompatibler Ausführung angeboten werden. Damit kann beispielsweise im Falle der Produktionseinstellung bei einer Herstellerfirma die Notwendigkeit eines umfangreichen Redesigns durch verbesserte Langzeitverfügbarkeit verhindert werden.

- Die Ansteuerung soll mittels eines einfachen Mikrocontroller-Boards erfolgen können, wobei für garantierte Kompatibilität Steuer-Prozessor und Druckwerk von der selben Herstellerfirma bezogen werden können.



Abbildung 5.1.1: Thermodruckwerk LTPV445

Quelle: <http://image.made-in-china.com/2f0j00YSlankiPfNce/Thermal-Printer-Mechanism-PT1041s-Seiko-LTPV-445-compatible-.jpg> aufgerufen
2016-07-06

Seiko Instruments LTPV445

Die Wahl fiel auf ein Thermodruckwerk der Firma „Seiko Instruments“ mit der Bezeichnung LTPV445. Es weist folgende technische Daten auf [8]:

- 112mm Papierbreite, wovon 104mm bedruckt werden können und auf beiden Seiten 4mm unbedruckt bleibt, siehe Abbildung 5.1.3
- Nur 90° gebogener Papierpfad, der Transport in beide Richtungen erfolgt mittels eines Schrittmotors
- Unterstützung für dickes Thermopapier, wie beispielsweise Nippon Paper Industries TC98KS-T1

- 832 Druckpunkte bei einer Auflösung von 8 Druckpunkten pro Millimeter, wie in Abbildung 5.1.2 dargestellt.
- Versorgungsspannung des Druckkopfes kann zwischen 4,2V und 8,5V variieren, wobei die maximale Stromaufnahme zwischen 4,5A und 10A beträgt und die Druckgeschwindigkeit zwischen 3 und 8,5cm/s variiert

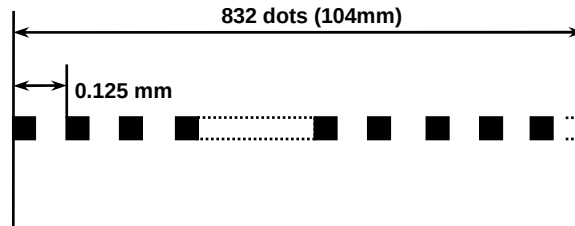


Abbildung 5.1.2: Pixel-Anordnung des Thermodruckwerks LTPV445 [8]

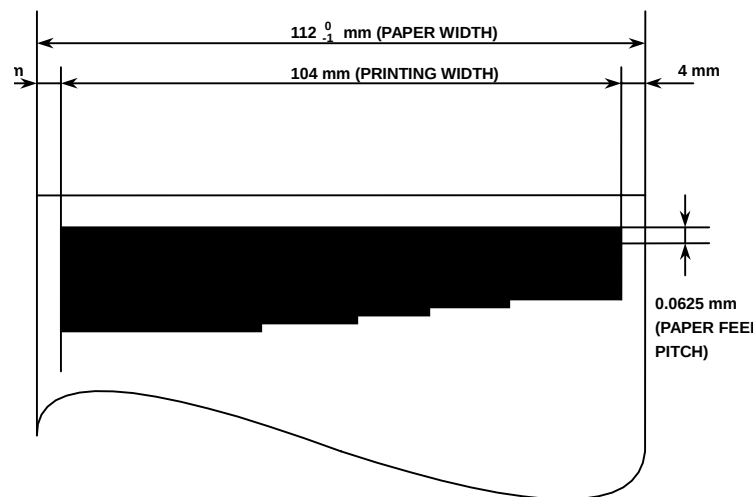


Abbildung 5.1.3: Druckbereich des Thermodruckwerks LTPV445 [8]

Ein zusätzlicher Faktor, welcher zur Auswahl des LTPV445 beigetragen hat, war die breite Verfügbarkeit von elektronisch und mechanisch kompatiblen Druckwerken anderer Herstellerfirmen. Außerdem bietet Seiko Instruments mit der Steuerungsplatine IFV101-01B-E die passende Elektronik an, um das Druckwerk von einem Mikrocontroller aus zu steuern.

Der Druckkopf selbst besteht aus 13 Blöcken mit jeweils 64 Punkten. Wie in Abbildung 5.1.4 ersichtlich, erfolgt die Datenübertragung zum Druckkopf seriell. Dabei werden die

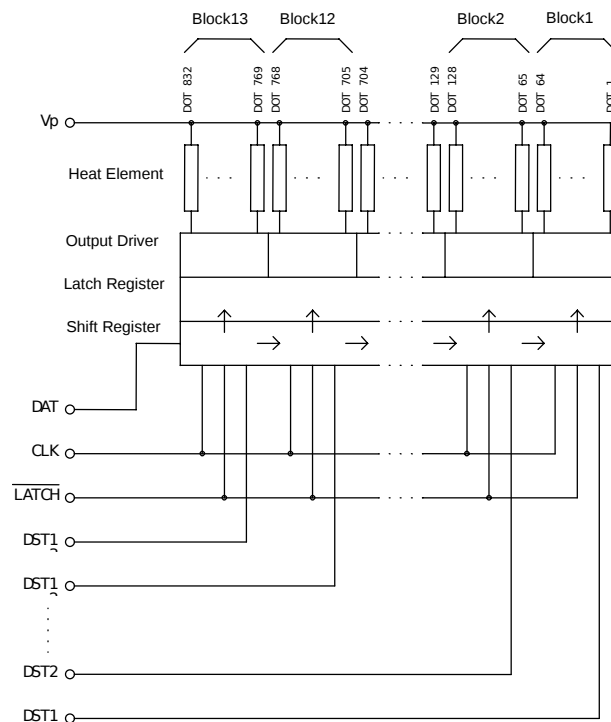


Abbildung 5.1.4: Datenübernahme und Ansteuerung des Thermodruckkopfes [8]

Daten zuerst in ein Schieberegister geladen, wobei mit jeder Flanke am CLK-Eingang die Bits des Schieberegisters um eine Position nach rechts verschoben werden und ein neuer Wert aufgenommen wird. Sobald eine Zeile vollständig im Schieberegister bereitsteht, können die Daten mittels einer negativen Flanke am !Latch-Eingang auf die Heizelemente des Druckkopfes durchgeschaltet werden. Um die für die Heizelemente notwendigen hohen Ströme zu erreichen, besitzt jedes Heizelement noch eine Treiberstufe. Der Widerstand eines Heizelements beträgt dabei zwischen 163 und 177 Ohm.

Die notwendige Anpassung der Transportgeschwindigkeit an die Spannung, die an den Heizelementen anliegt, um das Thermopapier gleichmäßig mit Wärme zu beaufschlagen, wird nicht vom Druckkopf selbst vorgenommen, sondern muss durch eine separate Controller-CPU vorgenommen werden. Der Papiervorschub selbst erfolgt mittels eines vierphasigen bipolaren Schrittmotors, wobei zwei Schritte einer Druckzeile entsprechen.

Das Thermodruckwerk verfügt über eine Vielzahl von Sensoren. So gibt es einen Sensor zur Erkennung, ob die Einzugmechanik geschlossen ist, welcher mit einem Schließer-Kontakt ausgeführt ist. Weiters ist ein Papiersensor vorhanden, der mittels Fotodiode feststellt,

ob sich Thermopapier im Einzug befindet, um die Gummiwalze der Einzugmechanik nicht durch einen Druckvorgang ohne Papier zu beschädigen, da dabei die aktivierten Heizelemente die Gummiwalze beschädigen würden. Aus Sicherheitsgründen besitzt das LTPV445 einen Temperatursensor zum Schutz vor Überhitzung. Der Richtpreis des Druckwerks bei Einzelbezug beläuft sich auf 65€.

Anmerkung zum Bezug von geeignetem Thermopapier

Für den Betrieb des Druckers hat sich die Verwendung von Thermopapier im Rollenformat als nicht praktikabel erwiesen. Thermopapier, welches in Rollenform erhältlich ist, ist üblicherweise sehr dünn (beispielsweise 40-50 g/m²) und besitzt dadurch zu wenig Steifigkeit, um entgegen der Schwerkraft in der Papierführung des Druckers aufrecht stehen zu bleiben. Zusätzlich sind diese Papiere, selbst nach wochenlanger Lagerung in ausgerollter Form, immer bestrebt, wieder eine gerollte Form anzunehmen.

Daher hat sich die Verwendung von Thermopapier-Einzelblättern bewährt, wie sie beispielsweise von der Firma Brother für die im Abschnitt 1.4 erwähnten Reisedrucker angeboten werden. Da jenes Papier auch für den Einsatz von geschäftlicher Korrespondenz beworben wird, besitzt es eine vergleichbare Stärke mit 80g/mm² Normalpapier. Auch wenn die Steifigkeit, verglichen mit diesem, etwas geringer ausfällt, hat sich das Papier für den im Rahmen dieser Diplomarbeit konstruierten Einzelblatteinzug gut bewährt. Als Nachteil dieses Papierformats muss die Notwendigkeit eines manuellen Zuschnitts erwähnt werden, da jenes Papier nur im Format A4 erhältlich ist.

5.2 Steuerungsprozessor Seiko Instruments PTV10P01-E

Zur Ansteuerung des LTPV445 Thermodruckwerks wurde die Steuerungsplatine IFV101-01B-E verwendet, welche mit einem PTV10P01-E Steuerungsprozessor ausgestattet ist, der ebenfalls von Seiko Instruments stammt. Der Steuerungsprozessor bildet die Schnittstelle zwischen dem System, welches die Druckaufträge generiert (und beispielsweise aus einem Mikrocontroller besteht oder mit einem PC aufgebaut ist) und dem Thermodruckkopf und ist dabei für folgende Aufgaben zuständig [9]:

- Datentransfer zum Thermodruckkopf. Die Daten werden mit 4,608 Mbps in das Schieberegister des Thermodruckkopfes übertragen, wobei der IFV101-01B-E auch den Takt für das CLK-Signal generiert.

- Intelligente Ansteuerung der 13 physischen Blöcke des Druckkopfes, um die Stromaufnahme zu begrenzen.
- Regelung der pro Fläche aufgetragenen Energiemenge. Diese ist abhängig von der Außentemperatur sowie der Versorgungsspannung des Druckkopfes. Bei niedrigerer Außentemperatur beziehungsweise Versorgungsspannung wird der Papiervorschub verlangsamt, um ein gleichmäßiges Druckbild zu erhalten.
- Ansteuerung des im Druckkopf für den Papiervorschub verbauten, vierphasigen bipolaren Schrittmotors inklusive Motorschutz durch Versorgung mit Konstantstrom.
- Umsetzung von „High-Level“ Zeichenbefehlen (Text, Liniengrafiken, Bar-Code) in Rastergrafik

Schnittstellen Die Verbindung zum PC oder Mikrocontroller-Board ist entweder über eine serielle oder parallele Schnittstelle möglich, wobei allerdings nur TTL-Pegel (0 V- 5V) zugelassen sind. Daher ist die direkte Verwendung an einem PC ohne Spannungswandler nicht möglich und kann zur Beschädigung des Steuerungsprozessors führen. Darüber hinaus gibt es die Möglichkeit, den PTV10P01-E mit einem zusätzlichen Schnittstellen-IC mit einer USB-Schnittstelle auszurüsten. Da jedoch die fertige Steuerungsplatine IFV101-01B-E nicht derartig ausgestattet ist, wurde im Verlauf der Diplomarbeit auf die serielle Schnittstelle zurückgegriffen.

Der PTV10P01-E besitzt eine Vielzahl von Schutzmechanismen, um Fehler im Betrieb zu erkennen und eine Beschädigung des Druckkopfes zu vermeiden. So wird während des Betriebs der Widerstand jedes einzelnen Heizelements sowie die Versorgungsspannung überwacht. Zusätzlich werden die Sensoren des Thermodruckkopfes ausgewertet. Tritt ein Fehler auf, so kann dieser über 3 Fehlerpins (ST0 - ST2) sowie einen Statuspin (!PERR) abgefragt werden - woraus sich 8 Fehlermöglichkeiten ergeben.

Die serielle Schnittstelle des PTV10P01-E unterstützt eine Vielzahl von verschiedenen Übertragungsstandards. So werden Baud-Raten zwischen 2400 und 115200 bps unterstützt, es kann zwischen 7 und 8 bit Wortlänge gewählt und die Art der Parität kann festgelegt werden. Zusätzlich beherrscht der PTV10P01-E Flusskontrolle in der Software mittels Xon/Xoff Steuerdaten sowie Flusskontrolle mit einem separaten „busy“-Pin.

Aufteilungsmodus Um die Stromaufnahme zu begrenzen, kann der PTV10P01-E den Druckkopf auf 3 unterschiedliche Arten anzusteuern [9]:

- Zweifache Aufteilung: Dabei sind maximal 50% der Druckpunkte aktiv (448), wobei die Stromaufnahme des Druckkopfs dabei Versorgungsspannung / 0.4 + 1A beträgt. Bei der Versorgungsspannung von 7,2V ergibt sich in diesem Betriebsmodus eine Stromaufnahme von bis zu 19A.
- Fünffache Aufteilung, bei welcher maximal 192 Punkte aktiv sind. Hier beträgt die Maximalstromaufnahme Versorgungsspannung / 0.97 + 1A. Daraus ergibt sich bei 7,2V Versorgungsspannung eine maximale Stromaufnahme von 8,42A.
- Dynamische Aufteilung: Bei der dynamischen Aufteilung zählt der PTV10P01-E die aktiven Pixel in jedem Block und passt anhand der Aussteuerung die Aufteilung an, wie in Abbildung 5.2.1 dargestellt. Vorteil der dynamischen Aufteilung ist, dass für Bereiche, auf welchen nur wenige Pixel schwarz gedruckt werden sollen, eine höhere Druckgeschwindigkeit erreicht werden kann. Dynamische Aufteilung erfordert allerdings einen sehr präzisen Papiervorschub, da, je nach zu druckenden Daten, theoretisch jede Zeile mit unterschiedlicher Geschwindigkeit gedruckt werden können muss. Für hochqualitative Ausdrücke, beziehungsweise wenn hohe Druckgeschwindigkeit nicht die höchste Priorität hat, wird daher eine fixe Aufteilung empfohlen.

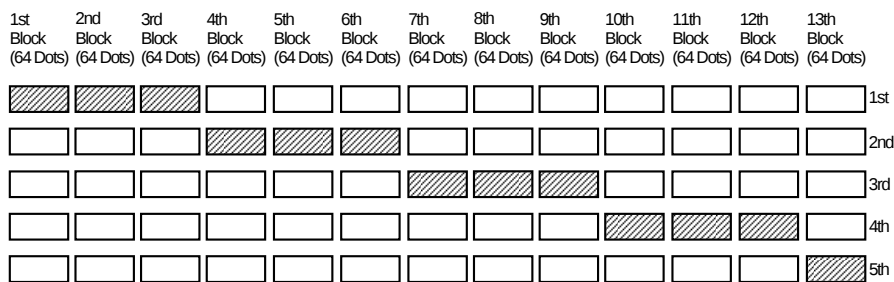


Abbildung 5.2.1: Dynamische Aufteilung für eine schwarze Linie [9]

Protokoll Der PTV10P01-E unterstützt eine breite Palette von Steuer- und Zeichenbefehlen. Diese umfassen beispielsweise Befehle zum Steuern des Papiervorschubs und Rückzugs, zum Hochladen von selbst definierten Zeichen sowie zum Konfigurieren verschiedener Druck-Modi wie beispielsweise gespiegelter Druck. Zusätzlich beherrscht

der PTV10P01-E eine Stempelfunktion, also einen weiteren Layer, welcher über der eigentlichen Inhaltsebene gedruckt wird.

Für die vorgesehene Verwendung mit einem Windows UniDrv-Druckertreiber ist der Grafikmodus von besonderer Bedeutung. Dabei wird ein Monochrombitmap fixer Breite (832 Bit) und variabler Höhe gedruckt - das Rastern der Zeichenbefehle übernimmt dabei der Druckertreiber am PC. Dies hat den großen Vorteil, dass das Druckergebnis exakt der Druckvorschau am Windows-PC entspricht, bedeutet im Gegenzug allerdings, dass sämtliche Rasterfähigkeiten des PTV10P01-E ungenutzt bleiben.

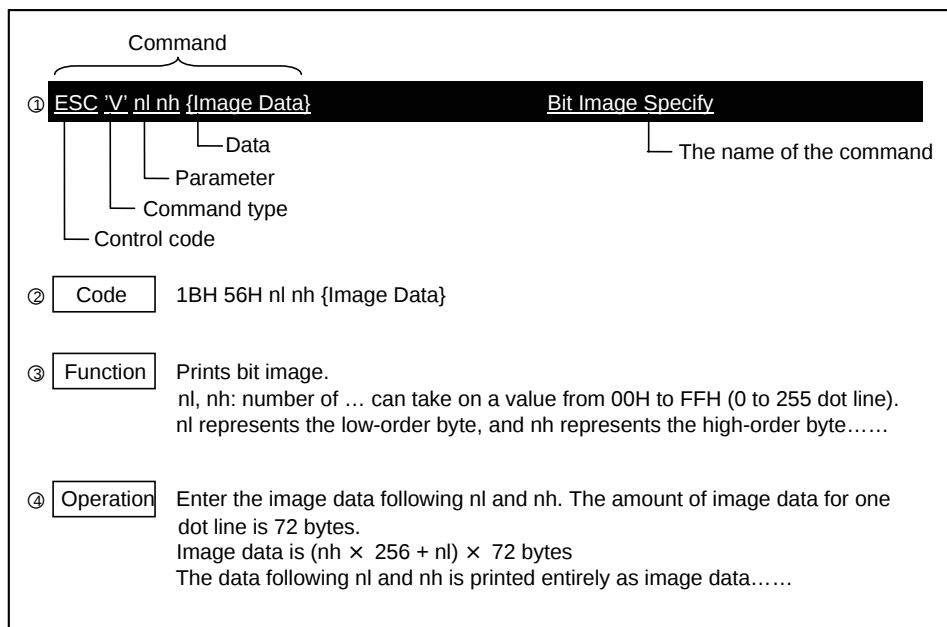


Abbildung 5.2.2: Befehlsstruktur der PTV10P01-E CPU [9]

Der Aufbau eines Kommandos des PTV10P01-E Druckprozessors ist in Abbildung 5.2.2 dargestellt. Jedes Kommando wird von einem Byte mit dem Wert 0x1B eingeleitet. Darauf folgt ein Byte, welches die Art der durchzuführenden Operation festlegt, also ein Opcode-Byte. Dem Opcode folgt eine variable Anzahl an Parametern, wie beispielsweise die Höhe eines zu druckenden Bildes sowie die Bilddaten.

5.3 Interface-Board Seiko Instruments IFV101-01B-E



Abbildung 5.3.1: Controllerplatine IFV101-01B-E

Quelle: http://eu.mouser.com/images/seikoinstruments/lrg/IFV101-01B-E_SPL.jpg aufgerufen 2016-07-06

Da es sich bei dem PTV10P01-E um einen *Integrated Circuit* (IC) mit *Surface Mounted Device* (SMD) -Gehäuse handelt und einiges an externer Beschaltung notwendig ist, fiel für den, im Rahmen der Diplomarbeit hergestellten, Prototypen die Wahl auf ein fertiges Interface Board. Dieses basiert auf dem PTV10P01-E-Prozessor und beinhaltet sämtliche externe Bauteile, die für den Betrieb des PTV10P01-E notwendig sind.

Das Block-Diagramm in Abbildung 5.3.2 stellt die, auf dem Interface-Board verbauten Komponenten, in vereinfachter Form dar. Erwähnenswert ist dabei der separat verbaute Motor-Treiber-IC, da der PTV10P01-E mit den Digitalausgängen nicht den notwendigen Strom aufbringen kann, um den Schrittmotor direkt zu treiben. Zusätzlich ermöglichen verbaute *Dual Inline Package* (DIP)-Schalter die Auswahl unterschiedlicher Betriebsparameter, wie beispielsweise die Betriebsmodi der seriellen Schnittstelle oder ob die Kommunikation mit dem Mikrocontroller über die serielle oder die parallele Schnittstelle erfolgen soll. Der Richtpreis der IFV101-01B-E Steuerungsplatine beläuft sich bei Einzelbezug auf 107€.

5.4 Arduino Leonardo

Beim Arduino Leonardo handelt es sich um eine Mikrocontroller-Platine auf Basis des Atmega32u4 Mikrocontrollers [1]. Der ATmega32 ist ein 8-bit Mikroprozessor aus der Atmel AVR Serie, welcher mit 16 Mhz betrieben wird und auf der Harvard-Architektur mit getrenntem Bus für Instruktionen und Daten basiert [2]. Direkt in den ATmega32u4

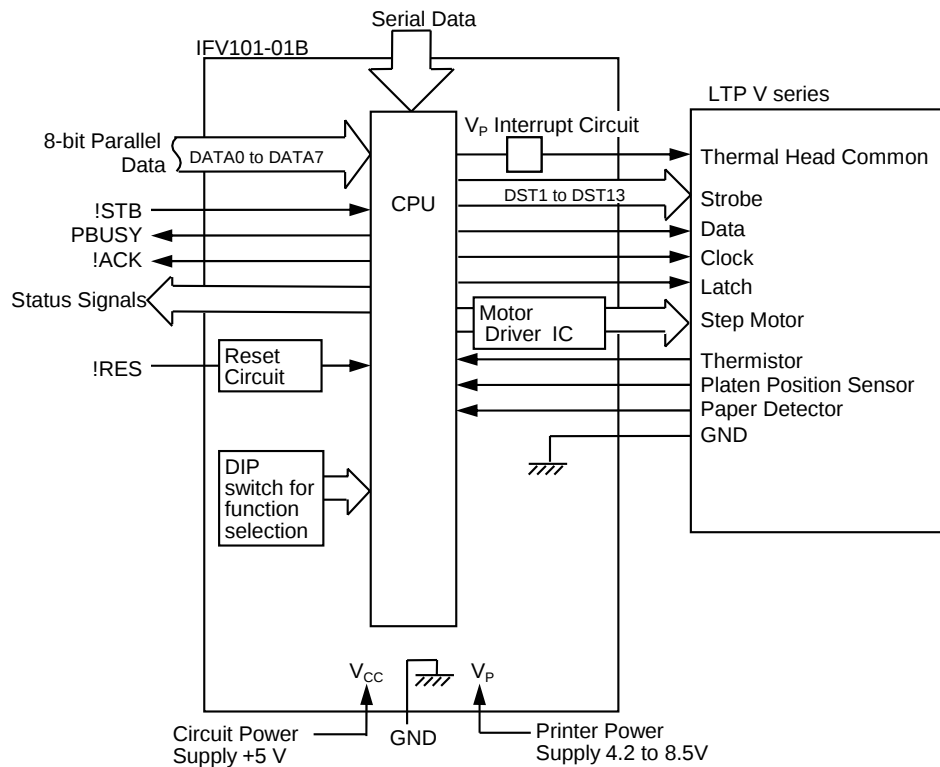


Abbildung 5.3.2: Blockdiagramm des IFV101-01B-E Interface Boards [7]

sind 32kB Flash Speicher, 2,5 kB *Static Random Access Memory* (SRAM) sowie 1kB *Electrically Erasable Programmable Read-Only Memory* (EEPROM) integriert. Somit sind sowohl Programm- als auch Arbeitsspeicher direkt im ATmega32-Chip integriert und es bedarf keiner zusätzlichen externen Bauelemente.

Stärken des ATmega32 sind die vielfältigen Input/Output Schnittstellen, von denen am Arduino Leonardo allerdings nicht alle herausgeführt werden, und seine geringe Stromaufnahme. Am Leonardo sind folgende Schnittstellen verfügbar:

- 1 Serial->USB Emulation (für die Datenübertragung zwischen Arduino-Leonardo und dem PC, über diese Schnittstelle wird der Arduino auch programmiert)
- 1 *Universal Asynchronous Receiver Transmitter* (UART), über welche das Interface-Board angebunden wird
- 20 digitale I/O-Pins

- 12 Analoge I/O-Pins
- 7 PWM-Kanäle

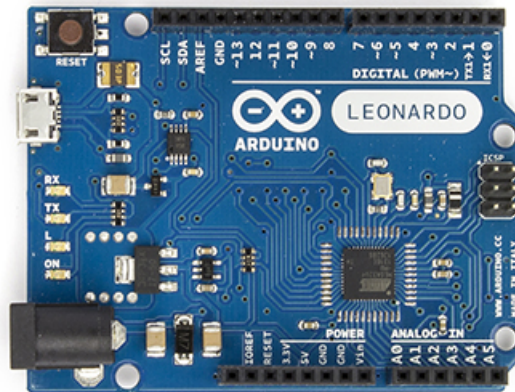


Abbildung 5.4.1: Leonardo Arduino-Platine in der Draufsicht

Quelle: <http://cdn.instructables.com/F00/9322/I7SVSZB5/F009322I7SVSZB5.MEDIUM.jpg> aufgerufen 2016-07-06

Erwähnenswert ist, dass die Serial->USB Emulation bereits direkt vom ATmega32u4 unterstützt wird. Im Vergleich zu einigen Arduino-Platinen mit anderen AVR-Typen benötigt der Leonardo keinen separaten *Future Technology Devices International* (FTDI)-Baustein für die Umsetzung. Für einen Mikrocontroller sehr positiv ist das große Register-File, welches 32 8-bit Register enthält (im Vergleich dazu weißt die Intel 8051-Architektur nur 1/4 der Register auf). Die große Anzahl an Registern, sowie die Tatsache, dass es sich dabei um „General Purpose“-Register handelt, welche flexibel für unterschiedliche Aufgaben verwendet werden können, wirkt sich günstig auf die Qualität von Compiler-generiertem Code aus. Zusätzlich sind auf dem Arduino Leonardo noch ein Quarzoszillator mit 16 Mhz verbaut, der den ATmega32u4 mit Takt versorgt. Außerdem finden sich auf der Platine noch Bauelemente zur Stabilisierung der Spannungsversorgung, die die Zuverlässigkeit im Betrieb bei Spannungsschwankungen erhöhen und Beschädigung bei zu hoher Versorgungsspannung vermeiden.

5.4.1 Software Entwicklung für Arduino

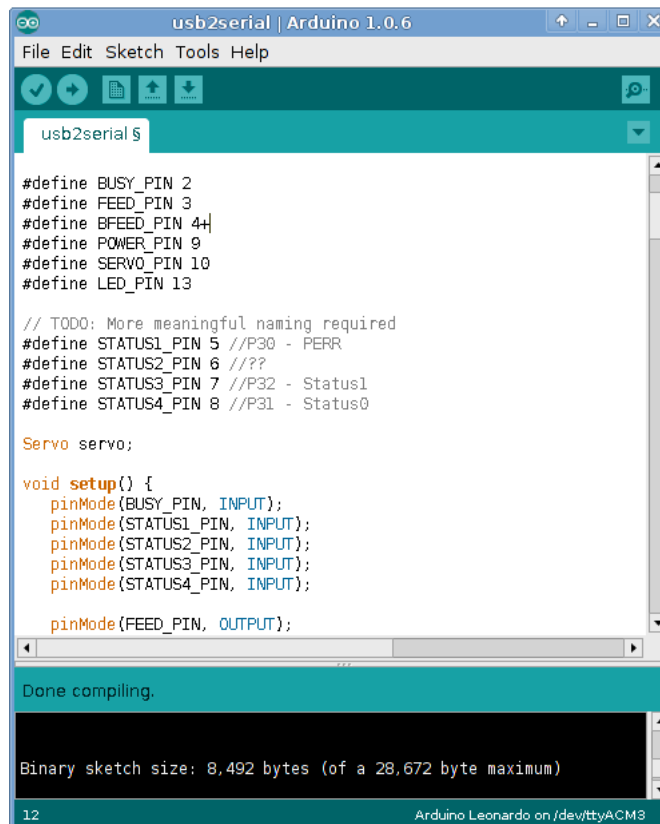


Abbildung 5.4.2: Arduino-IDE mit geladenem „Sketch“

Für die Entwicklung von Arduino-Software gibt es eine freie *integrierte Entwicklungsumgebung* (IDE) auf Java-Basis, wobei Arduino-Programme aufgrund ihrer üblicherweise geringen Größe als „Sketches“ bezeichnet werden. Für das Erzeugen vom Maschinencode, welcher anschließend auf dem ATmega32u4-Mikrocontroller ausgeführt wird, benutzt die Arduino-IDE den AVR-GCC Compiler, daher werden selbst relativ neue C/C++-Sprachstandards unterstützt und durch aufwändige Optimierungen eine hohe Qualität des generierten Codes ermöglicht.

Integraler Bestandteil der Arduino-IDE ist eine „Wiring“ genannte Softwarebibliothek, welche eine einfach zu nutzende Abstraktionsschicht für Input/Output-Aufgaben wie digitale und analoge Input/Output-Pins oder beispielsweise die serielle Schnittstelle darstellt. Listing 5.1 veranschaulicht die Verwendung der Wiring-Bibliothek anhand eines kurzen Beispiels, welches eine LED an einem digitalen I/O-Pin des Arduino blinken lässt. Dabei

gibt es die Funktion `setup()`, welche beim Start des Arduino-Sketches einmal ausgeführt wird und für aufwändigere Initialisierungen verwendet werden sollte. Anschließend wird die Funktion `loop()` in einer Endlosschleife ausgeführt. Der große Vorteil der Wiring-Bibliothek besteht in der Möglichkeit, selbst komplexere Ein-/Ausgabe-Operationen ohne zeitaufwändige Einarbeitung in das CPU-Datenblatt und dem manuellen Adressieren von Registern zu ermöglichen. Weiters stellt die Wiring-Bibliothek ein portables Interface dar, das für die verschiedensten Arduino-Plattformen zur Verfügung gestellt wird. So ist selbst für Arduino-Plattformen, welche nicht auf AVR, sondern stattdessen auf ARM oder Intel-Architekturen setzen, das Interface der Wiring-Bibliothek verfügbar.

Algorithmus 5.1 Codebeispiel zur Verwendung der Wiring-Bibliothek.

```
1 #define LED_PIN 13
2
3 void setup () {
4   pinMode (LED_PIN, OUTPUT); // Enable pin 13 for digital output
5 }
6
7 void loop () {
8   digitalWrite (LED_PIN, HIGH); // Turn on the LED
9   delay (1000); // Wait one second (1000 milliseconds)
10  digitalWrite (LED_PIN, LOW); // Turn off the LED
11  delay (1000); // Wait one second
12 }
```

5.5 Stromversorgung

Laut Anforderungsbeschreibung soll der Belegdrucker nur ein einziges Kabel benötigen, nämlich die Verbindung zwischen Drucker und PC. Diese soll mittels des USB-Bus ausgeführt werden. Ein separates Netzteil sollte, aus Gründen der Benutzerfreundlichkeit, falls möglich, vermieden werden, da davon ausgegangen wurde, dass die Stromversorgung des Druckers über die USB Schnittstelle realisierbar sein sollte.

Wie in Abschnitt 5.2 erläutert, benötigt das Thermodruckwerk selbst bei 5-fach Aufteilung der Druckpunkte bis zu 8,45A bei 7,2V, was einer Leistungsaufnahme von etwa 60W entspricht. Der USB-Bus stellt allerdings nur eine Spannung von 5V mit einer Stromstärke von bis zu 500mA zur Verfügung. Dies entspricht nur einem 1/24 der Leistung, die das Druckwerk bei 5V Versorgungsspannung als Spitzenlast aufnehmen würde. Außerdem sind bei dieser Berechnung die Steuerplatine IFV101-01B-E sowie das Arduino-Mikrocontroller-Board noch nicht berücksichtigt.

Als Lösungsweg für dieses Problem wurde entschieden, zur Abdeckung der Spitzenströme

während des Druckvorgangs Akkumulatoren einzusetzen. Dieser Lösungsweg geht davon aus, dass der Drucker um ein Vielfaches länger vom PC mit Strom versorgt wird, als er aktiv einen Druckvorgang ausführt. Bei einer Leistungsaufnahme von 60W während des Druckvorgangs, einer Leistungsbereitstellung von 2,5W seitens des USB-Busses und einem angenommenen Wirkungsgrad von etwa 70% der Akkumulatoren, ist es möglich, dass der Drucker bis zu 3% der Zeit, in welcher er mit dem PC verbunden ist, aktiv druckt.

Aus Sicherheits- und Umweltschutzgründen haben wir uns für die Verwendung von *Nickel-Metall-Hydrid* Akkumulatoren (NIMH) entschieden. Diese sind wie *Nickel-Cadmium* (NiCd) Akkumulatoren sicherheitstechnisch bei Ladeströmen um die 10% der Zellenstundenleistung unbedenklich. Im Gegensatz dazu würde die Verwendung von *Lithium-Ionen* (LION) oder *Lithium-Polymer* (LiPo) Zellen aufwändige Sicherheitsschaltungen erfordern, um das Risiko der Brandgefahr während des Ladevorgangs zu minimieren. So häuften sich beispielsweise in den letzten Jahren Berichte in den Medien über Mobilgeräte, die sich spontan entzündeten, wobei in den meisten Fällen defekte Lithium-basierte Akkumulatoren der Auslöser solcher Brände waren, was auch dazu geführt hat, dass Geräte mit Lithium-Zellen von Paketdiensten oft unter speziellen Sicherheitsvorkehrungen transportiert werden. Von Geräten mit NiCd beziehungsweise NIMH-Zellen sind dem Autor keine derartigen Fälle bekannt. Die geringere Energiedichte, welche NiCd/NiMH-Zellen, im Vergleich zu LION/LIPO-Zellen, aufweisen, spielt aufgrund des stationären Einsatzes des Belegdruckers nur eine untergeordnete Rolle. Des Weiteren sind NIMH-Zellen im Gegensatz zu NiCd-Zellen frei von Schwermetallen und somit in der Entsorgung vergleichbar mit Alkali-Batterien. Die Verwendung des Schwermetalls Cadmium ist der Auslöser, weshalb der Verkauf von NiCd-Akkumulatoren in der EU bis auf einige Nischenanwendungen untersagt wurde.

Als Schwierigkeit erweist sich die Tatsache, dass der USB-Bus mit 5V nur wenig mehr Spannung bereitstellt, als die 4,2V welche mindestens für den Betrieb des Druckwerks notwendig sind. Um das Druckwerk mindestens mit 4,2V versorgen zu können, wäre eine Serienschaltung aus 4 Nickel-Metall-Hydrid (NIMH) Zellen notwendig, welche 1,2V Zellenspannung aufweisen und woraus sich die Spannung der Serienschaltung von 4,8V ergibt. Da allerdings die Ladeschlussspannung von NIMH-Zellen bei einem Ladestrom von etwa 1/10 der Stundenleistung bei etwa 1,45 V liegt, wäre eine Spannung von mindestens 5,8V notwendig, um 4 Zellen in Serienschaltung vollständig aufladen zu können.

Um dennoch eine vollständige Ladung der Akkumulator-Zellen zu erreichen, wurde ein DC/DC-Konverter verwendet. Dieser wandelt die 5V Gleichspannung, welche vom USB-

Bus bereitgestellt wird, in 9V Gleichspannung um. Daraus ergibt sich die Möglichkeit, Akkupacks zu laden, welche aus 6 NIMH-Zellen aufgebaut sind und eine Nennspannung von 7,2V aufweisen, da die zur vollständigen Ladung notwendige Ladeschlussspannung von 8,7V unter der erreichten Spannung des DC/DC-Konverters liegt. Neben einer höheren Druckgeschwindigkeit, erlaubt die höhere Nennspannung auch eine längere Verwendung der Akkuzellen. Sollten diese im Laufe der Lebensdauer des Druckers aus Verschleißgründen einen höheren Innenwiderstand entwickeln, so bieten die 7,2V eine sehr große Spannungsreserve von 3V um welche die Nennspannung bei Belastung einbrechen darf, ohne dass das Druckwerk aufgrund zu geringer Spannung in einen Fehlerzustand wechselt.

Verwendeter Akkutyp Wir haben uns für einen 6 Zellen NIMH Akkupack der Firma Conrad entschieden, welcher eine Nennspannung von 7,2V sowie eine Kapazität von 2000mA/h aufweist. Durch die eigentlich vorgesehene Verwendung im Modellbau, erhoffen wir uns einen sehr geringen Innenwiderstand und somit nur einen geringen Spannungsabfall innerhalb der Zellen selbst bei den hohen Strömen, welche das Thermodruckwerk im Betrieb aufnimmt. Ein weiteres Kriterium für die Wahl eines Modellbau-Akkupacks ist die Vermutung, dass die darin verbauten Zellen, aufgrund der im Modellbau üblicherweise vorherrschenden rauen Bedingungen (Schnellladung, hohe Entladeströme), robuster konstruiert sind als Zellen die für die Verwendung in Consumer-Produkten gedacht sind und daher für den geplanten Einsatzzweck eine hohe Lebensdauer erreicht werden kann. Ob diese Annahme korrekt war, wird sich allerdings erst beim Einsatz des Belegdrucker-Prototyps unter realen Bedingungen zeigen.

Ladelogik Aufgrund der einfachen Realisierung wurde eine Ladeschaltung mittels Konstantstromverfahren gewählt. Für aufwändigere Ladeverfahren, welche eine schnelle Ladung der Zellen erlauben würden, würde außerdem nicht genügend Leistung vom USB-Bus zur Verfügung gestellt werden.

Bezüglich der Abschaltkriterien gibt es für NIMH-Zellen verschiedene Ansätze. Da die Ladeschlussspannung bei NIMH-Zellen stark von der Zelltemperatur abhängig ist, ist sie als alleiniger Maßstab zum Feststellen des Ladezustandes ungenügend. Daher werden üblicherweise folgende Techniken zum Erkennen des Ladeschlusses verwendet:

- Ladeschlusserkennung nach dem delta-U Verfahren. Da sich die Zelle am Ende des Ladevorganges erwärmt, sinkt der differentielle Widerstand, was sich in einem Spannungsabfall von 10-20mV bemerkbar macht. Dieser Spannungsabfall kann für

die Erkennung des Ladezustandes herangezogen werden. Dieser Spannungsabfall ist bei NiMH-Akkumulatoren im Gegensatz zu NiCD-Zellen nur wenig ausgeprägt und tritt erst bei höheren Ladeströmen von 0,1-0,2C auf, wobei C dabei für die Stundenleistung des Akkumulators steht. Für einen Akkumulator mit einer Stundenleistung von 2300mA entspricht ein Ladestrom von 0,1C daher 230mA.

- Bei der Ladeschlusserkennung nach dem Temperaturverfahren wird die Temperatur der Zellen überwacht. Überschreitet die Zelltemperatur einen gewissen Schwellenwert, so wird die Ladung beendet.

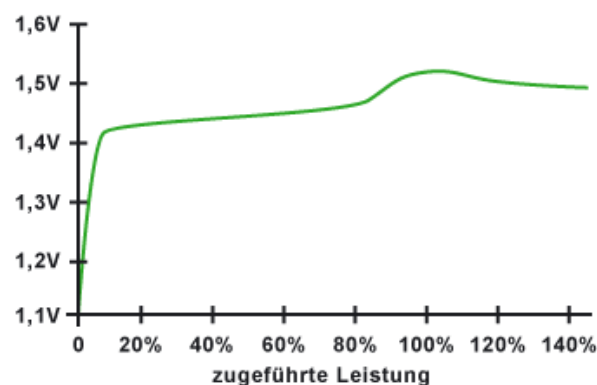


Abbildung 5.5.1: Ladekennlinie von NiMH-Akkumulatoren

Quelle: <http://www.elektronik-kompodium.de/sites/bau/1101251.htm> aufgerufen 2016-07-06

Beide Verfahren sind allerdings für den geplanten Einsatzzweck aus folgenden Gründen ungeeignet:

- Beide Verfahren beschreiben das Verhalten von vollständigen Lade- und Entladezyklen. Der Einsatz im Belegdrucker setzt sich allerdings aus längeren Ladephasen sowie kurzen Entladephasen zusammen. So könnte beispielsweise eine Zellenerwärmung, verursacht durch die hohe Leistungsaufnahme während des Druckvorgangs, zu einer Unterbrechung des Ladevorgangs bei Ladeschlusserkennung mit Temperaturverfahren führen.
- Beide Verfahren setzen höhere Ladeströme voraus, als durch den USB-Bus bereitgestellt werden kann. Bei einem Ladestrom von maximal 0,05-0,1C kann nicht davon ausgegangen werden, dass bei den verwendeten NiMH-Zellen ein ausgeprägter Spannungsabfall oder eine starke Erwärmung der Zellen beobachtet werden kann.

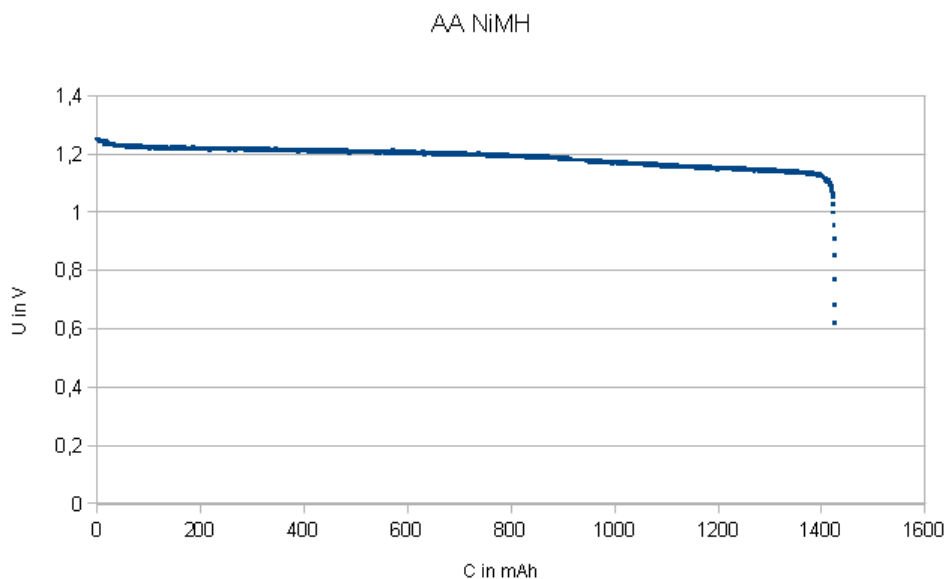


Abbildung 5.5.2: Spannungsverlauf bei Entladung einer NIMH-Zelle mit 1,4Ah
Quelle: https://upload.wikimedia.org/wikipedia/commons/4/4b/Aa-nimh_c-v.png aufgerufen 2016-07-06

Wir haben uns daher für eine abgewandelte Form der Dauerladung mit geringem Konstantstrom ohne Ladeschlusserkennung entschieden. Diese Form des Ladezyklus findet breite Anwendung bei NiCD-Zellen, welche ständig mit bis zu 0,1C geladen werden können, ohne Schaden zu nehmen. Die zugeführte, überschüssige Energie wird dabei in Wärme umgewandelt und an die Umgebung abgegeben. NIMH-Zellen sind diesbezüglich empfindlicher, so gibt beispielsweise der Zellenhersteller GP an, dass seine Zellen eine Dauerladung von 0,1C über einen Zeitraum von einem Jahr überstehen, ohne physisch Schaden zu nehmen¹. Aussagen hinsichtlich der Entwicklung von Innenwiderstand und Kapazität von derart geladenen Zellen gibt es nicht. Daher haben wir uns entschieden, eine abgewandelte Form der Konstantstromladung zu verwenden, welche versucht, die fortschreitende Überladung voller Zellen mit folgenden Mitteln zu vermeiden:

- Wird der Drucker mit dem PC verbunden, so werden die Akkumulatoren für die ersten 5 Stunden aufgeladen. Dies stellt sicher, dass die hohe Selbstentladung der NIMH-Zellen kompensiert wird und selbst nach Tiefentladung mindestens 50% der Zellenkapazität zur Verfügung stehen. Eine Ladestandserkennung mittels

¹<https://cellpacksolutions.co.uk/wp-content/uploads/2015/06/gp-batteries-gp211afh-technical-data-sheet.pdf> (letzter Aufruf: 2016-07-18)

Spannungsmessung ist aufgrund der sehr flachen Entladekurve von NIMH-Zellen, wie in Abbildung 5.5.2, nicht möglich.

- Anschließend wird für jeden ausgeführten Druckauftrag eine 30 Minuten dauernde Ladephase initiiert, wobei die Ladezeit bei mehreren Druckaufträgen akkumuliert wird.

Die Ladelogik wird dabei am Arduino Mikrocontroller-Board implementiert, welches zusätzlich die Spannung des Akkupacks über einen Analogeingang misst, um Tiefentladung oder Fehlfunktion durch zu geringe Spannung zu vermeiden. Dabei wird über die zum PC übertragenen Statusdaten auch ständig die Akkuspannung übertragen und vom Data Transfer Process ausgewertet.

Stromversorgung über USB 3.1

Als Alternative zu der Verwendung eines Pufferakkumulators bietet sich die Stromversorgung über den USB 3.1 Standard an [21]. Im Vergleich zum USB 2.0 Standard, der, wie im vorangegangenen Kapitel erwähnt, auf 2,5W limitiert ist, ermöglicht USB 3.1 eine Stromstärke von bis zu 2A bei 5V (10 W), was allerdings selbst wenn das Thermodruckwerk im Stromspardruckmodus betrieben werden würde, nicht ausreichend wäre.

Zusätzlich wurden bei USB 3.1 sogenannte „USB Power Delivery Profiles“ [20] eingeführt, wobei im leistungsstärksten Profile bis zu 100 W übertragen werden können, wie in Abbildung 5.5.4 dargestellt.

Da allerdings nicht jeder USB 3.1-fähige Host (wie beispielsweise Notebooks im Batteriebetrieb) in der Lage ist, 100 W bereitzustellen, verfügt USB 3.1 über ein Hand-Shake-Verfahren. Dabei werden anfangs nur 2A bei 5V bereitgestellt und erst anschließend ein Power-Delivery-Profile ermittelt, welches von Host und Device unterstützt wird. Zusätzlich müssen bei der Verwendung der neuen Power Delivery Profiles spezielle Kabel Verwendung finden. Weiters müssen sowohl Host als auch das Device USB 3.1 unterstützen, damit das Handshake-Verfahren durchgeführt werden kann - wobei die USB-Implementierung des Arduino nur USB 2.0 unterstützt.

Da die Verbreitung von USB 3.1 noch sehr gering ist und zum Zeitpunkt der Planung und Konstruktion des Prototyps nur ein Bruchteil der neu ausgelieferten PCs USB 3.1 unterstützt, haben wir uns entschieden vorerst auf die Verwendung dieser Schnittstelle und

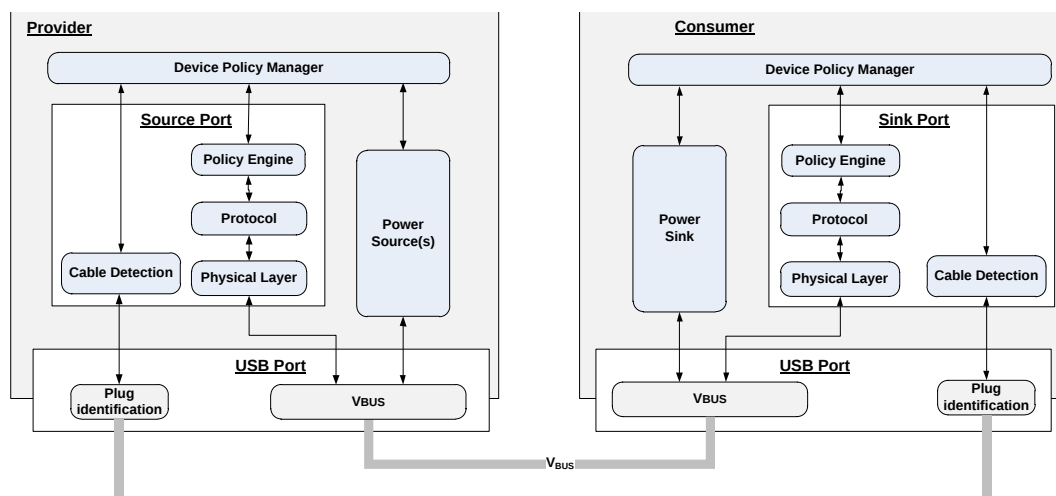


Abbildung 5.5.3: USB 3.1 Power Delivery Architecture [20]

deren erweiterten Stromversorgungsmodi zu verzichten. Außerdem wäre die Verwendung von USB 3.1 nur über einen USB-Hub möglich gewesen, welcher die Power Delivery Profiles unterstützt, da die USB-Schnittstelle des Arduino Leonardo auf USB 2.0 limitiert ist. Wir haben uns dennoch entschieden auf die Möglichkeit der Stromversorgung über USB 3.1 kurz hinzuweisen, um zukünftige Möglichkeiten aufzuzeigen, auf den im vorgestellten Design notwendigen Verschleißteil NIMH-Akku zu verzichten, sobald USB 3.1 eine breitere Unterstützung erfährt.

5.6 Ausführung des Papiereinzugs

Alle am Markt erhältlichen Thermodrucker, welche für 110 mm Papierbreite dimensioniert sind, sind auf Papiereinzug von einer Thermopapier-Rolle ausgelegt. Diese Eigenschaft rührt vom industriellen Einsatz, zum Beispiel in Ticketdruckern oder Fahrscin-automaten, her, in welchen diese Druckwerke typischerweise verbaut werden. Da ein Einzelblatteinzug hier zu fehleranfällig wäre, werden die Druckwerke von einer Rolle mit Thermopapier versorgt, auf das richtige Format wird das Papier mittels einer automatischen Schneidevorrichtung, welche als „Cutter“ bezeichnet wird, zugeschnitten. Ist der Papiervorrat auf der Rolle erschöpft, so erkennt das Druckwerk dies mittels einer Fotodiode und ein/eine Servicemitarbeiter/-in muss den Rollentausch vornehmen. Da der Rollentausch dem qualifizierten Fachpersonal vorbehalten ist, wird üblicherweise beim Design dieser Druckwerke wenig Fokus auf die Ergonomie gelegt. So ist auch der

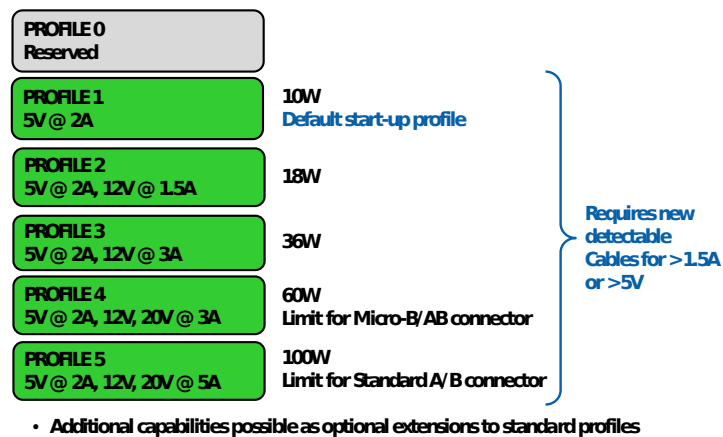


Abbildung 5.5.4: USB 3.1 Power Delivery Profiles [20]

Rollentausch beim verwendeten Druckwerk LTPV445 nur mit etwas Erfahrung und Fingerspitzengefühl erfolgreich durchführbar, für ältere Menschen mit leicht eingeschränkten motorischen und visuellen Fähigkeiten ist der Vorgang allerdings nicht zumutbar und würde das gesamte Konzept des Belegdruckers ad absurdum führen. Daher haben wir uns entschlossen, die Konstruktion des Druckwerks so zu verändern, dass damit ein Einzelblatteinzug möglich wird und der komplexe Vorgang des Rollentausches komplett entfällt.

Das LTPV445 Druckwerk ist als typisches Thermodruckwerk für den industriellen Einsatz auch, wie im vorangegangenen Absatz ausgeführt, für den Papiereinzug von einer Thermopapierrolle konzipiert. Für einen Einzelblatteinzug ergeben sich dadurch konstruktiv folgende Schwierigkeiten:

- Das Thermopapier wird zwischen einer Gummiwalze und dem eigentlichen Druckkopf transportiert, wobei die Gummiwalze über einen im Druckwerk verbauten Schrittmotor angetrieben wird. Dabei wird auf das Papier mittels Federkraft ein hoher Druck ausgeübt, einerseits um bei größeren Thermopapier-Rollen das Anzugmoment von der Walze auf das Papier zuverlässig übertragen zu können, um zu verhindern, dass das Papier über die Walze rutscht. Andererseits ist der hohe Anpressdruck notwendig, um einen guten thermischen Kontakt zwischen Papier und den Heizelementen des Druckkopfs zu gewährleisten. Der Fall, dass sich zwischen

Walze und Druckkopf kein Papier befindet, kann eigentlich nur durch eine leere Papierrolle verursacht werden und muss daher von einem/einer Servicetechniker/-in durch Rollentausch behoben werden. Diese Konstruktion hat dadurch allerdings den Nachteil, dass ohne Papier zwischen Walze und Druckkopf die Gummiwalze mit hoher Kraft auf den Druckkopf gepresst und dadurch der Antrieb der Walze blockiert wird. Es ist also mit der unmodifizierten Ausführung des Druckwerks nicht möglich, durch ein Rotieren-lassen der Walze ohne Papier den Transport eines lose an der Gummiwalze anliegenden Einzelblattes zu erreichen. Selbst wenn sich die Gummiwalze drehen lassen würde, wäre durch den hohen Anpressdruck kein Einzug durch die Engstelle des Kontaktpunktes zwischen Walze und Druckkopf gewährleistet.

- Alle Transportwege sind darauf ausgelegt, dass Endlospapier eingezogen wird. Es wurde bei der Konzeption des Druckwerks also nicht davon ausgegangen, dass eine Papierkante eingezogen werden könnte, welche gesondert geführt werden muss, da es kein Papier „davor“ gibt, welches über Zugkräfte die Richtung des nachfolgenden Papiers vorgeben könnte. Erschwerend kommt der Umstand hinzu, dass die Papierwege für die eigentliche Transportrichtung, welche der Druckrichtung entspricht, optimiert wurden, da üblicherweise ein Backfeed-Vorgang (dabei wird das Papier entgegen der eigentlichen Druckrichtung transportiert) gegebenenfalls nur für Korrekturen und Mehrfachbedruckung des selben Abschnittes benutzt wird. Da für den Belegdrucker allerdings nur ein Zugangspunkt sowohl für die Zuführung als auch die Entnahme des Einzelblattes vorgesehen ist, muss der Einzug entgegen der Druckrichtung im Backfeed-Modus erfolgen. Dabei muss eine Papierkante entgegen der vorgesehenen Transportrichtung bewegt werden, welche sich, mangels dafür ausgelegter Führungselemente, sofort im Druckwerk verkeilen würde.

Die Lösung dieser Problemstellungen wird in den nachfolgenden Unterkapiteln ausführlich beschrieben.

Aufhebung des Walzen-Anpressdrucks mittels Servomotors

Um eine Drehbewegung der Walze auch ohne Papier zwischen Walze und Druckkopf zu ermöglichen, wird während des Papier-Einzugvorganges mittels eines Servomotors eine Kraft ausgeübt, die der Federkraft des Druckkopfs entgegenwirkt und somit den Anpressdruck der Walze auf den Druckkopf aufhebt. Die Wahl fiel dabei auf einen Servomotor, da mittels der hohen Übersetzung des Servo-Getriebes die notwendige Kraft

aufgebracht werden kann. Wie in Abbildung 5.6.1 veranschaulicht, kann der Servo direkt vom Arduino, mittels eines *Pulsbreiten-modulierten* (PWM) Steuersignals, auf eine fixe Winkelposition eingestellt werden, wobei das Verhältnis zwischen Low- und High-Pegel über die Stellung des Servos entscheidet. Diese Stellung hält der Servo unabhängig von der auf den Servo ausgeübten Kraft konstant.

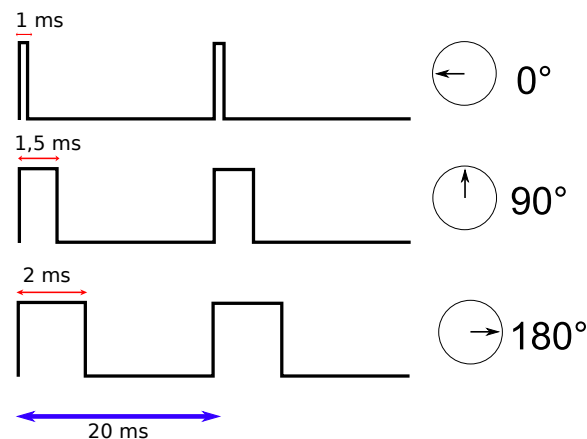


Abbildung 5.6.1: Ansteuerung des Servos über ein PWM-Signal

Quelle: <https://upload.wikimedia.org/wikipedia/commons/f/f6/TiemposServo.svg> aufgerufen 2016-07-06

Erreicht wird diese Eigenschaft mittels eines im Servo enthaltenen Potentiometers, über welchen der Servo seine Stellung ermitteln und gegebenenfalls korrigieren kann. Das für die Ansteuerung notwendige PWM-Signal wird dabei durch die im Arduino-Software-Stack enthaltene Servo-Klasse intern über Timer generiert, ohne zusätzliche CPU-Last am Arduino zu generieren. Daher musste auch kein PWM-Code implementiert werden und es war ausreichend, die Servo-Klasse für das dem Sensor entsprechende PWM-Verhältnis zu konfigurieren.

Wie in Abbildung 5.6.2 dargestellt, überträgt der Servo seine Kraft auf eine Öse, welche am beweglichen Teil des Druckkopfs angebracht ist. Um den Druckkopf nicht thermisch oder mechanisch zu beschädigen, fiel die Entscheidung auf eine Klebefestigung mittels eines hochfesten Epoxidharzklebers. Direkt auf dem Druckkopf wurde, um eine gleichmäßigere Belastung der Klebeflächen und eine ausgewogenere Kraftverteilung zu erreichen, eine längliches Stück Leiterplatte in die Ausnehmung des Druckkopfs eingepasst und verklebt. Auf diese Leiterplatte wurde anschließend die Öse geklebt, nachdem Versuche die Öse anzulöten, aufgrund zu geringer Festigkeit der Lötverbindung, scheiterten.

Zur Verbindung zwischen Öse und Servokreuz wurde Kupferlackdraht verwendet, da dieser einerseits leicht elastische Eigenschaften aufweist, andererseits jedoch, im Gegensatz zu ebenfalls getesteten Nylon-Drähten, langfristig sehr dehnungsstabil ist und somit seine Länge beibehält. Eine zusätzliche positive Eigenschaft ist außerdem, dass er an der Nahtstelle verlötet werden kann, was anders als bei verknüpften Nylon-Saiten, welche ebenfalls evaluiert wurden, eine dauerhaft feste Verbindung ergibt.

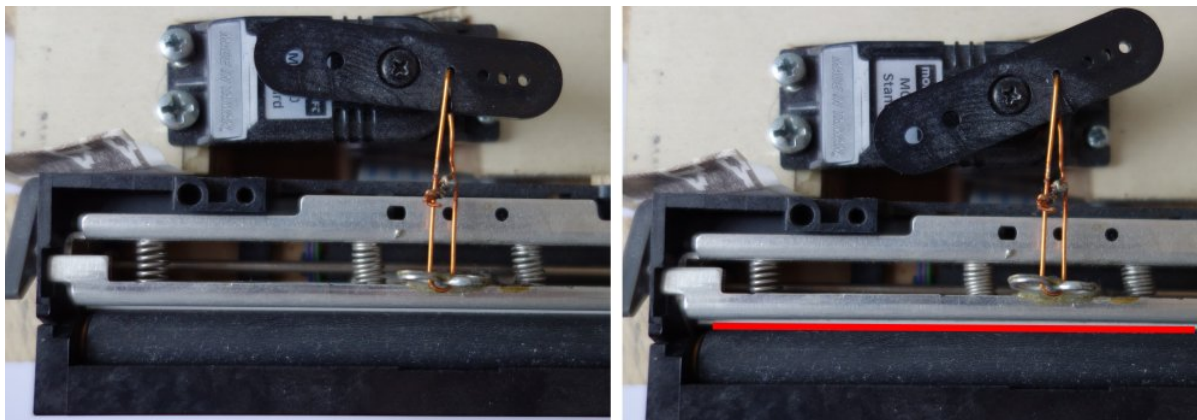


Abbildung 5.6.2: Einzugsmechanik geschlossen (links) sowie geöffnet (rechts)
Durch die vom Servomotor ausgeübte Kraft entsteht zwischen Walze und Druckkopf ein Spalt (rote Linie), durch welchen das Papier eingezogen werden kann.

Papierführung für Transport entgegen der Druckrichtung

Wie in Abschnitt 5.6 beschrieben, stellt die fehlende Papierführung bei Papiereinzug entgegen der Druckrichtung ein Problem dar, da das Papier nicht in den etwa 120° geneigten Papierausslass folgt, sondern geradlinig (180°) transportiert wird und sich daher im Druckwerk zwischen Papierführung und Druckkopfhalterung verklemmt. Da dieser Spalt allerdings durch die bewegliche Anordnung des Druckkopfs eine variable Breite aufweist, war es nicht möglich ein starres Umlenkelement anzubringen.

Diese Problemstellung wurde mittels einer Papierführung aus elastischem Papier realisiert, welche am Druckkopf mittels Epoxidharzkleber befestigt wurde und die in Abbildung 5.6.3 dargestellt wird. Um den Verschleiß durch Abrieb zu minimieren, wurde jene Fläche, an welcher das Papier beim Einzug schleift, mittels durchsichtigem Kunststoff-Klebeband verstärkt, welches außerdem bessere Gleiteigenschaften als Papier aufweist. Da allerdings an der Unterseite des Druckwerks Kunststoff-Rippen zur Erhöhung des Anpressdrucks zwischen Papier und Gummiwalze angebracht waren, mussten zuerst diese Rippen samt

2mm Kunststoff abgetragen werden um Platz für den Radius der Biegung der Umlenkung zu schaffen.

Zusätzlich zu der zuvor beschriebenen notwendigen Papierführung, stellte sich beim Papiereinzug das Problem, dass sich zwischen Druckkopf und der Metallschiene, auf welcher der Druckkopf fixiert ist, eine 1mm hohe, scharfe Kante befand. In dem Fall, dass sich das Papier beim Einlegen in die Papierführung des Gehäuses in dieser Kante verfängt, hätte es keinen Kontakt mehr mit der Transportrolle des Papiervorschubs gegeben, wodurch ein erfolgreicher Papiereinzug nicht möglich gewesen wäre. Daher wurde zusätzlich zu dem Umlenkelement aus Papier, welches sich in Richtung des Papiereinzugs nach dem Keramikelement des Druckkopfes befindet, die Kante vor dem Druckkopf durch einen Epoxydharzkleber mit Metallanteil vergossen. Dadurch wurde erreicht, dass zwischen Druckkopf und Metallschiene nun keine scharfe 90° Kante mehr existiert, stattdessen bildete das Epoxydharz einen runden Bogen aus, welchen das eingelegte Papier entlang rutscht, bis es schlussendlich auf die Transportrolle des Papiervorschubs trifft und durch das Druckwerk bewegt werden kann.

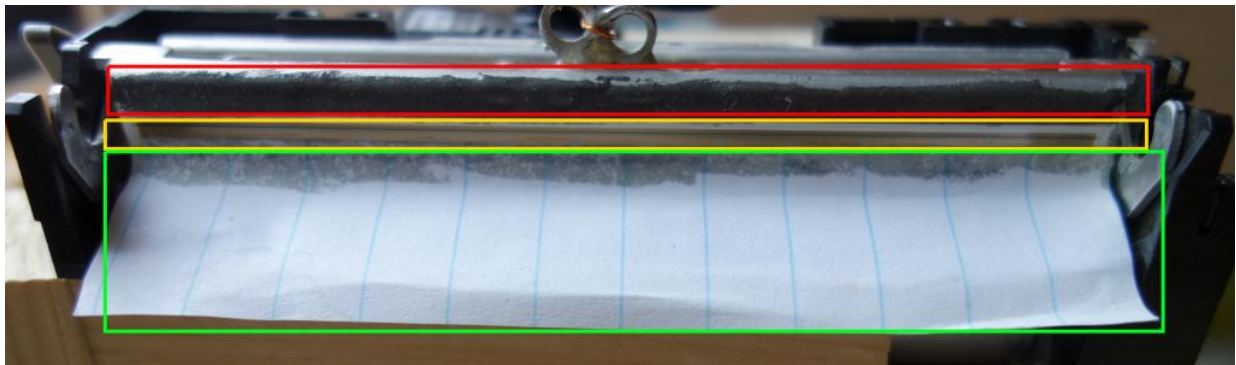


Abbildung 5.6.3: Papierführung innerhalb des Druckwerks

Mit Epoxydharz vergossene Kante zwischen Druckkopf und tragender Metallschiene (rot), Druckkopf auf Keramikträger (gelb), Papier-Umlenkvorrichtung, welche den Papiereinzug entgegen der Druckrichtung ohne Ausbau der Transportwalze ermöglicht (grün)

5.7 Schaltungsaufbau

Der Schaltungsaufbau wurde mittels einer Kombination aus freier Verdrahtung sowie einer Lochrasterplatine, welche im folgenden Unterkapitel näher beschrieben wird, realisiert.

Da das Arduino-Board mit Steckleisten auf Federkontaktbasis ausgestattet ist, wurde aus

Gründen der Flexibilität das Arduino-Board mit den anderen Leiterkarten über speziell angefertigtes Kabel verbunden, welche einerseits aus flexiblem Litzendraht aufgebaut sind, andererseits aber ein starres Ende aufweisen, um in die Arduno-Steckerleisen-Buchse gesteckt werden zu können.

Mangels passender Buchse, wurde die Kontaktierung des Interface-Boards direkt verlötet und mit Zugentlastung versehen. Der Power-Print hingegen erhielt Schraubklemmen und wurde über Kabel mit dem Rest des Systems (NIMH-Akkupack, Arduino, Servo, Thermodruckwerk) verbunden.

5.7.1 Verkabelung

Der Belegdrucker besteht aus folgenden elektronischen Einheiten:

- IFV Controllerboard
- Thermodruckwerk LTPV
- „Power-Print“
- Arduino Leonardo Board
- NIMH-Akkupack
- Servomotor

Das LTPV-Thermodruckwerk wird bereits mit den passenden Flachkabeln ausgeliefert, um es direkt mit dem IFV-Controllerboard verbinden zu können, wobei Falschmontage oder Verpolung durch das Steckerdesign ausgeschlossen ist. Daher wird im Folgenden auf diese Verbindung nicht weiter eingegangen.

Besonders zwischen den komplexeren Hauptkomponenten IFV-Controllerboard, Ardunio-Leonardo, sowie „Power-Print“ gibt es eine aufwändigere Verkabelung, wobei die Pinbelegung dieser Module in den Tabellen 5.1, 5.2 und 5.3 im Detail aufgelistet wird. Der NIMH-Akkupack ist dabei an die BAT-Klemme des Power-Prints angeschlossen, der Servo erhält seine Versorgungsspannung von der Versorgungsspannung des Druckwerks. Der aktuelle Winkel wird direkt über den Ardunio *General Purpose Input/Output* (GPIO) Pin 10 angesteuert, hierbei verbindet, zur Vermeidung von induktiven Fehlsteuerspannungen, ein 1kOhm Pull-Down Widerstand die Steuerleitung mit Masse.

Power-Print Klemme	Verbindungskabel Ziel
CTR	Arduino GPIO #9
PWR_OUT+	IFV-Controllerboard Vp, Servo+, Anode IR-LED Lichtschranke
PWR_OUT-	Servo -, Kathode IR-LED Lichtschranke
PWR_IN+	+5V IFV-Controllerboard, +5V Arduino (VK)
PWR_IN-	GND IFV-Controllerboard, GND Arduino (VK)
BAT+	NIMH-Akkupack +
BAT-	NIMH-Akkupack -

Tabelle 5.1: Belegung der Schraubklemmen des Power-Prints (VK = dient zusätzlich als Verbindungsklemme)

Pin Nr	Bezeichnung	Verbindungskabel Ziel
1-5	Vp	BAT+ Power-Print
6-10	GND	PWR_IN-, Power-Print
11	Vcc	PWR_IN+, Power-Print
12	ST2 (status 2)	Arduino GPIO, Pin6
13	RxD	Arduino GPIO Pin1, TxD
14	TxD	Arduino GPIO, Pin0, RxD
15	SBUSY (serial busy)	Arduino GPIO, Pin2
27	!FEED	Arduino GPIO, Pin3
28	!BFEEED	Arduino GPIO, Pin4
30	!PERR (error)	Arduino GPIO, Pin5
31	ST0 (status 0)	Arduino GPIO, Pin8
32	ST1 (status 1)	Arduino GPIO, Pin7

Tabelle 5.2: Belegung der Anschluss-Pins des IFV-Controller-Boards

5.7.2 Power-Print

Wie in Abbildung 5.7.1 ersichtlich, sind auf dem „Power-Print“ einerseits alle Komponenten untergebracht, welche für die Erzeugung der Ladespannung des NIMH-Akkupacks notwendig sind, andererseits befindet sich darauf das Leistungsrelais, welches den Leistungselektronik-Teil des Thermodruckwerkes sowie den für den Papiereinzug notwendigen Servo nur bei Bedarf mit der Batteriespannung versorgt. Das Relais wird mittels eines kleinen *Solid State Relais* (SSR) von einem GPIO-Pin des Arduino angesteuert, um den nötigen Schaltstrom aufbringen zu können. Außerdem leitet das Solid State Relais die induzierte Spannung der Schaltspule ab, welche entsteht, sobald das Relais geöffnet wird. Gespeist wird der Power-Print von der „Power“-Steckleiste des Arduino-Moduls, welche direkt mit den 5V-Leitungen der USB Schnittstelle verbunden ist.

Bezeichnung Leiste	Pin Nr	Bezeichnung	Verbindungskabel Ziel
POWER	-	GND	PWR_IN-, Power-Print
POWER	-	5V	PWR_IN+, Power-Print
GPIO	0	RxD	IFV TxD
GPIO	1	TxD	IFV RxD
GPIO	2	-	IFV SBUSY
GPIO	3	-	IFV !FEED
GPIO	4	-	IFV !BFEEED
GPIO	5	-	IFV !PERR
GPIO	6	-	IFV ST2
GPIO	7	-	IFV ST1
GPIO	8	-	IFV ST0
GPIO	9	-	CTR Power-Print
GPIO	10	-	Servo CTR
ANALOG IN	1	A1	Emitter Photo-Transistor Lichtschranke

Tabelle 5.3: Belegung der Anschluss-Pins des Arduino-Leonardo-Boards

Wie in Abbildung 5.7.2 ersichtlich, wurde der Power-Print mittels einer Lochraster-Platine aufgebaut. Dadurch konnte ein flexibler und schneller Aufbau erreicht werden, der dennoch die notwendige mechanische Stabilität erreicht, welche für die anschließende Verwendung des Prototyps notwendig ist.

5.7.3 Lichtschranke

Die Abläufe beim Einzug eines Papier-Einzelblattes gliedern sich in folgende Schritte:

- Mittels Servo wird der Anpressdruck aufgehoben, den der Druckkopf auf die Transportwalze ausübt
- Anschließend wird das Druckwerk angewiesen, das Papier gegen die Druckrichtung zu transportieren
- Wurde das Papier vollständig eingezogen, so wird der Servo angewiesen, den Anpressdruck für den nachfolgenden Druckvorgang wiederherzustellen

Um die oben aufgelisteten, notwendigen Abläufe zum Einzug eines Papier-Einzelblattes automatisiert ablaufen lassen zu können, ist es notwendig, über einen Sensor Informationen zum Status des Einzelblattes erfassen zu können. Beispielsweise soll kein Einzugvorgang

sowie folgender Druckvorgang ausgeführt werden, wenn sich kein Einzelblatt im Papierschlacht befindet. Weiters ist es nicht möglich, über die Dauer des Transport-Vorgangs genau auf die transportierte Papierlänge zu schließen, da einerseits ein gewisser Schlupf zwischen Transportwalze und Papier vorherrscht und es andererseits oft einige Zeit dauert, bis die Transportwalze, beim Beginn des Einzugvorgangs, Kraft auf das Papier auswirken kann.

Als Sensor zum Messen des Papierstatus wurde daher eine Lichtschranke realisiert, welche wie in Abbildung 5.7.3 dargestellt, aus einer Infrarot-LED sowie einem Foto-Transistor besteht. Die IR-LED leuchtet den Foto-Transistor an, wodurch ein Strom zwischen Kollektor und Emitter zu fließen beginnt. Dieser Strom führt zu einem messbaren Spannungsabfall an einem Widerstand zwischen Emitter und Masse, welcher über einen analogen Eingang am Arduino erfasst werden kann. Als Foto-Transistor kam ein NPN-Silizium-Transistor des Typs BPY62-4² der Firma OSRAM zum Einsatz, wobei die Basis, wie im Schaltplan dargestellt, unbeschaltet blieb.

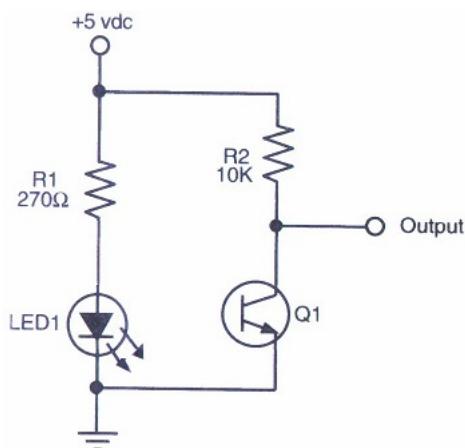


Abbildung 5.7.3: Schaltungsprinzip Lichtschranke

Beide Komponenten sind vor der Transportwalze angebracht. Die Lichtschranke ist daher unterbrochen, sobald sich ein Blatt Papier im Papierschlacht befindet. Wird nun das Blatt Papier entgegen der Druckrichtung in das Druckergehäuse eingezogen, so wird die Lichtschranke geöffnet, sobald das Blatt nahezu vollständig in das Druckwerk transportiert wurde. Auf diese Weise kann die Lichtschranke auch zur Positionsbestimmung während des Einzugvorgangs genutzt werden.

²http://www.osram-os.com/Graphics/XPic9/00101791_0.pdf (zuletzt abgerufen: 2016-06-14)

5.7.4 Akustischer Signalgeber

Um dem/der Benutzer/-in des Belegdruckers, unabhängig vom PC, eine akustische Rückmeldung über den aktuellen Status geben zu können, wurde ein Miniatur-Lautsprecher („PC-Speaker“) mit 8 Ohm Impedanz verbaut. Tests haben ergeben, dass die vom Arduino auf den Pins bereitgestellte Ausgangsleistung für die Erzeugung eines gut hörbaren Tons ausreichend ist, daher kann auf eine zusätzliche Verstärkerschaltung verzichtet werden, was, wie in Abbildung 5.7.4 dargestellt, der Einfachheit des Schaltungsaufbaus zu Gute kommt.

Die Ansteuerung des Lautsprechers erfolgt mittels der bereits von der Arduino-Bibliothek bereitgestellten Funktion `tone(int pin, int frequency, int duration)`, welche von der im ATmega enthaltenen PWM-Engine Gebrauch macht. Um das Software-Design möglichst flexibel und erweiterbar zu gestalten, wurde für die Ansteuerung des Lautsprechers ein dedizierter Kommando-Code in das im Kapitel 4.2 näher beschriebene Protokoll zwischen PC und Arduino aufgenommen. Damit kann der Treiber am PC die Ausgabe von Tönen veranlassen und somit ist eine Vielzahl von akustischen Rückmeldungen möglich, ohne die Software im Arduino explizit anpassen zu müssen.

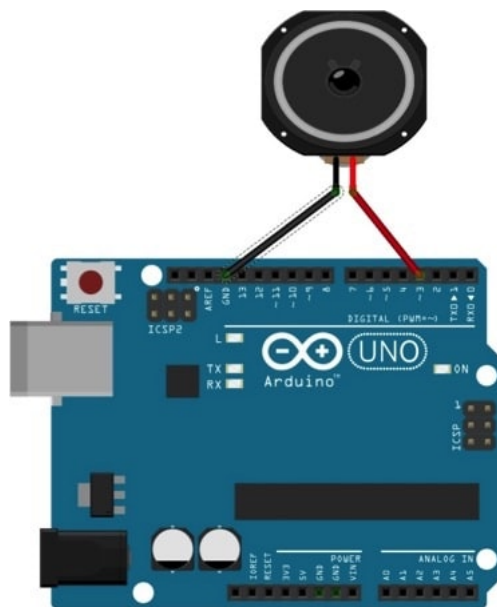


Abbildung 5.7.4: Schematische Darstellung der direkten Ansteuerung des Lautsprechers über einen PWM-fähigen Output-Pin des Arduino

Quelle: <http://diyhacking.com/arduino-speaker/> aufgerufen 2016-07-21

5.8 Mechanischer Aufbau

Der Prototyp wurde aus einer Kombination von verleimtem Sperr- und Vollholz konstruiert, wobei folgende Eigenschaften für die Materialwahl ausschlaggebend waren:

- Flexibel anpassbar an sich ergebende Anforderungen
- Leicht zu verarbeiten
- Verwindungssteif gegenüber den hohen, vom Servomotor ausgehenden, Kräften und dennoch leicht
- Hochohmig, dadurch keine Gefahr von Kurzschlüssen während der Konstruktions- und Experimentierphase

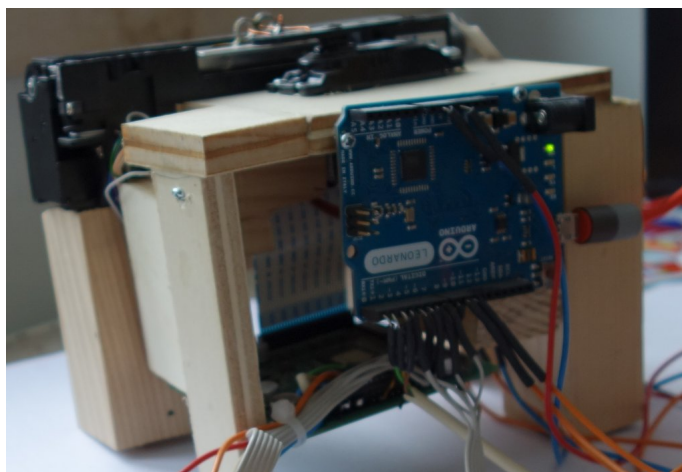


Abbildung 5.8.1: Mechanischer Aufbau basierend auf verleimtem Holzrahmen

Wie in Abbildung 5.8.1 dargestellt, hat die Holzkonstruktion die Form eines vierbeinigen Tisches, wobei an der Oberseite das Druckwerk sowie der Servomotor angebracht sind. Direkt unter dem Druckwerk befindet sich die in Abschnitt 5.3 näher beschriebene Controllerplatine, eine Notwendigkeit, die sich aus der nur wenig flexiblen Datenleitung (Flachbandkabel) zwischen Thermodruckwerk und Controllerplatine ergibt.

Der Arduino, welcher sowohl Steuerungsaufgaben ausführt, als auch die USB-Verbindung mit dem PC übernimmt, ist wiederum leicht zugänglich auf der Rückseite der Konstruktion montiert. Die Verbindungen zur Controllerplatine des Thermodruckwerks sowie zum, in Abschnitt 5.7.2 beschriebenen, Power-Print wurden über eigens angefertigte

steckbare Kabel ausgeführt. Dies erlaubt eine flexible Umkonfigurierung in späteren Entwicklungsphasen. Weiters stehen dadurch sämtliche Signal- und Versorgungsleitungen als Messpunkte zur Verfügung.

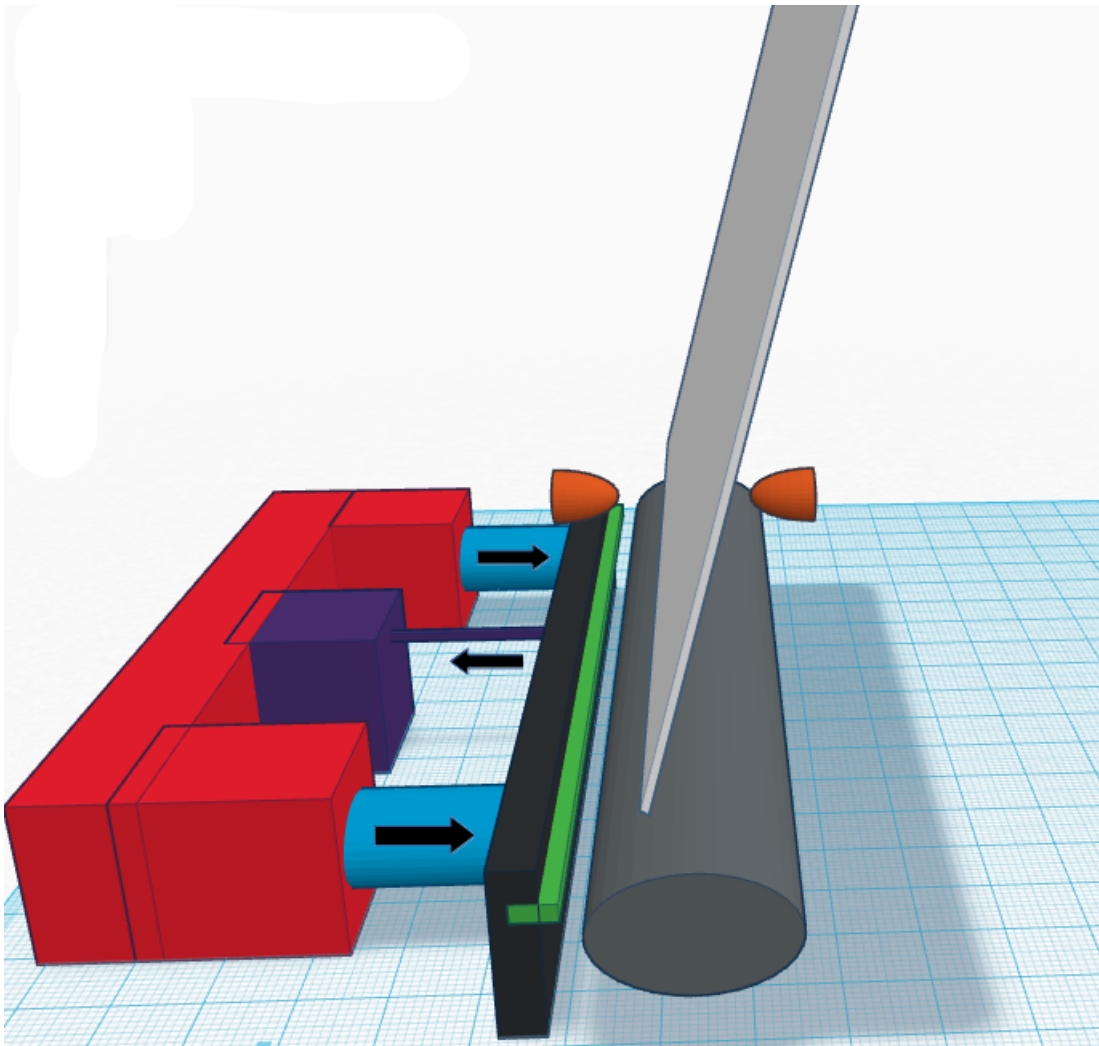


Abbildung 5.8.2: 3D-Modell des mechanischen Aufbaus

Folgende Komponenten werden dargestellt: Papierablage (hellgrau), Gummiwalze für Papiertransport (dunkelgrau), Keramik-Druckkopf (grün), Druckkopfhalterung (schwarz), Holzrahmen (rot), Federn stilisiert (blau), Servo (violett), Lichtschranke (orange). Die Kräfteinwirkung der Federn und des Servos auf die Druckkopfhalterung wird durch die schwarzen Pfeile dargestellt.

5.9 Gehäuse

Das realisierte Gehäusedesign nimmt sich Anleihen an dem, in Kapitel 3 vorgestellten, Design von Pavlovic. Aufgrund technischer Gegebenheiten sowie des Platzbedarfs des Thermo-Druckwerks und der dazugehörigen Elektronik sowie der abweichenden Fertigungsmethode, wurden folgende Eigenschaften adaptiert:

- Der leicht vorstehende Unterteil des Gehäuses wurde vergrößert, um das Druckwerk sowie die Platine des Arduino aufnehmen zu können.
- Statt gebogener und geschwungener Kanten weist das Gehäuse gerade Flächen auf, um den Fertigungsaufwand bei manueller Fertigung zu reduzieren.
- Um einen veränderbaren Winkel der Papierführung zu ermöglichen, ist die Papierauflage kein fixer Teil der Gehäuseausenfläche, sondern stattdessen eine separate Platte, welche mittels Scharnier mit den starren Gehäuseteilen verbunden ist.
- Durch die Änderung der Papierführung, ergibt sich eine verringerte Gehäusehöhe, was wiederum eine Reduktion des nutzbaren Volumens zur Folge hat. Um dennoch ausreichend Platz für das eingezogene Papier (welches während des Druckvorgangs vollständig im Gehäuse untergebracht werden muss) sowie den Akkumulator zu erhalten, wurde die Tiefe des Gehäuses erhöht.

Das, in Abbildung 5.9.1 aus verschiedenen Winkeln abgebildete, Gehäuse des Belegdruckers wurde aus 2,5mm starkem, weißen Polystyrol gefertigt. Dazu wurden, mit einer Bandsäge, aus einer Polystyrolplatte, die jeweiligen Einzelteile ausgeschnitten und sofern starre Verbindungen zulässig sind mittels Spezial-Kunststoffkleber verklebt. Dies hat den Vorteil, dass für jene Verbindungen keine Schraubenköpfe an der Außenseite des Gehäuses notwendig sind, was der Optik besonders im für den/die Benutzer/-in sichtbaren Front-Bereich des Gehäuses aufwertet. Gehäuseteile, welche für spätere Wartungs- und Adaptionsarbeiten abnehmbar bleiben sollten, werden mittels 90° Winkel, welche an der Gehäuseinnenseite angebracht sind, verschraubt. Sämtliche, für die Papierführung verwendeten, Flächen wurden mit Metallscharnieren an den starren Gehäuseflächen befestigt, um auch nach der abgeschlossenen Fertigung des Gehäuses die Möglichkeit bieten zu können, die Winkel, mit denen das Papier am Druckwerk angelegt wird, adaptieren zu können.

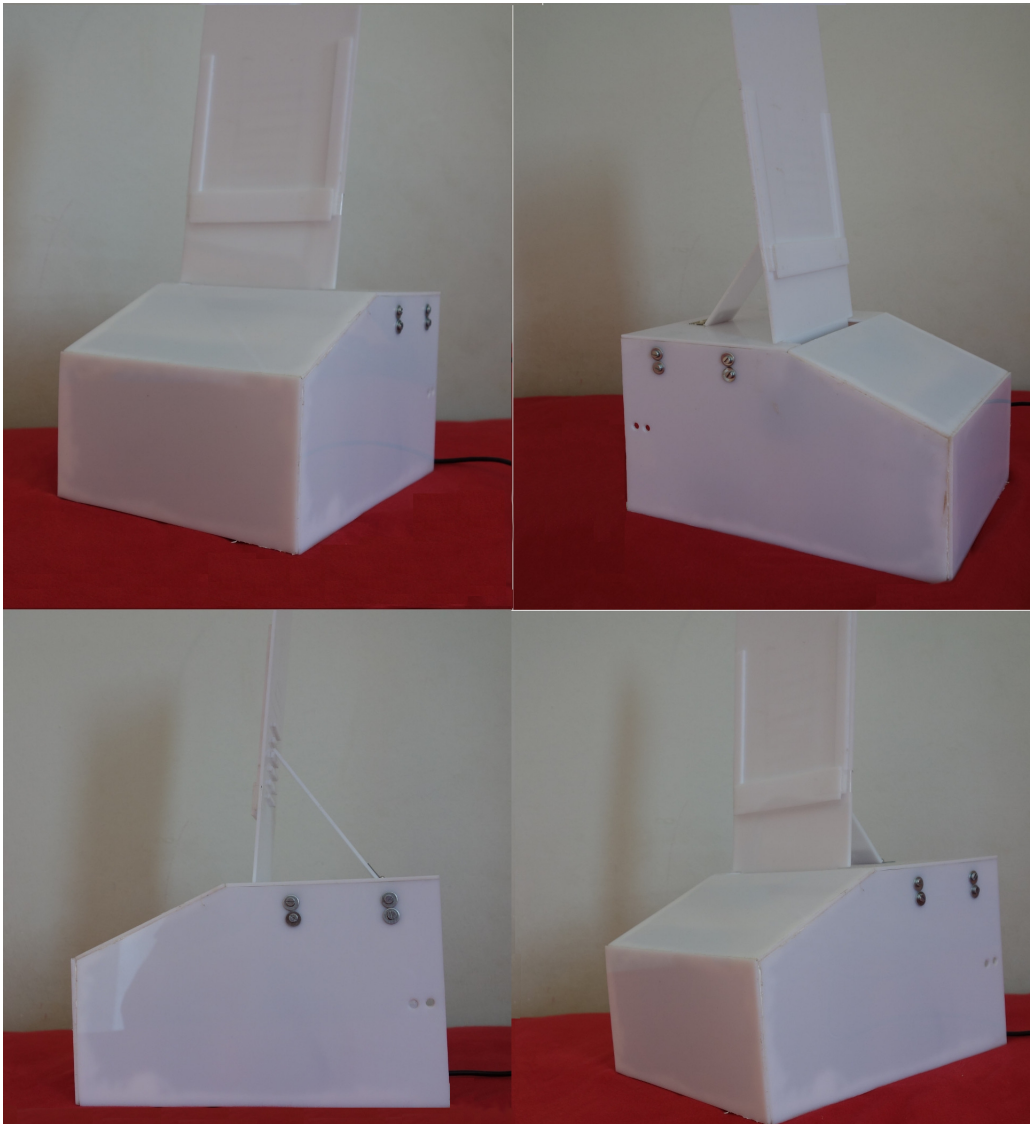


Abbildung 5.9.1: Im Rahmen der Arbeit entwickeltes Polystyrol-Gehäuse

Fazit und Ausblick

6.1 Fazit

Im Rahmen dieser Diplomarbeit wurde, auf Basis der Designstudie von Pavlovic [17], ein Drucker für Senioren/-innen konzipiert, welcher die geforderten Eigenschaften von Wartungsfreiheit und einfacher Verwendung erfüllt. Dazu wurde ein Thermodruckwerk, welches für den industriellen Einsatz konzipiert ist, von Rolleneinzug auf Einzelblatteinzug umgebaut, wobei durch eine Umlenkvorrichtung sogar der Einzug entgegen der üblichen Papierlaufrichtung möglich ist, um, wie in den Anforderungen gefordert, nur ein Fach für Papier-Einzug und Papierausgabe zu erhalten. Durch die Integration in das Windows-Druckersubsystem ist nicht nur die Verwendung durch speziell angepasster Software möglich, stattdessen kann der Drucker mit jeder beliebigen Windows-Software angesprochen werden. Die Anforderung, nur mittels eines einzigen Kabels sowohl Datenübertragung als auch die Stromversorgung zu realisieren, wurde mittels eines Pufferakkumulators umgesetzt, wobei für die Zukunft Konzepte aufgezeigt wurden, den Verschleißteil Pufferakkumulator durch die Versorgung über USB 3.1 „Power Profiles“ überflüssig zu machen. Für eine eventuelle Kleinserie wurden zudem Möglichkeiten recherchiert, die Kosten für die eingesetzten Komponenten zu senken.

6.2 Ausblick

Für eine Fertigung des vorgestellten Druckers in Kleinserie wäre eine Kostenoptimierung der verbauten Komponenten notwendig. Hohes Einsparungspotential liegt im Druckwerk

selbst, hier wäre die Verwendung eines LTPV-kompatiblen Druckwerks eines Alternativherstellers möglich. Ebenso könnten, durch die Verwendung einer Arduino-Leonardo kompatiblen Mikrocontroller-Platine, die Kosten gesenkt werden. Bezüglich der IFV-Controllerplatine wäre es ebenso möglich, den Controller-Baustein selbst zu kaufen und eine eigene Leiterkarte zu designen oder einen günstigeren Alternativenanbieter ausfindig zu machen. Neben der Auswahl günstigerer Komponenten könnte auch ein späterer Umstieg auf eine Stromversorgung über USB-3.1 zur Kostensenkung beitragen, da damit der Pufferakkumulator samt Ladeelektronik überflüssig werden würde und somit auf den „Power-Print“, sowie sämtliche Bauteile, welche darauf verbaut sind, verzichtet werden könnte.

Ein weiterer Punkt welcher noch Klärung bedarf, ist die Verfügbarkeit von Thermopapier in einem für das Druckwerk und die gewählte Einzugsart passenden Format und Stärke. Im Rahmen dieser Arbeit wurde anfangs dünnes Thermopapier von Endlosrollen auf dickeres Normalpapier geklebt, ein Vorgehen, das zeitintensiv und bei höherem Papierverbrauch nicht praktikabel ist. Später konnte ein Thermopapier im Format A4 ausfindig gemacht werden, welches von der Firma Brother vertrieben wird und dort als Verbrauchsmaterial für Reisedrucker dient. Dieses Papier entspricht, bezüglich Dicke und Haptik, am ehesten normalem 80g Kopierpapier, muss jedoch aufgrund des A4-Formats zuerst zugeschnitten werden.

Geplant ist die Integration des Druckers in das LUI-System, ein Touch-basierendes Computersystems für Senioren/-innen, welches an der TU Wien entwickelt wurde. Dabei soll der Drucker schwerpunktmäßig für Aufgaben, wie den Belegdruck von Bankbelegen, Gutscheinen, dem Druck von ärztlichen Verschreibungen und Rezepten sowie ähnlich dem „Chat Machine“-Projekt, für den Druck von Emails oder Online-Information verwendet werden [11]. Dabei gliedert sich der Drucker in eine Reihe von Projekten und Abschlussarbeiten, welche unter dem Motto „Zurück zum Papier“ am Institut durchgeführt werden und unter dem Überbegriff „Papyrus“ zusammengefasst werden. Schlussendlich wird die Integration des Druckers in diese Projekte über den Erfolg des Belegdruckers beim Einsatz im Feld entscheiden.

Appendix - Installationsanleitung Belegdruckertreiber

Ziel bei der Entwicklung des Druckertreibers war, sämtliche selbst entwickelten Bestandteile des Treibers im Userspace ausführen zu können. Dies erhöht die Wartbarkeit und reduziert den Entwicklungsaufwand. Weiters stehen im Userspace die vielfältigen Debugging-Möglichkeiten der gewohnten Standard-Tools zur Verfügung, während für Kernespace-Komponenten spezielle und teilweise stark eingeschränkte Hilfsprogramme genutzt werden müssen.

Dieses Design bedingt ein Zusammenspiel von verschiedensten Komponenten, deren Installation und Einrichtung hier beschrieben werden soll.

7.1 Download und Installation von Redmon

7.1.1 Beschreibung

Bei Redmon, dem „*Port Redirection Monitor*“, handelt es sich um einen Windows Port Monitor Treiber. Üblicherweise ist die Aufgabe von Port Monitor Treibern, die bereits aufbereiteten Rasterdaten an den Drucker zu senden, sie sind für die Abwicklung der Kommunikation und der Datenübertragung zuständig. Beim Belegdrucker soll dieser Ablauf allerdings von einem separaten Userspace-Prozess gesteuert werden. Die Aufgabe von RedMon bei diesem Anwendungsfall ist, einen Treiber für eine „virtuelle“ Druckerschnittstelle bereitzustellen und die Rasterdaten von Windows entgegenzunehmen.

RedMon täuscht so dem Windows-Printing Subsystem vor, die Daten würden über eine „reale“ Hardware-Schnittstelle übertragen werden. Anschließend startet RedMon den eigentlichen Kontrollprozess im Userspace und übermittelt diesem die zu übertragenden Rasterdaten per Standardeingabe (stdin).

7.1.2 Download

RedMon ist unter der GNU General Public License lizenziert, vorkompilierte Pakete sind unter <http://pages.cs.wisc.edu/~ghost/redmon/> verfügbar. Beim Download ist darauf zu achten, die aktuelle Version für Windows-XP und höher herunterzuladen (zum Zeitpunkt des Verfassens der Arbeit ist Version 1.9 aktuell), da für Windows-9x zusätzlich eine ältere Version angeboten wird.

7.1.3 Installation

Das Installationspaket von RedMon ist im ZIP-Format erhältlich. Dieses ist zuerst zu entpacken und anschließend ist die Datei setup.exe (für 32-bit Windows-Installationen) beziehungsweise setup64.exe (für 64-bit Windows-Installationen) zu starten und den Hinweisen des Installationsprogramms Folge zu leisten.

7.2 Installation des Belegdrucker UniDRV-Minitreibers

7.2.1 Start des Installationsassistenten

Der eigentliche UniDrv Minitreiber besteht, wie in Abschnitt 4.1 erläutert, aus einer GPD-Datei sowie einer INF-Datei, welche für den Installationsvorgang notwendige Informationen enthält. Da es sich bei dem Belegdrucker nicht um eine Plug&Play-kompatible Hardware handelt, muss dessen Treiber wie in Abbildung 7.2.1 dargestellt über den Menüpunkt „Legacyhardware hinzufügen“ des Windows-Gerätmanagers installiert werden.

7.2.2 Auswahl „Manuelle Installation“

Anschließend erscheint ein Windows-Assistent, der den/die Benutzer/-in durch die weitere Installation des Treibers begleitet. Im ersten Schritt muss der/die Benutzer/-in dabei angeben, ob ein Treiber automatisch gesucht und installiert werden soll, oder ob der/die Benutzer/-in den Treiber manuell auswählen möchte. Wie in Screenshot 7.2.2 aufgezeigt, ist es notwendig, die dem Treiber beiliegende INF-Datei manuell auszuwählen.



Abbildung 7.2.1: Manuelle Installation über den Menüpunkt „Legacyhardware hinzufügen“ des Windows-Gerätmanagers



Abbildung 7.2.2: Der Belegdruckertreiber erfordert eine manuelle Auswahl der dem Treiber beiliegenden INF-Datei

7.2.3 Auswahl des Typs der zu installierenden Hardware

Für den Fall, dass ein Gerät in Betrieb genommen werden soll, für welches bereits Treiber installiert wurden oder für welche Microsoft Treiber als Teil von Windows bereitstellt, wird im folgenden Schritt des Treiber-Installations-Assistenten eine Auswahlliste von Treibern für diverse Geräte verschiedener Herstellerfirmen angezeigt. Da allerdings ein neuer Treiber installiert werden soll und bei der Darstellung zusätzlich keine Einschränkung der angezeigten Treibertypen erwünscht ist, muss in diesem Schritt der Eintrag „Alle Geräte anzeigen“ wie in Abbildung 7.2.3 ausgewählt werden. Anschließend kann, mittels des „Weiter“-Buttons, zur nächsten Seite des Installations-Assistenten gewechselt werden.

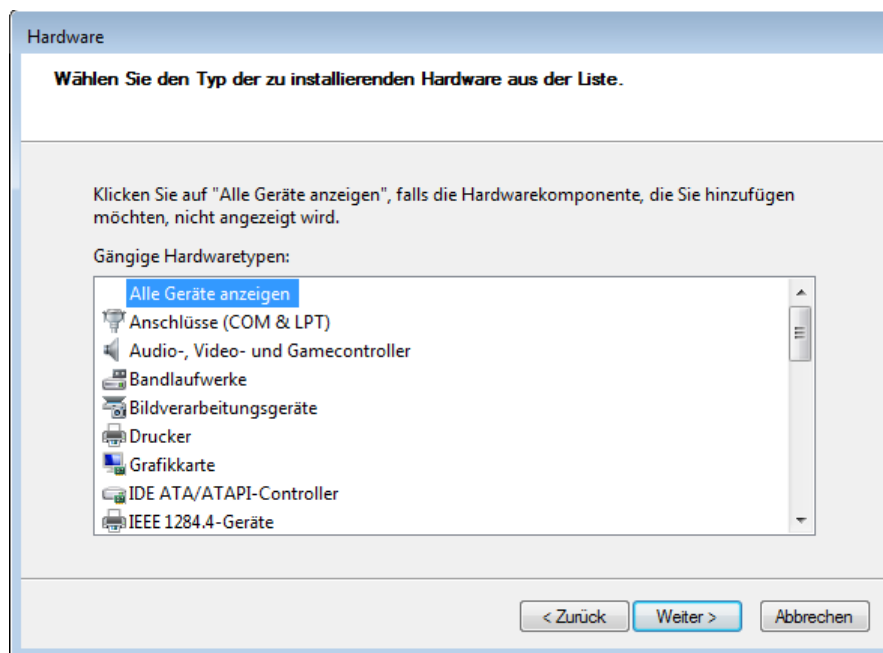


Abbildung 7.2.3: Um einen neuen Treiber manuell zu installieren, den Eintrag „Alle Geräte anzeigen“ selektieren

7.2.4 Manuelle Auswahl des zu installierenden Treibers

Auf der anschließend erscheinenden Seite des Treiber-Installations-Assistenten besteht, wie in Abbildung 7.2.4 dargestellt, die Möglichkeit, manuell eine Treiber-INF-Datei von einem Datenträger auszuwählen. Dazu klickt der/die Benutzer/-in auf den Button, welcher mit „Datenträger...“ beschriftet ist. Die Liste mit möglichen Treibern ist initial noch leer, erkennbar an der Beschriftung „(Eine Liste aller Geräte wird abgerufen)“.

Anschließend öffnet sich ein Dateiauswahl-Dialog, in welchem die Datei „Belegdrucker.ini“ auszuwählen ist. Nach dem Bestätigen der Auswahl wird die INF-Datei vom Treiber-Installations-Assistenten ausgewertet und mit den darin enthaltenen Informationen wird die Treiber-Auswahl-Liste vervollständigt. Wie in Abbildung 7.2.5 dargestellt, sollte der Eintrag „Belegdrucker“ ausgewählt und mit einem Klick auf den Button „Weiter“ bestätigt werden. Die Meldung, dass der Treiber keine gültige Signatur aufweist, kann getrost ignoriert werden, da der Belegdrucker-Treiber, welcher sich in das Windows Printing Subsystem integriert, keinerlei ausführbaren Code enthält, sondern nur aus Beschreibungsdateien besteht. Sämtlicher ausführbarer Code ist im separaten DTP enthalten, welcher als normaler Prozess im Userspace abläuft und somit kein Sicherheitsrisiko darstellt.

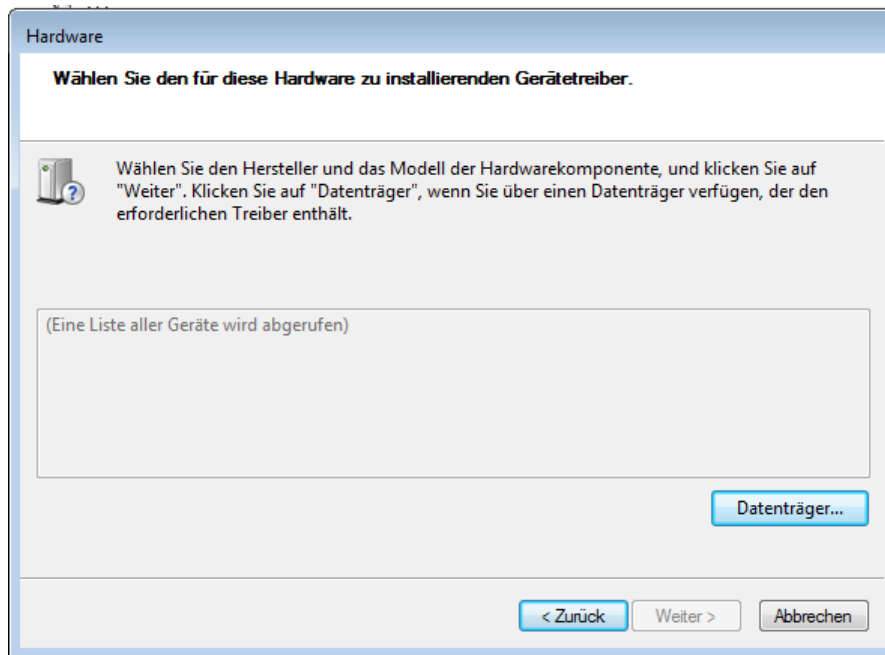


Abbildung 7.2.4: Auswahl der „INF“-Treiberdatei
Über den Button „Datenträger...“ kann manuell eine Treiber-INF Datei von einem Datenträger ausgewählt werden.

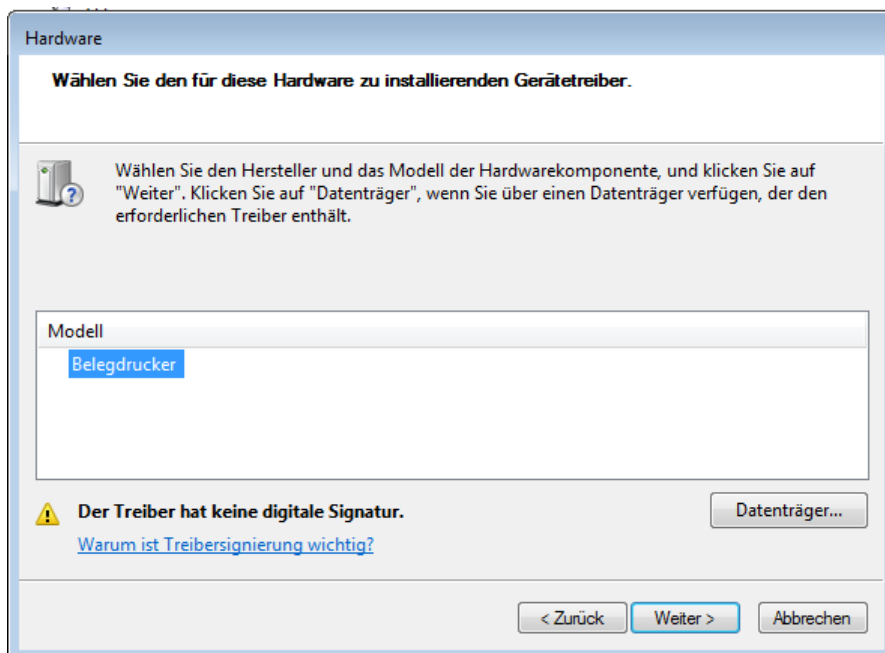


Abbildung 7.2.5: Auswahl des Belegdruckertreibers
Nach Auswahl der Belegdrucker-INF-Datei kann der Treiber für den Belegdrucker ausgewählt werden.

7.3 Anlegen des Belegdruckers

Nach der Installation des Treibers für den Belegdrucker, kann nun ein Drucker angelegt werden, dem dieser Treiber zugeordnet wird. Dazu ist entweder über die Windows-Systemsteuerung oder, ab Windows Vista, über den Startmenü-Eintrag „Geräte und Drucker“ der gleichnamige Dialog aufzurufen.

7.3.1 Aufruf des Assistenten zum Anlegen eines Druckers

Wie in Abbildung 7.3.1 dargestellt, kann über ein Kontextmenü in diesem Dialog ein neuer Drucker angelegt werden (Eintrag „Drucker hinzufügen“). Das Kontextmenü selbst erscheint, indem auf eine nicht von Icons besetzte weiße Fläche ein Rechtsklick durchgeführt wird. Dabei ist darauf zu achten, dass nicht aus Versehen ein bereits existierendes Gerät ausgewählt wird, da ansonsten das Kontextmenü dieses Geräts angezeigt wird.

Nach Auswahl des Eintrags „Drucker hinzufügen“ erscheint ein Dialog, welcher es dem/der Benutzer/-in ermöglicht, entweder einen „lokalen Drucker“ oder einen Netzwerkdrucker (kabelgebundenes Netzwerk, Wireless Local Area Network (LAN) beziehungsweise Bluetooth) hinzuzufügen. Da der Belegdrucker über USB als Schnittstelle angebunden ist, muss der Belegdrucker als „lokaler Drucker“ hinzugefügt werden. Im darauf folgenden Dialogschritt wird der/die Benutzer/-in aufgefordert, einen Anschluss für den zu installierenden Drucker auszuwählen. Da diesbezüglich, wie später in Abschnitt 7.4 genauer erläutert, spezielle Einstellungen notwendig sind, welche über einen separaten Dialog vorgenommen werden müssen, kann vorerst ein beliebiger Anschluss ausgewählt werden.

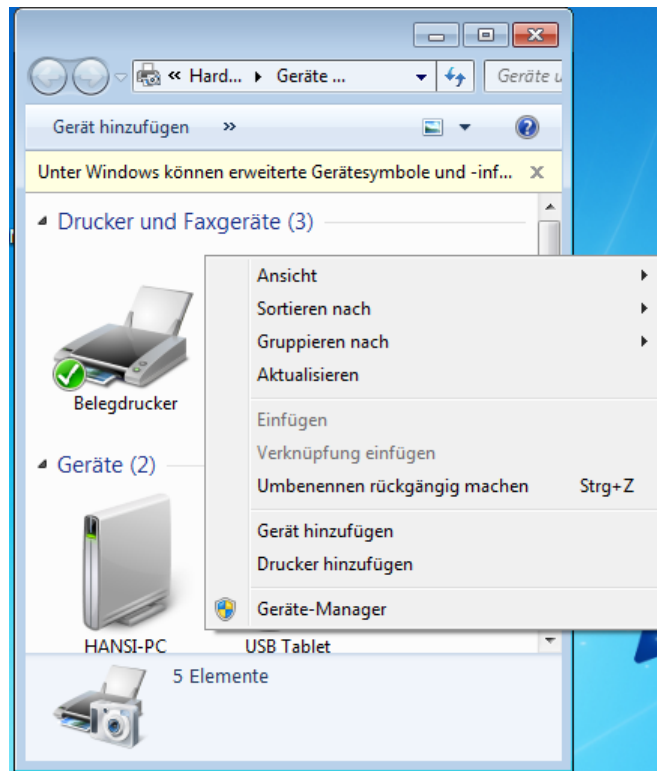


Abbildung 7.3.1: Eintrag „Drucker hinzufügen“
Über den Eintrag „Drucker hinzufügen“ im Kontextmenü des Dialogs „Geräte und Drucker“ können neue Drucker angelegt werden.

7.3.2 Auswahl des Belegdrucker-Treibers

Als ersten Schritt im neu geöffneten Assistenten ist die Auswahl eines Treibers für den neu anzulegenden Drucker notwendig. Wurde die im Abschnitt 7.2 beschriebene Installation des Treibers für den Belegdrucker erfolgreich abgeschlossen, so erscheint in der Liste der Eintrag „Belegdrucker“ des Herstellers „CE“, wie in Abbildung 7.3.2 dargestellt. Nachdem dieser Treiber wie in der Abbildung selektiert wurde, kann diese Auswahl durch Klicken des Buttons „Weiter“ bestätigt werden.

Anschließend sind keine weiteren relevanten Einstellungen im Rahmen des Assistenten zu tätigen, daher bitte den Anweisungen des Assistenten Folge zu leisten, bis der neue Drucker angelegt wird.

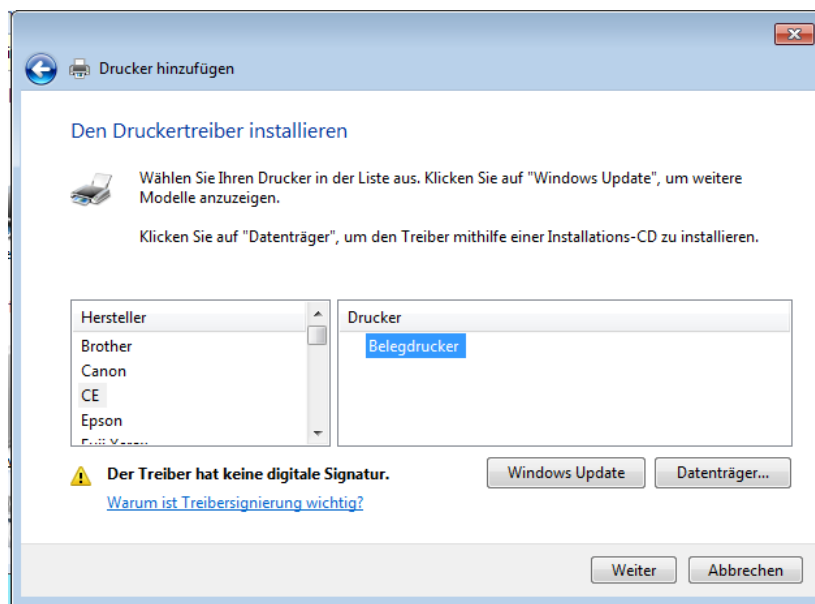


Abbildung 7.3.2: Auswahl des installierten Belegdruckertreibers
Nachdem der Treiber des Belegdruckers erfolgreich installiert wurde, steht dieser im Assistenten zum Anlegen eines neuen Druckers zur Auswahl bereit.

7.4 Konfiguration des Port-Redirectors

Nachdem der Drucker wie in Abschnitt 7.3 beschrieben angelegt wurde, muss nun der Port-Redirector konfiguriert werden, welcher die von Windows erzeugten Rasterdaten als virtueller Port entgegen nimmt und an den eigentlichen Druckprozess weiterleitet, welcher im Userspace ausgeführt wird.

7.4.1 Öffnen des Drucker-Eigenschaften Dialogs

Durch einen Rechtsklick auf den neu erstellten Belegdrucker erscheint ein Kontextmenü, über welches der Eigenschaften-Dialog des Druckers geöffnet werden kann. Dazu bitte den Menüpunkt „Druckereigenschaften“ im Kontextmenü wählen.

7.4.2 Einfordern von Administrator-Rechten

Das Anlegen eines RedMon-Ports ist nur Anwendern/-innen mit Administrator-Rechten erlaubt. Da jedoch die Systemsteuerung sowie alle enthaltenen Dialoge keine separaten Programme, sondern Teil der Bedienoberfläche „Windows Explorer“ sind, ist es nicht möglich den Drucker-Eigenschaften-Dialog einfach als User „Administrator“ zu starten.

Stattdessen bietet Windows an einigen Stellen die Möglichkeit an, über *User Access Control* (UAC) erhöhte Rechte zu erlangen.

Im Drucker-Eigenschaften-Dialog besteht diese Möglichkeit, indem der Button „Freigabeoptionen ändern“ geklickt wird. Dieser ist, wie in Abbildung 7.4.1 dargestellt, in der Registerkarte „Freigabe“ enthalten. Auch wenn anschließend keinerlei Freigabeeinstellungen verändert werden, läuft der Dialog daraufhin mit erhöhten Rechten und ermöglicht so die Installation des Port-Redirector-Treibers RedMon.

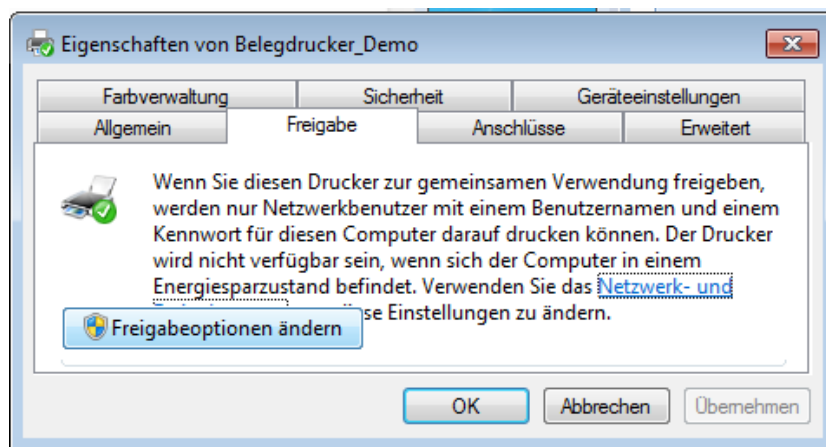


Abbildung 7.4.1: Anfordern höherer Rechte über den Button „Freigabeoptionen ändern“
Über den Button „Freigabeoptionen ändern“ fordert der Druckereigenschaften-Dialog erhöhte Rechte an, die zur Konfiguration des Port-Redirector-Treibers RedMon notwendig sind.

7.4.3 Anschluss / Port anlegen

Nachdem der Druckereigenschaften-Dialog nun über erhöhte Privilegien verfügt, kann unter der Registerkarte „Anschlüsse“ über den Button „Hinzufügen“ ein neuer „Redirected Port“ angelegt werden, wie in Abbildung 7.4.2 veranschaulicht. Durch Klick auf den Button „Neuer Anschluss...“ wird dieser neu angelegt und wird nun ebenfalls in der Liste der Anschlüsse im Druckereigenschaften-Dialog aufgelistet.

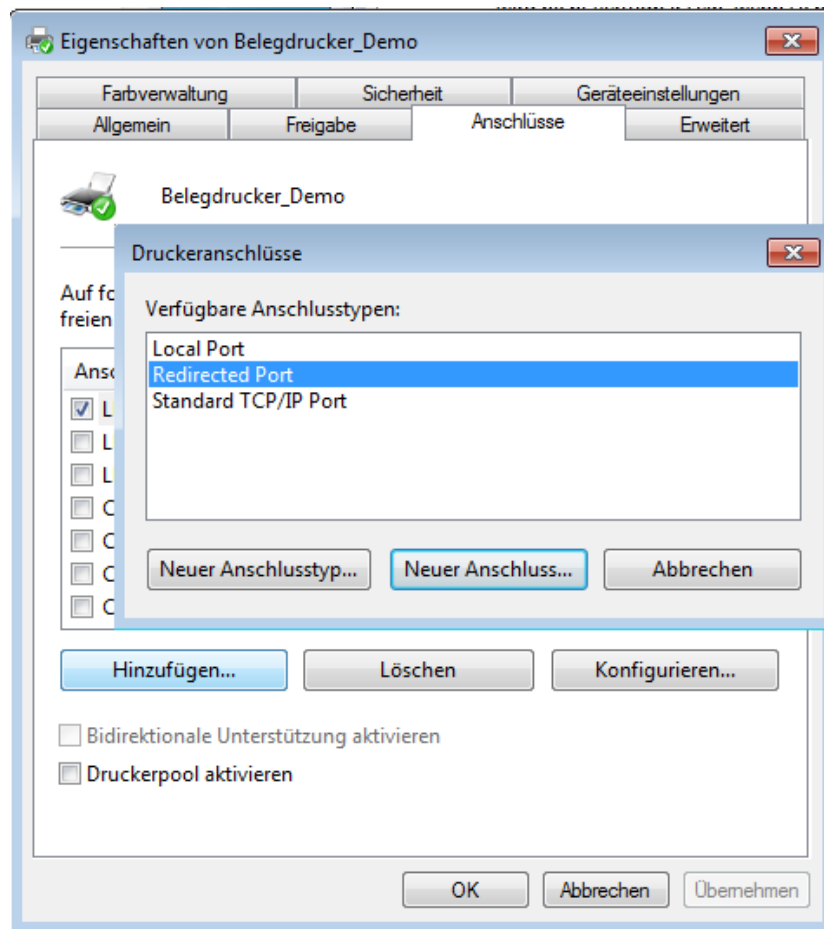


Abbildung 7.4.2: Anlegen eines neuen „Redirected Ports“ im Druckereigenschaften-Dialog

7.4.4 Anschluss / Port konfigurieren

Nach dem Anlegen des neuen Ports, kann dieser anschließend konfiguriert werden, indem der neue Port in der Anschluss-Liste ausgewählt wird und anschließend der Button „Konfigurieren...“ betätigt wird. Daraufhin öffnet sich der Port-Properties-Dialog, welcher bereits Teil von RedMon ist.

Wie in Abbildung 7.4.3 dargestellt, sollen die von Windows erzeugten Rasterdaten an das Userspace-Kontrollprogramm „QTController.exe“ weitergeleitet werden. Wird ein Druckauftrag gestartet, so nimmt RedMon die generierten Daten entgegen, startet seinerseits das Programm „QTController.exe“ und übergibt diesem die Rasterdaten per Standardeingabe.

Wichtig ist zusätzlich, dass in der „Output“-ComboBox der Eintrag „Program handles output“ ausgewählt ist. Damit wird RedMon angewiesen, dass die alleinige Verantwortung für die Verarbeitung der Rasterdaten dem angeführten Programm obliegt und RedMon selbst keine zusätzlichen weiteren Aktionen mit den entgegengenommenen Daten durchzuführen hat (wie zum Beispiel diese zusätzlich an einen „realen“ Drucker zu senden).

Über das Textfeld mit der Überschrift „Arguments for this program are“, können dem aufzurufendem Programm Argumente übergeben werden. Diese Funktionalität wird benutzt, um dem Userspace-Kontrollprogramm mitteilen zu können, über welchen virtuellen COM-Port es mit dem Arduino kommunizieren soll.

Zusätzlich ermöglicht RedMon noch die Angabe einer Log-Datei in welcher neben RedMon-Statusmeldungen auch die Standardausgabe des aufgerufenen Programms umgeleitet wird, ein Feature welches sich für die Fehlersuche als äußerst wertvoll erwiesen hat.

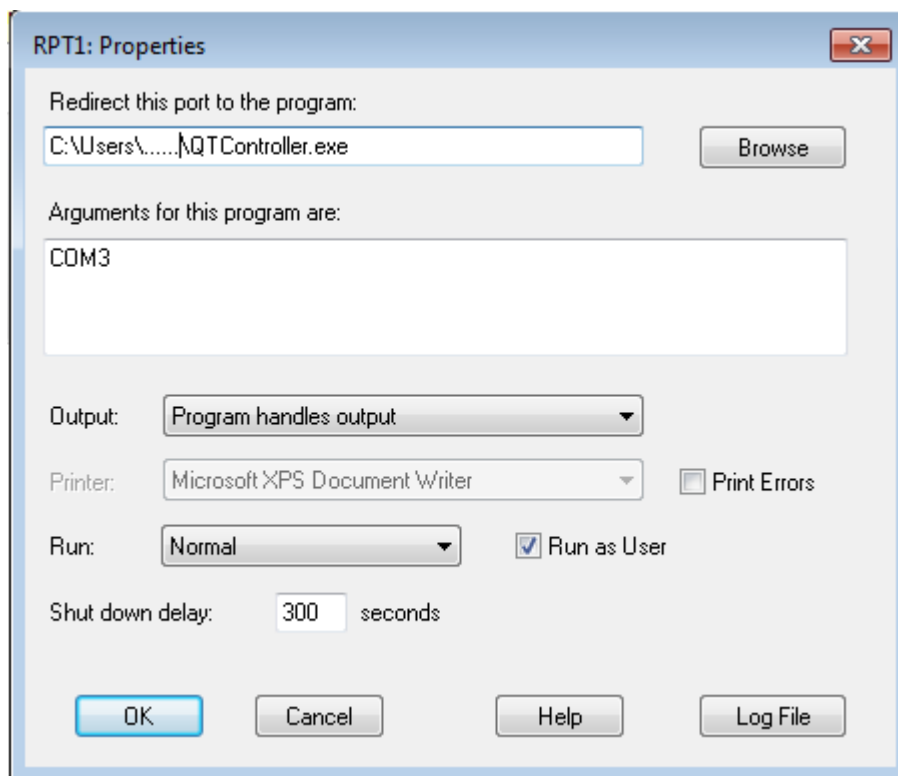


Abbildung 7.4.3: Konfigurationsdialog eines „redirected ports“, bereitgestellt von RedMon

Abbildungsverzeichnis

2.1.1 Bevölkerungsentwicklung bis 2075 [4]	10
2.1.2 Demographische Auswirkungen des „Babybooms“[4]	11
2.2.1 Movement Rank je nach Altersgruppe und Geschlecht [3]	13
3.3.1 3D Modell des Drucker-Gehäuses nach Pavlovic	26
3.3.2 Mit Schaumkarton verstärkte Gehäuse-Prototypen	27
4.1.1 Komponenten des Microsoft Universal Printing Driver	33
4.1.2 Komponenten des Windows Print Spoolers	34
4.1.3 Datenfluss vom Print-Processor bis hin zum Drucker	35
4.1.4 Spooling-Ablauf bei Druck auf einem lokalen Drucker	36
4.1.5 Dialog zum Anlegen eines Druckeranschlusses	37
5.1.1 Thermodruckwerk LTPV445	51
5.1.2 Pixel-Anordnung des Thermodruckwerks LTPV445 [8]	52
5.1.3 Druckbereich des Thermodruckwerks LTPV445 [8]	52
5.1.4 Datenübernahme und Ansteuerung des Thermodruckkopfes [8]	53
5.2.1 Dynamische Aufteilung für eine schwarze Linie [9]	56
5.2.2 Befehlsstruktur der PTV10P01-E CPU [9]	57
5.3.1 Controllerplatine IFV101-01B-E	58
5.3.2 Blockdiagramm des IFV101-01B-E Interface Boards [7]	59
5.4.1 Leonardo Arduino-Platine in der Draufsicht	60
5.4.2 Arduino-IDE mit geladenem „Sketch“	61
5.5.1 Ladekennlinie von NIMH-Akkumulatoren	65
5.5.2 Spannungsverlauf bei Entladung einer NIMH-Zelle mit 1,4Ah	66
5.5.3 USB 3.1 Power Delivery Architecture [20]	68
5.5.4 USB 3.1 Power Delivery Profiles [20]	69
5.6.1 Ansteuerung des Servos über ein PWM-Signal	71

5.6.2 Einzugmechanik geschlossen (links) sowie geöffnet (rechts)	72
5.6.3 Papierführung innerhalb des Druckwerks	73
5.7.1 Schematische Darstellung des „Power-Prints“	77
5.7.2 „Power-Print“ in Draufsicht	77
5.7.3 Schaltungsprinzip Lichtschranke	78
5.7.4 Schematische Darstellung der direkten Ansteuerung des Lautsprechers über einen PWM-fähigen Output-Pin des Arduino	79
5.8.1 Mechanischer Aufbau basierend auf verleimtem Holzrahmen	80
5.8.2 3D-Modell des mechanischen Aufbaus	81
5.9.1 Im Rahmen der Arbeit entwickeltes Polystyrol-Gehäuse	83
7.2.1 Manuelle Installation über den Menüpunkt „Legacyhardware hinzufügen“ des Windows-Gerätmanagers	89
7.2.2 Der Belegdruckertreiber erfordert eine manuelle Auswahl der dem Treiber beiliegenden INF-Datei	89
7.2.3 Um einen neuen Treiber manuell zu installieren, den Eintrag „Alle Geräte anzeigen“ selektieren	90
7.2.4 Auswahl der „INF“-Treiberdatei	91
7.2.5 Auswahl des Belegdruckertreibers	91
7.3.1 Eintrag „Drucker hinzufügen“	93
7.3.2 Auswahl des installierten Belegdruckertreibers	94
7.4.1 Anfordern höherer Rechte über den Button „Freigabeoptionen ändern“	95
7.4.2 Anlegen eines neuen „Redirected Ports“ im Druckereigenschaften-Dialog . . .	96
7.4.3 Konfigurationsdialog eines „redirected ports“, bereitgestellt von RedMon . . .	97

Tabellenverzeichnis

4.1	Mögliche Fehlerzustände der IFV101-01B Steuerungsplatine	47
5.1	Belegung der Schraubklemmen des Power-Prints (VK = dient zusätzlich als Verbindungsklemme)	75
5.2	Belegung der Anschluss-Pins des IFV-Controller-Boards	75
5.3	Belegung der Anschluss-Pins des Arduino-Leonardo-Boards	76

Nomenclature

ASCII	American Standard Code for Information Interchange
CSS3	Cascading Style Sheets 3
DC	Direct Current
DIB	Device Independent Bitmap
DIP	Dual Inline Package
DLL	Dynamic Link Library
DPI	Dots Per Inch
DTP	Data Transfer Process
EEPROM	Electrically Erasable Programmable Read-Only Memory
FTDI	Future Technology Devices International
GCC	GNU Compiler Collection
GDI	Graphics Device Interface
GNU	GNU is not Unix
GPD	Generic Printer Description
GPIO	General Purpose Input/Output
GPL	GNU General Public License
GUI	Graphical User Interface
HTML	Hypertext Markup Language

IC	Integrated Circuit
IDE	Integrated Development Environment
INF	Setup Information File
IR-LED	Infrarot-Leuchtdiode
ITK	Informations- und Kommunikationstechnologien
LGPL3	GNU Lesser General Public License 3
LION	Lithium-Ionen
LiPo	Lithium-Polymer
MinGW	Minimalist GNU for Windows
MSDN	Microsoft Developer Network
NiCd	Nickel-Cadmium
NIMH	Nickel-Metall-Hydrid
PC	Personal Computer
PDF	Portable Document Format
PIN	Personal Identification Number
PJL	Printer Job Language
PWM	Pulse Width Modulation
RPC	Remote Procedure Call
SMD	Surface Mounted Device
SRAM	Static Random Access Memory
SSR	Solid State Relais
TAN	Transaction Authentication Number
UAC	User Access Control
UART	Universal Asynchronous Receiver Transmitter

UniDRV	Microsoft Universal Printing Driver
USB	Universal Serial Bus
WDK	Windows Driver Kit
WLAN	Wireless Local Area Network
WPF	Windows Presentation Foundation
WYSIWYG	What You See is What You Get
XPS	Open XML Paper Specification

Literaturverzeichnis

- [1] Arduino. <http://www.arduino.cc/en/pmwiki.php?n=main/arduinoboardleonardo> accessed 2013-05-23.
- [2] Atmel. Atmega16u4/atmega32u4 8-bit microcontroller with 16/32 bytes of isp flash and usb controller - datasheet.
- [3] Bank4Elder. Report with general user needs, 3 2012.
- [4] Familie und Jugend Bundesministerium fuer Wirtschaft. 5. familienbericht 1999 - 2009. 2010.
- [5] Jennifer L. Davidson and Carlos Jensen. Participatory design with older adults: An analysis of creativity in the design of mobile healthcare applications. In *Proceedings of the 9th ACM Conference on Creativity & Cognition, C&C '13*, pages 114–123, New York, NY, USA, 2013. ACM.
- [6] Tad Hirsch, Jodi Forlizzi, Elaine Hyder, Jennifer Goetz, Chris Kurtz, and Jacey Stroback. The elder project: Social, emotional, and environmental factors in the design of eldercare technologies. In *Proceedings on the 2000 Conference on Universal Usability, CUU '00*, pages 72–79, New York, NY, USA, 2000. ACM.
- [7] Seiko Instruments Inc. Ifv101-01b-e interface board technical reference. 2006.
- [8] Seiko Instruments Inc. Ltp v series thermal printer mechanism technical reference. 2006.
- [9] Seiko Instruments Inc. Ptv10p01-e cpu ltp v series control cpu technical reference. 2006.
- [10] Mikko Jäkälä and Samuli Pekkola. From technology engineering to social engineering: 15 years of research on virtual worlds. *SIGMIS Database*, 38(4):11–16, October 2007.

- [11] D. Orradre S. Heuer L. Jungmann C. Becker C. Klotz. Chat-machine: An approach to a new inter-generation social device for elderly people. 2015.
- [12] Stephen Lindsay, Daniel Jackson, Guy Schofield, and Patrick Olivier. Engaging older people using participatory design. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '12, pages 1199–1208, New York, NY, USA, 2012. ACM.
- [13] Mohammad Mannan and P. C. van Oorschot. Security and usability: The gap in real-world online banking. In *Proceedings of the 2007 Workshop on New Security Paradigms*, NSPW '07, pages 1–14, New York, NY, USA, 2008. ACM.
- [14] Beth Meyer, Dominic G. Bouwhuis, Sara J. Czaja, Wendy A. Rogers, Matthias Schneider Hufschmidt, and James L. Fozard. Senior chi: How can we make technology elder-friendly?". In *CHI '99 Extended Abstracts on Human Factors in Computing Systems*, CHI EA '99, pages 81–82, New York, NY, USA, 1999. ACM.
- [15] H. H. Nap and H. P. de Greef. Self-efficacy & stress in senior computer interaction. In *Proceedings of the 28th Annual European Conference on Cognitive Ergonomics*, ECCE '10, pages 227–230, New York, NY, USA, 2010. ACM.
- [16] Office of Communications. Accessing the internet at home. *OFCOM*, 2009.
- [17] Goran Pavlovic. Onlinebanking-belegdrucker fuer senior/-innen. Master's thesis, TU Wien, 2014.
- [18] Statistika. Anteil der Nutzer von Online-Banking in Deutschland im jahr 2013 nach Altersgruppen, 11 2013.
- [19] Statistika. Anteil der Nutzer von Online-Banking in Deutschland in den Jahren 1998 bis 2013, 11 2013.
- [20] USB-IF. Usb power delivery specification 1.0. 2012.
- [21] USB-IF. Universal serial bus revision 3.1 specification. 2013.
- [22] John Vines, Mark Blythe, Paul Dunphy, and Andrew Monk. Eighty something: Banking for the older old. In *Proceedings of the 25th BCS Conference on Human-Computer Interaction*, BCS-HCI '11, pages 64–73, Swinton, UK, UK, 2011. British Computer Society.

- [23] John Vines, Mark Blythe, Stephen Lindsay, Paul Dunphy, Andrew Monk, and Patrick Olivier. Questionable concepts: Critique as resource for designing with eighty somethings. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '12, pages 1169–1178, New York, NY, USA, 2012. ACM.
- [24] John Vines, Paul Dunphy, Mark Blythe, Stephen Lindsay, Andrew Monk, and Patrick Olivier. The joy of cheques: Trust, paper and eighty somethings. In *Proceedings of the ACM 2012 Conference on Computer Supported Cooperative Work*, CSCW '12, pages 147–156, New York, NY, USA, 2012. ACM.
- [25] Mayer Wolfgang Friesdorf, Achim Heine. *Sentha - seniorenerechte Technik im haeuslichen Alltag: ein Forschungsbericht mit integriertem Roman*. Springer-Verlag, Berlin Heidelberg, 2007.
- [26] Larry Wood. Semi-structured interviewing for user-centered design. *Magazine interactions*, 1997:48–61.
- [27] Muna Khalil Yousef. Assessment of metaphor efficacy in user interfaces for the elderly: A tentative model for enhancing accessibility. In *Proceedings of the 2001 EC/NSF Workshop on Universal Accessibility of Ubiquitous Computing: Providing for the Elderly*, WUAUC'01, pages 120–124, New York, NY, USA, 2001. ACM.