Benutzerhandbuch Dokumentation Aufbauanleitung

El Fueradoro

Funkgesteuertes Zündsystem für Feuerwerke

Felix Pflaum (f.pflaum@gmail.com)

31.01.2015, 21:53

Wichtige Hinweise

Vor Inbetriebnahme sollte in jedem Fall diese Anleitung gelesen und verstanden werden! Speziell die folgenden, weiter hinten in der Anleitung detaillierter ausgeführten Hinweise müssen beachtet werden, um Personenschäden oder Beschädigungen an den Geräten zu vermeiden:

1. **Das Metallgehäuse des SMA-Antennenanschlusses an der Oberseite der Zündbox liegt auf Massepotential, die roten Anschlussklemmen nach dem Einschalten und unter Umständen auch noch Minuten nach dem Ausschalten bei 22,5 V. Ein Kurzschluss zwischen diesen beiden Punkten ist daher unbedingt zu vermeiden, um Schäden an der Schaltung und ungewollte Zündungen zu verhindern. Daher, sofern der Antennenanschluss nicht isoliert ist, beim Anschließen die Zünder immer zuerst mit der schwarzen Klemme, danach mit der roten verbinden, beim Abklemmen zuerst das Kabel an der roten, dann das an der schwarzen Klemme lösen. Weiterhin sollte man die Zündbox erst in ihrer endgültigen Position einschalten, so dass ein unbeabsichtigtes Herausziehen der Kabel tunlichst ausgeschlossen ist. Sicherheitshalber kann auch der SMA-Stecker nach dem Anschrauben mit Klebeband isoliert werden.**

2. **Devices nie ohne angeschlossene Antenne einschalten, um Rückreflexionen an offenen Steckern/Buchsen zu vermeiden, welche die Sendeendstufe zerstören könnten.**

3. **Keine Softwareupdates an Zündboxen durchführen, solange Zünder angeschlossen sind, da das Verhalten nach einem fehlgeschlagenen Update unvorhersehbar ist.**

Trotz aller Sorgfalt bei der Entwicklung von Hard- und Software sowie ausgiebigen Tests seitens der Entwickler kann keinerlei Garantie für Sicherheit und Funktionalität von *El Fueradoro* sowie keine Haftung für Sach- und Personenschäden, welche sich direkt oder indirekt aus dem Einsatz von *El Fueradoro* ergeben, übernommen werden. Es werden mit *El Fueradoro* dem Anwender lediglich Schaltpläne, Layouts und Programmcode sowie Hinweise zu Aufbau und Verwendung einer elektronischen Funkzündanlage, welche er frei weiter verbreiten und verändern darf, zur Verfügung gestellt. *El Fueradoro* ist und bleibt ein Hobby-Bastelprojekt, kein geprüftes oder in irgendeiner Form zertifiziertes Produkt!

Der Umgang mit Feuerwerkskörpern und explosivem Material unterliegt in Deutschland gesetzlichen Restriktionen, deren Einhaltung im Verantwortungsbereich des Anwenders liegt. *El Fueradoro* darf nicht von Personen unter 18 Jahren bedient werden.

Danksagung

Vielen Dank an alle, die in irgendeiner Form zur Entwicklung von *El Fueradoro* beigetragen haben!

Ein ganz spezielles Dankeschön an:

- Marc Weissmann, der mit seinem Projekt FIRE - Feuerwerk im Röthelheimpark Erlangen den Ansporn zu dieser Entwicklung gegeben und ihr von Beginn an immer voll vertraut hat.
- Das gesamte private Umfeld, das die Arbeit am Projekt stets mit Fassung getragen hat
- `mikrocontroller.net` für Hilfe in Sachen Hardware und Programmierung
- `feuerwerk-forum.de` für Steuersoftware und Erklärungen zum Thema elektrische Zündung
- `Saleae` für den grandiosen Logic 8, der wertvolle Dienste zur Erschließung des `RFM69` geleistet hat
- Die vielen tollen Open-Source- und Freeware-Lösungen, welche die Softwareentwicklung erst möglich gemacht haben:
  - Die GCC-Community für die `AVR-GCC-Toolchain` zum Programmieren des Mikrocontrollers
  - Die Eclipse-Foundation für die Programmierumgebung `Eclipse`
  - Kees Bakker und Thomas Holland für die Eclipse-Erweiterung `AVREclipse`
  - Peter Dannegger für den robusten, aber dennoch schlanken Bootloader `FastBoot`
  - `Subversion` und `TortoiseSVN`
  - LaTeX zur Erstellung des Handbuchs

Inhaltsverzeichnis

I Benutzerhandbuch **13**

1 Das System **15**
1.1 Aufbau . . . 15
1.2 Transmitterbox und Zündboxen . . . 16
1.3 Energieversorgung . . . 17
1.4 Die Status-LEDs . . . 19
1.5 Das LCD der Transmitterbox . . . 19
1.6 Schalter an den Zündboxen . . . 20

2 Vorbereitung des PCs **21**
2.1 Installation eines USB-RS232-Adapters . . . 21
2.2 Einrichtung des Terminalprogramms . . . 22
2.3 Einrichtung von *Pyro Ignition Control* . . . 23

3 Kommunikation zwischen PC und Devices **25**
3.1 Puttytel . . . 25
3.1.1 Befehlsübersicht . . . 25
3.1.2 Die Konfiguration . . . 26
3.1.3 Die Systemübersicht . . . 29
3.1.4 Manuelles Senden . . . 30
3.1.5 Funkmodul-Zugriff . . . 31
3.2 GUI . . . 32

4 Firmwareupdate **33**

II Dokumentation **35**

5 Schaltpläne & Layouts **37**

6 Pinbelegung **43**

7 Software **45**

III Aufbauanleitung **47**

8 Materialliste **49**

9 Platinenherstellung **55**

10 Aufbau **59**
10.1 Kabel 59
10.2 Platinen 60
10.3 Funkmodul-Adapter 61
10.4 Peripherie 61
10.4.1 Transmitter 61
10.4.2 Zündbox 63
10.4.3 Koffer 67

11 Aufspielen des Bootloaders **69**

12 Tipps und Tricks **71**
12.1 LCD an 3,3 V 71
12.2 Antennenbau 71
12.3 Kompilieren der Firmware 74

Abbildungsverzeichnis

1.1 Systemübersicht *El Fueradoro* 16
1.2 Transmitter 17
1.3 Zündbox von oben, eingebettet in Koffer 18
1.4 LCD während der Show 19
1.5 Schalter und serielle Schnittstelle an der Zündbox 20

2.1 Eintrag des USB-RS232-Adapters im Gerätemanager 22
2.2 Einstellungen für `Puttytel` 23

3.1 Startbildschirm des Konfigurationsprogramms 27
3.2 Beispiel einer ferngesteuerten ID-Zuweisung 29
3.3 Systemübersicht 30

5.1 Schaltplan des Transmitters 38
5.2 Schaltplan der Zündbox 39
5.3 Layout des Transmitters (keine Originalgröße!) 40
5.4 Layout der Zündbox (keine Originalgröße!) 41

6.1 Pinbelegung des Mikrocontrollers bei Transmitter und Zündbox 43

9.1 Ober- und Unterseite des Transmitters für Toner-Transfer-Verfahren/Belichtung 56
9.2 Ober- und Unterseite der Zündbox für Toner-Transfer-Verfahren/Belichtung . . . 57
9.3 Adapterplatine für Funkmodule (8 Stück) für Toner-Transfer-Verfahren/Belichtung 58

10.1 Abmessungen (in mm) des LCD 62
10.2 Bohrschablone für Zündboxoberseite 65

11.1 Pinbelegung des ISP-Platinensteckers: Ansicht von oben, Aussparung an Pin 5. Quelle: `mikrocontroller.net` 69

12.1 Antenne ohne Schrumpfschlauch (links) und komplett fertig (rechts) 72
12.2 VSWR-Messung der gefertigten Antenne am Netzwerkanalysator 74

Tabellenverzeichnis

1.1 Farben und Funktionen der Status-LEDs 19

2.1 Konfiguration der seriellen Schnittstelle 22

3.1 Kommandos zur Konfiguration über die serielle Schnittstelle 26
3.2 Umrechnung Hexadezimal-, Dezimal- und Binärwerte 31
3.3 Struktur des `RFM69CW`-Befehls 31

7.1 Quellcodedateien und ihre Funktionen 45

8.1 Materialliste für den Transmitter 50
8.2 Materialliste für die Zündbox 54

10.1 Übersicht über benötigte Kabelverbindungen 60

Teil I

Benutzerhandbuch

1 Das System

El Fueradoro ist eine Eigenentwicklung zur automatisierten Zündung von Feuerwerkschoreographien per Computer und Funk, der Name *El Fueradoro* leitet sich aus den spanischen Wörtern fuego, radio und oro her und bedeutet frei übersetzt daher so etwas wie „Goldenes Funkfeuer“. Sie wurde zur Verwendung mit der frei verfügbaren Software *Pyro Ignition Control* von Yannic Wilkening (Version 1.4.5) geplant.

Es besteht aus einer mit dem PC verbundene Transmitterbox, welche die von *Pyro Ignition Control* generierten Zündbefehle per Funk an die Zündboxen weitergibt und einer oder mehreren Zündboxen mit je 16 Kanälen zur elektrischen Zündung. Der klassische Aufbau des Systems ist schematisch in Abbildung 1.1 gezeigt. Die reine Transmitterbox wird nicht zwingend benötigt; es kann alternativ auch eine Zündbox per Kabel mit dem PC verbunden werden und neben ihrer originären Aufgabe auch die Transmitteraufgaben bewältigen.

Um möglichst große Flexibilität bei der Gestaltung eines Feuerwerks mit *El Fueradoro* zu gewährleisten, was z.B. die Anzahl der verwendeten Zündboxen oder auch die Tatsache angeht, dass man identische Effekte gerne zeitgleich an zwei weiter voneinander entfernten Orten zünden möchte, können den Boxen per Software zwei Kennnummern (Slave-ID und Unique-ID) zugewiesen werden. Genaueres dazu findet sich im Abschnitt 3.

Transmitterbox und Zündbox sind in Abbildung 1.2 bzw. 1.3 mit ihren wesentlichen Bestandteilen gezeigt. Sie werden im weiteren Text unter dem Begriff „Devices“ zusammengefasst, falls sich Aussagen auf beide Teile beziehen, und in den folgenden Kapiteln näher beschrieben.

1.1 Aufbau

Detailliertere Infos zum Aufbau erhalten interessierte Leser im nächsten Teil dieses Dokuments. An dieser Stelle soll nur kurz auf die Funktionalität eingegangen werden. Herzstück aller Devices ist der 8-bit-Mikrocontroller `ATmega328P` von Atmel, welcher die verschiedenen Peripheriebausteine kontrolliert. In jedem Device befinden sich ein Funkmodul `RFM69CW` von HopeRF, welches die drahtlose Kommunikation zwischen den einzelnen Devices übernimmt, ein RS232-Treiberbaustein `MAX202` mit an die Gehäuseaußenseite geführter Sub-D-Buchse zur Kommunikation zwischen Mikrocontroller und PC sowie vier Status-LEDs.

Die Transmitterbox ist mit einem LC-Display mit 20 Spalten und 4 Zeilen ausgestattet.

In den Zündboxen befinden sich zwei vom Mikrocontroller gesteuerte kaskadierte Schieberegister

Abbildung 1.1: Systemübersicht *El Fueradoro*

`74HC595`, welche mit ihren insgesamt 16 Ausgängen 16 Feldeffekttransistoren vom Typ `IRF3708` für die Zündung ansteuern, außerdem ist auf der Platine mit dem `MC33063` ein Hochsetzsteller zur Erzeugung einer höheren Zündspannung integriert sowie ein Schlüsselschalter zum Scharfschalten der Boxen. Parallel zur Drain-Source-Strecke jedes Feldeffekttransistors ist eine LED mit Vorwiderstand geschaltet, um zu signalisieren, an welchen Kanälen Zünder angeschlossen sind.

1.2 Transmitterbox und Zündboxen

Die in Abbildung 1.2 gezeigte Transmitterbox dient dazu, die Zündbefehle des PCs per Funk an die über das Gelände verteilt stehenden Zündboxen weiterzuleiten sowie das komplette Funksystem zu überwachen. Hierzu ist im Inneren der Transmitterbox sowie der Zündboxen ein Funkmodul für das – im Rahmen der Vorschriften der jeweiligen nationalen Aufsichtsbehörde, in Deutschland der Bundesnetzagentur – frei nutzbare Frequenzband um 868 MHz verbaut. Die Datenübertragung vom PC geschieht über eine serielle Schnittstelle.

Die Zündboxen, in Abbildung 1.3 dargestellt, empfangen die Anweisungen und zünden die einzelnen Kanäle. Jede Zündbox verfügt über 16 einzeln ansteuerbare Zündkanäle. Jeweils eine rote und eine schwarze Klemme bilden einen Zündkanal, die Nummerierung beginnt links unten mit Kanal 1 und endet rechts oben mit Kanal 16. Alle roten Klemmen sind nach dem Einschalten unmittelbar mit einer Gleichspannung von 22,5 V verbunden, die schwarzen Klemmen mit dem Drain-Anschluss eines von 16 n-Kanal-MOSFETs. Parallel zur üblicherweise sperrenden Drain-Source-Strecke jedes MOSFETs ist eine grüne LED mit Vorwiderstand geschaltet, welcher den

Abbildung 1.2: Transmitter

Strom über den LED-Zweig auf unter 5 mA begrenzt, so dass ein Zünden über die LED ausgeschlossen ist. Sobald die zugehörige schwarze Klemme mit einer roten Klemme – die Zuordnung kann beliebig erfolgen, obwohl zwecks Übersichtlichkeit natürlich ratsam ist, die nebeneinander liegenden Klemmen zu nutzen – verbunden ist, also ein geschlossener Strompfad von 22,5 V zur Schaltungsmasse besteht, wird dies durch Leuchten der zur schwarzen Klemme gehörigen LED signalisiert.

1.3 Energieversorgung

Die in unmittelbarer Nähe des zu steuernden PCs platzierte Transmitterbox bezieht ihre nötige elektrische Energie aus dem USB-Port eines PCs. Das fest mit der Transmitterbox verbundene USB-Kabel dient ausschließlich diesem Zweck. Sie verfügt über keinen Ein/Aus-Schalter, sondern ist eingeschaltet, solange sie mit dem USB-Port verbunden und der PC eingeschaltet ist. Da der USB-Port „angezapft“ wird, ohne in der sonst üblichen Weise mit dem Controller zu kommunizieren, empfiehlt sich, die USB-Verbindung erst nach komplett abgeschlossenem Bootvorgang herzustellen und vor dem Herunterfahren des Rechners wieder zu trennen.

Aufgrund der anzunehmenden Platzierung der Zündboxen im freien Feld, abseits von Steckdosen und anderen Energiequellen werden sie über eine Batterie versorgt. **Um ordnungsgemäße Funktionalität zu garantieren und Schäden an der Schaltung zu vermeiden, muss die Batteriespannung zwischen 8 V und 15 V liegen!** Empfohlen wird die Verwendung

Abbildung 1.3: Zündbox von oben, eingebettet in Koffer

Farbe	Funktion
orange	Funkmodul empfängt
grün	Funkmodul sendet
gelb	Daten kommen über serielle Schnittstelle an
rot	Zündbox ist scharf geschaltet

Tabelle 1.1: Farben und Funktionen der Status-LEDs

Abbildung 1.4: LCD während der Show

eines Blei-Vlies-Akkus mit Nennspannung 12 V, wie er auch in Abbildung 1.3 über der Zündbox zu erahnen ist.

1.4 Die Status-LEDs

Alle Devices verfügen über vier Status-LEDs. Bei den Transmitterboxen liegen sie direkt neben dem Kabel für die Energieversorgung auf der Seite, bei den Zündboxen auf der Oberseite. Diese sind mit ihrer Bedeutung in Tabelle 1.1 aufgeführt und leuchten, wenn das jeweilige Device die mit der LED verknüpfte Tätigkeit ausführt. Das Blinken der grünen LED während des Initialisierungsvorgangs jedes Devices ist von der sonstigen Funktion unabhängig.

Bei einem Zündvorgang leuchten für die Dauer der Zündung (11 ms) alle vier LEDs.

1.5 Das LCD der Transmitterbox

Aktuelle Statusanzeigen werden bei der Transmitterbox auf dem LC-Display ausgegeben. Ein Beispiel zeigt Abbildung 1.4. In Zeile 1 wird hinter der Abkürzung „Tx“ (Transmitted) der letzte

Abbildung 1.5: Schalter und serielle Schnittstelle an der Zündbox

gesendete Befehl (Zündbefehl oder Identifizierungsaufforderung) angezeigt, in Zeile 2 – im Bild nicht zu sehen – ggf. hinter der Abkürzung „Rx“ (Received) die letzte empfangene Rückmeldung (angeforderte Parameter).

In den Zeilen 3 und 4 werden die letzten sechs gesendeten Kommandos aufgelistet. Zwei zweistellige Zahlen getrennt durch ein Flammensymbol stehen dabei für einen Zündbefehl. Die erste Zahl gibt die Slave-ID, die zweite den zu zündenden Kanal an. Für den Fall, dass eine Aufforderung zur Identifikation gesendet wurde, erscheint „IDENT“, für eine Aufforderung zur Temperaturmessung „TEMP“. Das „x“ steht immer vor dem bis dato letzten Befehl, springt also mit jedem neuen Befehl eine Stelle weiter.

Alle Zeilen werden nach einer bestimmten Zeit automatisch gelöscht.

1.6 Schalter an den Zündboxen

Die Zündboxen verfügen, wie in Abbildung 1.5 zu erkennen, über zwei Schalter, einen schwarzen Wippschalter zum Ein- und Ausschalten der Energieversorgung sowie einen Schlüsselschalter, um die Zündbox „scharf“ zu schalten.

Die Scharfschaltung durch den Schlüsselschalter geschieht auf die Weise, dass durch eine Zustandsabfrage vor der Zündung letztere nur ausgeführt wird, wenn das Schloss auf den grünen Punkt am Gehäuse zeigt. Befindet sich der Schlüssel in waagrechter Stellung und zeigt auf den roten Punkt, so ist der Schalter geöffnet und Zündbefehle werden von der Box ignoriert. Für das Scharfschalten der Box ist der Anwender selbst verantwortlich.

Ob Boxen scharf geschaltet sind, ist an der Box – wie in Abschnitt 1.4 ausgeführt – durch das Dauerleuchten der roten Status-LED erkennbar, kann aber auch durch eine Identifizierungsabfrage ausgelesen werden (siehe Abschnitt 3.1.2).

2 Vorbereitung des PCs

Zur Kommunikation mit einem Computer verfügen alle Devices über eine serielle Schnittstelle. *Pyro Ignition Control* sollte ohnehin auf dem Rechner installiert sein, für die serielle Kommunikation gibt es für Windows zudem zahlreiche Terminalprogramme.

2.1 Installation eines USB-RS232-Adapters

Zunächst steht man allerdings in der Regel vor dem Problem, dass zwar die Devices eine serielle Schnittstelle besitzen, moderne Rechner aber nicht mehr mit dem früher standardmäßig verbauten 9-poligen Sub-D-Stecker der RS232-Schnittstelle ausgestattet sind. Diese wurden seit dem Ende der 1990er-Jahre von den USB-Schnittstellen verdrängt. Sollte wider Erwarten am einzusetzenden Rechner ein derartiger Anschluss vorhanden sein, können die nächsten Absätze übersprungen und der COM-Port direkt im Gerätemanager anhand von Tabelle 2.1 konfiguriert werden. Wer nur über USB-Ports verfügt, lese unmittelbar weiter.

Weil in vielen Bereichen noch immer auf RS232 zurückgegriffen wird, existieren Adapterkabel wie in Abbildung 1.2 mit USB-Anschluss für den Rechner und einem 9-poligen RS232-Stecker für den Anschluss an der Peripherie, also die Devices von *El Fueradoro*. In diesen Adapterkabeln ist ein Chip verbaut, um die Signalumsetzung von USB auf RS232 und umgekehrt zu bewerkstelligen. Übliche verwendete Chips sind der CH340[1], welcher sich in vielen über eBay aus China angebotenen Modellen befindet, der Prolific PL2303[2] in verschiedenen Versionen oder – bei edleren und somit auch teureren Varianten – der uneingeschränkt zu empfehlende FTDI232, mit welchem die in den folgenden Abschnitten beschriebenen Probleme nicht auftreten sollten.

Das Plug-and-Play-Traumszenario, dass sich der Adapter bei der Verbindung des USB-Steckers mit dem Rechner automatisch korrekt installiert, tritt gerade bei den günstigen Adaptern leider nur sehr selten ein. Die in den Fußnoten verlinkten Treiber sollten, sofern die automatische Treiberinstallation von Windows versagt, ihren Dienst tun, müssen allerdings teilweise mit sanfter Gewalt installiert werden. Hat man die Installation erfolgreich absolviert, sollte bei angeschlossenem Adapterkabel ein neuer Eintrag in der Art von Abbildung 2.1 im Gerätemanager auftauchen.

Weil viele Chips dazu tendieren, sich beim Anschluss an immer wieder andere USB-Ports neu zu

[1] Treiber CH340/341: `http://wch.cn/download/list.asp?id=5`

[2] Falls die automatische Treiberinstallation unter Windows fehlschlägt, funktioniert – mit zeitweiligen Aussetzern – oft der Treiber unter: `http://www.cartft.com/support/drivers/TFT/tftdrivers/GPS/PL2303_Prolific_GPS_1013_20090319.zip`

Abbildung 2.1: Eintrag des USB-RS232-Adapters im Gerätemanager

installieren bzw. eine andere COM-Port-Nummer anzunehmen, wird empfohlen, für den USB-RS232-Adapter stets denselben USB-Steckplatz zu nutzen.

Durch Doppelklick auf den Eintrag und den Reiter Anschlusseinstellungen kann die nun vorhandene serielle Schnittstelle unter Windows konfiguriert werden. Um Kompatibilität mit *Pyro Ignition Control* zu gewährleisten, ist das Hauptfenster nach Tabelle 2.1 zu konfigurieren.

Unter der Schaltfläche erweitert kann man zudem die Puffer ausschalten, was aber nicht zwingend notwendig ist und die Funktionsweise normalerweise weder positiv noch negativ beeinflusst, sowie die Portnummer für den neu geschaffenen COM-Port einstellen.

2.2 Einrichtung des Terminalprogramms

Hardware- und treiberseitig steht einer erfolgreichen Kommunikation von Rechner und Devices nun nichts mehr im Wege, für eine komfortable Unterhaltung außerhalb von *Pyro Ignition Control* fehlt aber noch die entsprechende Software. Empfohlen wird die Verwendung des kostenlosen

Bits pro Sekunde	9600
Datenbits	8
Parität	keine
Stoppbits	1
Flusssteuerung	Hardware

Tabelle 2.1: Konfiguration der seriellen Schnittstelle

Abbildung 2.2: Einstellungen für `Puttytel`

Programms *Puttytel*[3], mit welchem auch die im Rahmen dieser Anleitung gezeigten Beispiele durchgeführt werden. Es besteht nur aus einer einzigen ausführbaren Datei.

`Puttytel` kann per Doppelklick gestartet werden, woraufhin man zu einem Startbildschirm gelangt. Man wählt in der linken Spalte unten links „Serial“ und stellt die Parameter – analog zur Konfiguration des COM-Ports nach Tabelle 2.1 – wie in Abbildung 2.2 ein, ehe man die serielle Verbindung per Klick auf „Open“ startet. Vor dem Start der Verbindung sollte man das verbundene Device mit Strom versorgen.

Um bei `Puttytel` nicht immer all diese Einstellungen per Hand vornehmen zu müssen, bietet sich an, unter Windows eine Verknüpfung auf `puttytel.exe` zu erstellen und in den Verknüpfungseigenschaften als Ziel anzugeben:

```
"c:\programme\puttytel\puttytel.exe" -serial com24 -sercfg 9600,8,1,n,R
```

Die ohne Leerzeichen auf „com“ folgende Zahl ist natürlich entsprechend dem verwendeten seriellen Anschluss (COM5, COM37, ...) anzupassen.

2.3 Einrichtung von *Pyro Ignition Control*

Um eine reibungslose Kommunikation zwischen *Pyro Ignition Control* und *El Fueradoro* sicherzustellen, muss in *Pyro Ignition Control* als wesentliche Einstellung unter dem Menüpunkt „Einstellungen → Optionen“ im Reiter „Output“ der richtige COM-Port eingestellt werden.

[3] Herunterzuladen unter: `http://www.chiark.greenend.org.uk/~sgtatham/putty/download.html`

Über „Einstellungen → Connect“ wird die serielle Verbindung aufgebaut und in der untersten Leiste anzeigt, ob der Verbindungsaufbau erfolgreich war.

Zudem sollte im Reiter „Allgemein“ die globale Verzögerung erfahrungsgemäß auf etwa 0,07 s eingestellt werden. Dies ist die Zeit, die aufgrund von Datenübertragungen und Rechenvorgängen zwischen dem Beginn des Sendens des Befehls vom PC zum Transmitter und dem Zünden des Kanals an der Zündbox vergehen.

Als minimale Zeitdauer zwischen zwei Zündungen sollte 100 ms nicht unterschritten werden, das Scharfschalten vor Beginn der Show ist ebenfalls nicht zu vergessen!

Anmerkung: Eine serielle Verbindung kann immer nur ENTWEDER zwischen `Puttytel` ODER *Pyro Ignition Control* ODER dem Firware-Updater und dem Device bestehen. Man muss also immer die bestehende Verbindung trennen, bevor man mit einem anderen Programm eine neue aufbauen kann.

3 Kommunikation zwischen PC und Devices

In diesem Abschnitt wird die Systemüberwachung bzw. -konfiguration über die serielle Schnittstelle mittels `Puttytel` oder die *El Fueradoro*-GUI behandelt. Auf die Kommunikation zwischen *Pyro Ignition Control* und der Transmitterbox wird an dieser Stelle nicht eingegangen, da hier – wenn alle Einstellungen wie in Abschnitt 2.3 erläutert getroffen wurden – alles quasi-automatisch und ohne Zutun des Benutzers stattfindet.

3.1 Puttytel

3.1.1 Befehlsübersicht

Hat man mittels `Puttytel` eine Verbindung zwischen einem Device und dem PC aufbauen können, sieht man vor sich zunächst nur einen schwarzen Bildschirm. Um nun mit dem Device kommunizieren zu können, existieren einige Befehle gemäß Tabelle 3.1.

Diese können, sofern `Puttytel` die aktive Anwendung ist, direkt über die PC-Tastatur eingegeben werden und sollten zur unmittelbaren Ausführung mit Druck auf die Taste *ENTER* abgeschlossen werden. Sobald das erste Zeichen eingegeben wurde, leuchtet die gelbe Status-LED am Device. Unbekannte Befehle werden ignoriert, sämtliche Buchstaben als Kleinbuchstaben interpretiert. Korrekturen sind unter Verwendung der *BACKSPACE*-Taste möglich.

Aufgrund der eingebauten Timeout-Funktion, welche ein Hängenbleiben des Programms während einer Show verhindern soll, bricht die Firmware die Eingabe ab, wenn zwischen der Eingabe der einzelnen Buchstaben mehr als 3 s vergehen. Lässt man diese Zeit verstreichen, wird automatisch ein Drücken der *ENTER*-Taste übermittelt, die Befehlseingabe also abgeschlossen und das Device ist unmittelbar bereit, einen neuen Befehl aufzunehmen. Wird also nach Eingabe eines gültigen Befehls die *ENTER*-Taste nicht gedrückt, wird der Befehl durch den Timeout dennoch ausgelöst. Möchte man dies vermeiden, sollte man den Befehl vor Ausführung durch Eingabe weiterer Zeichen ungültig machen oder durch Entfernen aller Zeichen mittels *BACKSPACE* löschen.

Von den in Tabelle 3.1 aufgeführten Befehlen funktionieren lediglich zwei nicht bei allen Devices. „orders“ setzt voraus, dass das angeschlossene Device zuvor – über das durch „conf“ zu erreichende Menü – als Transmitterbox konfiguriert wurde bzw. im Fall von „zero“ nicht als Transmitterbox.

Befehl	Wirkung
conf	Startet das Konfigurationsprogramm zur lokalen Zuweisung von Slave- und Unique-ID
remote	Startet das Konfigurationsprogramm zur ferngesteuerten Zuweisung von Slave- und Unique-ID
list	Zeigt die Systemübersicht (Zuweisung Unique- und Slave-ID, Batteriespannung jeder Box, Scharfschaltungsstatus Temperatur, RSSI, Anzahl Boxen je Slave-ID)
send	Startet das Menü zur manuellen Eingabe einer Anweisung ans Funkmodul (Zündbefehl, Identifizierungsaufforderung oder Temperaturmessung)
fire	Führt zu einer Eingabemaske, in die Slave-ID und Kanal für die Zündung einzugeben sind
ident	Sendet eine Identifizierungsaufforderung an alle anderen Devices
temp	Gibt über die serielle Schnittstelle die Temperatur aus und fordert alle anderen Devices ebenfalls zur Temperaturmessung auf. Zum Auslesen der neu gemessenen Temperaturen muss dann eine Identifizierungsanfrage geschickt werden
rfm	Erlaubt unmittelbaren Zugriff auf das Funkmodul durch Eingabe einer 16-Bit-Hexadezimalzahl, um Registerwerte auszulesen oder neu zu setzen
orders	Gibt letztes gesendetes und empfangenes Pattern auf LCD aus
zero	Markiert alle Kanäle der verbundenen Zündbox als noch nicht abgefeuert
cls	Löscht den Terminal-Bildschirm
kill	Löst einen Neustart des Device aus

Tabelle 3.1: Kommandos zur Konfiguration über die serielle Schnittstelle

3.1.2 Die Konfiguration

Lokal

Mit „conf“ gelangt man ins Konfigurationsmenü für die lokale Konfiguration der IDs, welches den Benutzer wie in Abbildung 3.1 gezeigt empfängt. Hier sind die beiden wichtigsten Parameter jedes Devices, die Unique-ID und die Slave-ID, aufgeführt, die der Benutzer nach eigenen Bedürfnissen vergeben kann.

Die **Unique-ID** dient der Identifikation jeder einzelnen Zündbox im Funksystem. Jeder verwendeten Zündbox muss daher, um die Funktion von *El Fueradoro* gewährleisten zu können, eine andere Unique-ID im Bereich von 01-30 (zweistellige Eingabe!) zugeteilt werden.

Die **Slave-ID** entscheidet, auf welche Zündbefehle eine Zündbox reagiert. Sollen also zwei oder mehr Boxen stets zur selben Zeit denselben Kanal zünden, kann ihnen einfach die gleiche Slave-ID zugewiesen werden. Wie oben ausgeführt muss die Zuweisung der Unique-IDs allerdings eineindeutig sein, d.h. jede Box hat eine unterschiedliche Unique-ID bzw. jede Unique-ID gehört zu genau einer Zündbox.

Abbildung 3.1: Startbildschirm des Konfigurationsprogramms

Die Null als Slave- und Unique-ID identifiziert ein Device als Transmitterbox.

Die Zuweisung von Slave- und Unique-ID vom Startbildschirm des Konfigurationsprogramms aus geschieht, indem man den Anweisungen auf dem Bildschirm folgt. U bzw. u (Groß- oder Kleinschreibung spielt hier keine Rolle) ändert die Unique-ID, S bzw. s die Slave-ID einer Zündbox. T bzw. t macht das Device zum Sender, indem Unique- und Slave-ID auf 0 gesetzt werden. Die neue Unique- bzw. Slave-ID muss stets zweistellig ohne Bestätigung durch Enter oder eine andere Taste eingegeben werden, die Konfiguration als Transmitter kann nicht über die Eingabe von Unique- und Slave-ID, sondern nur über die Eingabe `T` bzw. `t` im Hauptbildschirm erfolgen.

Möchte man also die Unique-ID 5 zuweisen, muss man zunächst „u“ und anschließend „05“ eingeben, für Slave-ID 12 dementsprechend „s“ und „12“. Ein Sonderfall tritt auf, wenn man ein zuvor als Transmitter konfiguriertes Device nun als Zündbox konfigurieren will. Es erscheint dann nach Eingabe der neuen Slave-ID (Unique-ID) automatisch die Eingabeaufforderung für die neue Unique-ID (Slave-ID), da entweder beide IDs den Wert 0 haben müssen (Transmitter) oder beide IDs einen anderen Wert als 0 (Zündbox). Die zugewiesenen IDs werden an drei Stellen im internen Speicher mit Prüfsummen hinterlegt und bleiben sowohl nach dem Ausschalten als auch nach einem Firmwareupdate erhalten.

Wie bereits angesprochen ist es möglich, auch eine Zündbox als Transmitter zu verwenden. Damit diese aber gleichzeitig noch als Zündbox fungieren kann, muss die Slave-ID größer sein als 0, da mit *Pyro Ignition Control* nur Slaves von 01-30 angesprochen werden können.

Als Transmitter (Slave-ID 0 und Unique-ID 0) konfiguriert werden können nur Devices wie in Abbildung 1.2 mit Display und ohne Klemmen für den Anschluss von Zündern. Der Grund hierfür liegt in der in Abbildung 6.1 auf Seite 43 gezeigten Pinbelegung des Controllers. Wie zu erkennen ist, werden Controlleranschlüsse beim Transmitter als Steuerung des Displays verwendet, die bei der Zündbox den Zustand des Schlüsselschalters einlesen oder die Schieberegister zur Zündung der Kanäle ansteuern. Da die Software davon ausgeht, dass ein Device mit Slave-ID und Unique-ID 0 an diesen Pins ein LC-Display besitzt, werden die Pins entsprechend konfiguriert und angesteuert. Im Fall des Schlüsselschalter-Pins kommt es daher zu einem Kurzschluss,

wenn die Box scharf geschaltet ist, was den Controller beschädigen kann, die Ansteuerung des Schieberegisters mit LCD-Befehlen führt dazu dass Zündkanäle durchschalten und Zündungen unkontrolliert ausgelöst werden, weil die Software glaubt, die Datenpins des LCD zu modulieren, in Wahrheit aber die Schieberegister ansteuert.

Die Firmware erkennt durch eine Überprüfung des Devices beim Start, ob es sich bei dem angeschlossenen Device um einen Transmitter oder eine Zündbox handelt. Bei Zündboxen ist die Option „Als Transmitter konfigurieren“ daher nicht verfügbar. Der umgekehrte Fall, eine Transmitterbox als Zündbox zu konfigurieren, ist nicht sinnvoll und führt dazu, dass auf dem LCD nur zwei komplett gefüllte Zeilen zu sehen sind. Dies führt aber zu keinerlei Schäden, sobald das Device wieder als Transmitter konfiguriert ist, wird das LCD wieder einwandfrei funktionieren.

Unabhängig davon, dass eine Zündbox nicht als Transmitter konfiguriert werden, also nicht Unique- und Slave-ID 0 besitzen darf, kann sie theoretisch trotzdem zur Steuerung und Koordinierung eines Netzes und einer Show eingesetzt werden, indem man sie über die serielle Schnittstelle mit dem PC verbindet. Bis auf die Darstellung am LCD erfüllt sie dieselben Aufgaben wie ein Transmitter und kann auch parallel noch als Zündbox fungieren. Auf entsprechenden Sicherheitsabstand zu sensibler Technik und Lebewesen ist dabei selbstverständlich zu achten!

Ferngesteuert

Über den Befehl „remote“ gelangt man ins Konfigurationsprogramm zur ferngesteuerten Vergabe von Unique- und Slave-ID. Der Programmablauf ist beispielhaft in Abbildung 3.2 gezeigt: Die Eingabe der alten und neuen IDs erfolgt analog zur Eingabe bei lokaler Konfiguration, am Ende muss die Änderung noch bestätigt werden. Werden als alte Unique- und Slave-ID die Daten der verbundenen Box eingegeben, so werden deren Kennzahlen wie bei einer lokalen Konfiguration geändert und kein weiterer Befehl gesendet.

Durch „remote“ und die Eingabe von Unique- und Slave-ID einer nicht per Kabel verbundenen Box ist es möglich, die IDs einer eingeschalteten Zündbox per Funk zu ändern. Voraussetzung ist dabei, dass die angesprochene Zündbox nicht scharf geschaltet ist – hierdurch soll ein Eingriff von außen während einer Show unterbunden werden, aus Sicherheitsgründen ist es außerdem nicht möglich, ein Device ferngesteuert als Transmitter zu konfigurieren.

Der Anwender hat selbst darauf zu achten, durch ein ferngesteuertes Update nicht einem Device eine bereits vergebene Unique-ID zuzuweisen! Falls dies dennoch geschieht, ist das weitere Vorgehen davon abhängig, ob die Devices auch die gleiche Slave-ID besitzen oder nicht. Sind die Slave-IDs nicht identisch, so kann durch einen weiteren „remote“-Befehl die Unique-ID-Zuweisung geändert werden. Bei identischen Slave-IDs funktioniert dies nicht, da stets alle Devices auf den Änderungsbefehl in gleicher Weise reagieren würden. Hier müssen daher alle Devices mit identischen IDs bis auf eines ausgeschaltet werden, dem man dann neue IDs zuweisen kann. Nun kann dann jeweils ein weiteres Device eingeschaltet und seine IDs neu gesetzt werden, bis wieder alle unterschiedliche Unique-IDs besitzen.

Abbildung 3.2: Beispiel einer ferngesteuerten ID-Zuweisung

3.1.3 Die Systemübersicht

Mit „list“ ist es möglich, sich die Systemübersicht entsprechend Abbildung 3.3 anzeigen zu lassen. Es werden zwei Tabellen ausgegeben, wobei die obere nach Unique-ID geordnet anzeigt:

1. Welche Slave-ID der Unique-ID zugewiesen ist.

2. Welche Spannung die Batterie der Box mit der entsprechenden Unique-ID liefert.

3. Ist die Box mit der jeweiligen Unique-ID scharf geschaltet: (j)a oder (n)ein.

4. Temperatur im Inneren der Box, sofern die Box über einen eingebauten Temperatursensor verfügt.

5. Wie stark ist das von der Box empfangene Signal (RSSI = Received Signal Strength Indication). Werte stehen nur bei Aufruf von „list“ an einem Device mit `RFM69` zur Verfügung.

Die untere Tabelle listet auf, wie viele Boxen mit der entsprechenden Slave-ID derzeit aktiv sind.

Zwischen den beiden Tabellen wird die Anzahl der fehlerhaften IDs aufgelistet. Dies kann entweder auf doppelte Zuweisung von Unique-IDs oder Fehler beim Auslesen der IDs (fehlerhafte Prüfsummen) zurückzuführen sein. Für normalen Betrieb sollte dieser Wert stets 0 betragen.

Der dargestellte Zustand entspricht den empfangenen Parametern nach der letzten Identifikationsaufforderung, für eine möglichst aktuelle Liste sollte also zuvor, wie im Abschnitt 3.1.4 beschrieben eine Identifikationsaufforderung gesendet werden.

Abbildung 3.3: Systemübersicht

3.1.4 Manuelles Senden

Zu Testzwecken oder um die Systemübersicht zu aktualisieren, können mittels „send“ Zündbefehle und die Aufforderung zur Identifizierung oder Temperaturmessung manuell versendet werden. Nach Eingabe von „send“ muss dies mit „f“ (=fire), „i“ (=identify) oder „t“ (=temperature) ausgewählt werden. Wählt man „i“ oder „t“ ist keine weitere Eingabe nötig, bei „f“ müssen anschließend noch Slave-ID und Kanal jeweils zweistellig eingegeben werden. Statt „send“ und den entsprechenden Buchstaben anzugeben, können auch die direkten Befehle „fire“, „ident“ und „temp“ verwendet werden.

Das Senden einer Identifikationsaufforderung hat zudem den Effekt, dass alle Kanäle in allen Zündboxen als noch nicht abgefeuert gekennzeichnet werden.

Jede andere Angabe als „f“, „i“ oder „t“ beendet den Modus ohne irgendetwas zu senden. Denselben Effekt hat die Eingabe einer Slave-ID oder Kanalnummer außerhalb der jeweils zulässigen Zahlenbereiche.

Hexadezimal	0	1	2	3	4	5	6	7
Dezimal	0	1	2	3	4	5	6	7
Binär	0000	0001	0010	0011	0100	0101	0110	0111
Hexadezimal	8	9	A	B	C	D	E	F
Dezimal	8	9	10	11	12	13	14	15
Binär	1000	1001	1010	1011	1100	1101	1110	1111

Tabelle 3.2: Umrechnung Hexadezimal-, Dezimal- und Binärwerte

3.1.5 Funkmodul-Zugriff

Die zwingend für den Betrieb von *El Fueradoro* notwendigen Zugriffe auf das Funkmodul werden von der Software automatisch getätigt, so dass diese Funktion in der Regel nicht gebraucht wird, dennoch ist es möglich das verwendete Funkmodul `RFM69CW`[1] unmittelbar über das Terminalprogramm anzusprechen, um Werte aus den Registern zu lesen oder die Register neu zu beschreiben.

Nach Eingabe von „rfm“ und Bestätigung mit *ENTER* erscheint eine Aufforderung zur Befehlseingabe. Diese hat im Hexadezimalformat als 16-Bit-Wert zu erfolgen, d.h. vierstellig mit den zulässigen Zeichen 0-9 und A-F bzw. a-f. Jedes eingegebene Zeichen symbolisiert dabei vier Bits, die Umrechnung ist in Tabelle 3.2 gezeigt.

Die Bedeutung der Eingabe für das Funkmodul ist in Tabelle 3.3 illustriert. Hierbei sollte auch klar werden, wie sich die Werte für die Zeichen 1-4 zusammensetzen. Ist ein Bit gesetzt, muss die entsprechende Zahl (8, 4, 2, 1) zum Zeichenwert addiert werden, so dass sich bei vier gesetzten Bits als Maximalwert 15 ergibt, ist nur das oberste Bit gesetzt, lautet der Wert 8, ist nur das unterste gesetzt 1, sind die beiden mittleren Bits gesetzt 6, usw.

Es ist zu erkennen, dass das erste einzugebende Zeichen sowohl das Schreiben/Lesen-Bit enthält als auch die obersten drei Bit der Registeradresse. Die acht Datenbits sind lediglich für einen Schreibbefehl relevant, bei einem Lesezugriff kann als drittes und viertes Zeichen ohne Konsequenzen ein beliebiger Hexadezimalwert im Bereich von 0x00 bis 0xFF übertragen werden.

Bit	w/$\bar{r}$	r6	r5	r4	r3	r2	r1	r0	d7	d6	d5	d4	d3	d2	d1	d0
Wert	8	4	2	1	8	4	2	1	8	4	2	1	8	4	2	1
Zeichen	Zeichen 1				Zeichen 2				Zeichen 3				Zeichen 4			

w/$\bar{r}$	Schreib- oder Lesezugriff (0 = lesen, 1 = schreiben)
r6 ... r0	Registeradresse
d7 ... d0	Zu schreibender Registerwert (beliebig falls w/$\bar{r}$ = 0)

Tabelle 3.3: Struktur des `RFM69CW`-Befehls

[1] Links zum Datenblatt: Homepage von Hersteller HopeRF oder Pollin

Auslesen der eingestellten Sendeleistung

Zur Veranschaulichung soll hier die Abfrage der aktuell eingestellten Sendeleistung und eine anschließende Änderung derselben simuliert werden: Aus dem Datenblatt, dem die Bedeutungen aller Registeradressen und ihrer acht Registerbits zu entnehmen sind, kann die Registeradresse 0x11 als diejenige identifiziert werden, in der die Informationen zur Sendeleistung hinterlegt sind. Um nun den aktuellen Wert auszulesen, gibt man im Terminalprogramm „rfm“ gefolgt von „ENTER“ ein und anschließend die Zeichenfolge „11FF“, wobei die beiden hinteren Stellen wie erwähnt keine Rolle spielen.

Das Modul antwortet nun mit einem 8-Bit-Wert, z.B. mit dem Wert 0x9A, welcher dem Binärwert 10011010 entspricht, dessen Bedeutung dem Datenblatt entnommen werden kann: Das oberste Bit signalisiert, dass die Verstärkerstufe PA0 aktiv ist, die beiden folgenden Bits sind 0, da PA1 und PA2 in der Variante `RFM69CW` nicht genutzt werden können. Die unteren fünf Bits schließlich stehen für die eingestellte Sendeleistung, wobei man vom aus den fünf Bits berechneten Wert noch 18 abziehen muss, um die Sendeleistung in dBm zu erhalten. Gesetzt sind die Bits 4, 3 und 1, was dem Wert 26 ($= 2^4 + 2^3 + 2^1$) entspricht, daraus resultiert eine eingestellte Sendeleistung von 8 dBm.

Setzen der Sendeleistung

Will man die Sendeleistung nun auf 6 dBm anpassen, muss also ein Wert von 24 für die Ausgangsleistungs-Bits gesetzt werden, dazu natürlich auch das oberste Bit für den PA0. Als Wert für die Registerbits ergibt sich damit $2^7 + 2^4 + 2^3 = 0x98$. Die Registeradresse bleibt gleich, jedoch muss dem Modul mitgeteilt werden, dass es sich um einen Schreibzugriff handelt, weshalb die erste Stelle um den Wert 8 erhöht werden muss. Um nun den neuen Wert von 6 dBm einzuschreiben, gibt man im Terminalprogramm „rfm“ gefolgt von „ENTER“ ein und anschließend die Zeichenfolge „9198“.

Als Antwort erhält man vom Modul den Registerwert von VOR dem Schreibzugriff, im Beispiel also den Wert 0x9A. Ein nochmaliges Auslesen des Registers wie im Abschnitt 3.1.5 sollte dann den eben eingeschriebenen Wert 0x98 zurückgeben.

3.2 GUI

Die GUI stellt eine graphische Umsetzung zur Implementierung der `Puttytel`-Funktionen dar.

4 Firmwareupdate

WICHTIG: Aus Sicherheitsgründen darf kein Update durchgeführt werden, solange Zünder mit dem Device verbunden sind, da sich die Devices im Falle eines Übertragungsfehlers völlig unvorhersehbar verhalten können.

Die aktuelle Firmware kann aus dem SVN-Repository

`http://el-fueradoro.googlecode.com/svn/trunk/`

heruntergeladen werden. Sie beinhaltet den kompletten C-Quellcode, das AVR-Eclipse-Projekt, die kompilierten iHex-Dateien für Firmware und Bootloader, die nötigen Software-Tools zur Übertragung zwischen PC und Mikrocontroller sowie die vorliegende Anleitung. Zum Selbstkompilieren wird der Compiler AVR-GCC benötigt.

El Fueradoro bietet die Möglichkeit, die Firmware via serielle Schnittstelle vom PC aus zu aktualisieren. Hierfür gibt es das Programm *fwupdate.exe*, welches zunächst einen Reset auslöst, um den Bootloader des Devices zu aktivieren und anschließend die im iHex-Format vorliegende Firmware überträgt.

fwupdate.exe muss über die Kommandozeile mit zwei Parametern gestartet werden, nämlich der Angabe der seriellen Schnittstelle und dem Namen der zu übertragenden Firmware-Datei. Aufgrund der Möglichkeit, zwei verschiedene Mikrocontroller und zwei verschiedene Funkmodultypen zu verwenden, existieren vier verschiedene Firmwaredateien, welche sich im gleichen Ordner wie *fwupdate.exe* befinden sollten.

Um nun die neue Firmware zu übertragen, lautet das Kommando für ein Device am *COM4* mit dem *ATmega328P* und dem Funkmodul *RFM69CW*:

fwupdate.exe /c4 /fm:Pyro_atmega328p_RFM69.hex

Die Angabe der Dateiendung *.hex* kann hierbei – ebenso wie das *.exe* hinter *fwupdate* – auch weggelassen werden, die Firmware-Datei muss jedoch zwingend auf *.hex* enden.

fwupdate.exe ist, sofern bereits eine korrekt funktionierende Firmware auf dem Device vorhanden ist, in der Lage, automatisiert zu ermitteln, welche Firmwaredatei die benötigte ist, das Kommando für ein Update über *COM4* lautet dann:

fwupdate.exe /c4 /fa

Der Updater führt nach Übertragung der Daten einen CRC-Check durch. Sollte dieser fehlschlagen, wurde die Firmware nicht korrekt übertragen. Dies kann zufällig passieren oder auf ein Hardwareproblem, welches in der Regel beim USB-RS232-Adapter liegt, zurückzuführen sein. Für den Fall eines CRC-Fehlers sollte die Firmware erneut übertragen werden. Bleibt das Update beim Punkt „COMx at 9600 baud:“ stehen, sollte die Stromversorgung des Device kurz unterbrochen und wieder aktiviert werden. Das Kabel für die serielle Verbindung bleibt währenddessen mit Device und PC verbunden.

Teil II

Dokumentation

5 Schaltpläne & Layouts

In den Abbildungen 5.1 und 5.2 sind die Schaltpläne von Transmitter und Zündbox nach Funktionseinheiten unterteilt gezeigt, in den Abbildungen 5.3 und 5.4 die Layouts.

Abbildung 5.1: Schaltplan des Transmitters

Abbildung 5.2: Schaltplan der Zündbox

Abbildung 5.3: Layout des Transmitters (keine Originalgröße!)

Abbildung 5.4: Layout der Zündbox (keine Originalgröße!)

6 Pinbelegung

Die Pinbelegung des `ATmega328p` in den verschiedenen Devices ist in Abbildung 6.1 gezeigt.

Besondere Bedeutung kommt dem Pin rechts oben (LCD-RW bzw. V_BATT) zu, da er beim Starten der Firmware als Eingang geschaltet wird und aufgrund der dort anliegenden Spannung die sicherheitsrelevante Erkennung, um welchen Devicetyp es sich handelt, vorgenommen wird. Aufgrund eines internen Pullup-Widerstands im LCD werden bei angeschlossenem Transmitter stets 3,3 V an diesem Pin anliegen, bei Zündboxen bewegt sich die heruntergeteilte Spannung der Versorgungsbatterie im Bereich unterhalb von 1,1 V.

Abbildung 6.1: Pinbelegung des Mikrocontrollers bei Transmitter und Zündbox

7 Software

Die Steuerungssoftware für *El Fueradoro* ist unter Zuhilfename des `AVR-GCC` in der Programmiersprache `C` geschrieben. Sie umfasst insgesamt 15 Sourcefiles (.c) mit zugehörigen Headerdateien (.h), eine Headerdatei zur Generierung von Registeradressen „portmakros.h“ sowie eine globale Headerdatei „global.h“, in welcher alle anderen erfasst sind.

Die Quellcodedateien und ihre Funktionen sind in Tabelle 7.1 aufgelistet.

Dateiname	**Aufgabe(n)**
pyro.c	Hauptprogramm, Interruptroutinen und anlagenspezifische Funktionen (Schalterinitialisierung, spezielle LCD-Symbole, ...)
1wire.c	Steuerung des Temperatursensors `DS18B20`
adc.c	Erfassung der Versorgungsspannung und des Devicetyps mittels Analog-Digital-Converter
addresses.c	Unique- und Slave-ID aus dem Speicher holen, speichern, überprüfen
crcchk.c	Überprüfen der Korrektheit empfangener Zeichenketten
dht.c	Steuerung der Temperatursensoren `DHT22` oder `DHT11` (obsolet)
eeprom.c	Direkter Zugriff auf den EEPROM des Controllers
lcd.c	Steuerung des LCD
leds.c	Kontrolle der vier Status-LEDs
rfm12.c	Funktionen für das Funkmodul `RFM12` (obsolet)
rfm69.c	Funktionen für das Funkmodul `RFM69CW`
shiftregister.c	Schieberegister-Initialisierung und -Datenübertragung
terminal.c	„GUI“ zur Benutzerinteraktion via Terminalprogramm
timer.c	Funktionen zur Timer-Steuerung
uart.c	Kommunikation über serielle Schnittstelle

Tabelle 7.1: Quellcodedateien und ihre Funktionen

Teil III

Aufbauanleitung

8 Materialliste

In den Tabellen 8.1 und 8.2 sind die benötigten Teile für den Aufbau eines Transmitters und einer Zündbox zusammen mit möglichen Bezugsquellen und Preisen (Stand Januar 2015) aufgelistet.

Materialliste Transmitter

Bauteil	Anz.	Beschreibung	Händler	Artikelnr.	Einzel	Ges.	Bemerkung
			Kondensatoren				
C11, C13	2	Elektrolyt-kondensator, 100uF	Reichelt	RAD 105 100/35	0,04 €	0,08 €	
C14	1	Elektrolyt-kondensator, 10uF	Reichelt	RAD 10/100	0,04 €	0,04 €	
C1…7, C10, C12, C15, C16	11	Keramik-kondensator, 100nF	Reichelt	X7R-2,5 100N	0,04 €	0,44 €	
C8, C9	2	Keramik-kondensator, 15pF	Reichelt	KERKO 15P	0,05 €	0,10 €	
			Integrierte Schaltungen				
IC2	1	ATMEGA 328P	Reichelt	ATMEGA 328P-PU	2,65 €	2,65 €	
IC1	1	MAX202	Reichelt	MAX 202 ECPE	1,40 €	1,40 €	
U$4	1	RFM69CW	Pollin	810 303	4,60 €	4,60 €	
U$2	1	DS18B20	Reichelt	DS 18B20	1,60 €	1,60 €	
IC8	1	LM3940-3,3	Reichelt	LM 3940 IT3,3	1,10 €	1,10 €	
			Display				
	1	LCD, 20x4 Zeichen, HD44780-komp.	eBay	400448319287	4,80 €	4,80 €	
			LEDs				

LED_O	1	LED, 5mm, orange	Reichelt	LED 5MM R OR	0,12 €	0,12 €	
LED_R	1	LED, 5mm, rot	Reichelt	LED 5MM RT	0,06 €	0,06 €	
LED_Y	1	LED, 5mm, gelb	Reichelt	LED 5MM GE	0,06 €	0,06 €	
LED_G	1	LED, 5mm, grün	Reichelt	LED 5MM GN	0,06 €	0,06 €	
				Quarz			
Q1	1	Standardquarz, Grundton, 9,8304 Mhz	Reichelt	9,8304-HC49U-S	0,15 €	0,15 €	
				Widerstände			
R13	1	6k8	Reichelt	1/4W 6,8K	0,08 €	0,08 €	
R12, R14	2	3k3	Reichelt	METALL 3,90K	0,08 €	0,16 €	
R1, R2, R3, R7	4	10k	Reichelt	METALL 10,0K	0,08 €	0,33 €	
R4…6, R8…11	7	1k0	Reichelt	METALL 1,00K	0,08 €	0,57 €	10 billiger!
				Mechanische Bauteile			
X1	1	Sub-D-Buchse, 9-pol	Reichelt	D-SUB BU 09US	0,23 €	0,23 €	
SV2	1	Wannenstecker, 10-pol	Reichelt	WSL 10G	0,08 €	0,08 €	
	1	Box	Reichelt	GEH KS 50	2,65 €	2,65 €	
	1	USB-Kabel als Stromkabel	Reichelt	AK 670/2-1,0	0,70 €	0,70 €	
				HF-Komponenten			
	1	SMA-Kabel Funkmodul-Gehäuse	eBay	371168735628	1,60 €	1,60 €	
	1	Antenne 868 MHz	eBay	331449006319	3,65 €	3,65 €	
	1	SMA-Platinenbuchse, 1,6mm	eBay	220952712009	1,10 €	1,10 €	
						28,42 €	

Tabelle 8.1: Materialliste für den Transmitter

Materialliste Zündbox

Bauteil	Anz.	Beschreibung	Händler	Artikelnr.	Einzel	Ges.	Bemerkung
			Kondensatoren				
C13, C27, C19	3	Elektrolyt-kondensator, 100uF	Reichelt	RAD 105 100/35	0,04 €	0,12 €	
C14	1	Elektrolyt-kondensator, 10uF	Reichelt	RAD 10/100	0,04 €	0,04 €	
C21, C22	2	Elektrolyt-kondensator, 4700uF 35V	Reichelt	RAD 4.700/35	0,45 €	0,90 €	
C1…7, C15, C16, C20, C23…26	14	Keramik-kondensator, 100nF	Reichelt	X7R-2,5 100N	0,04 €	0,56 €	
C8, C9	2	Keramik-kondensator, 15pF	Reichelt	KERKO 15P	0,05 €	0,10 €	
C18	1	Keramik-kondensator, 220pF	Reichelt	KERKO 220P	0,05 €	0,05 €	
			Dioden				
D3	1	1N4002	Reichelt	1N 4002	0,02 €	0,02 €	
D2	1	1N5819	Reichelt	1N 5819	0,06 €	0,06 €	
			Integrierte Schaltungen				
IC2	1	ATMEGA 328P	Reichelt	ATMEGA 328P-PU	2,65 €	2,65 €	
IC4	1	MC33063	Reichelt	MC 33063 AP1	0,51 €	0,51 €	
IC1	1	MAX202	Reichelt	MAX 202 ECPE	1,40 €	1,40 €	
IC6, IC7	2	74HC595	Reichelt	74HC 595	0,36 €	0,72 €	
IC8	1	LM1086	Reichelt	LM 1086 IT3,3	1,25 €	1,25 €	
U$4	1	RFM69CW	Pollin	810 303	4,60 €	4,60 €	
U$2	1	DS18B20	Reichelt	DS 18B20	1,60 €	1,60 €	
			Induktivität				
L1	1	68 uH, stehend	Reichelt	L-07HCP 68μ	0,30 €	0,30 €	
			LEDs				
LED_O	1	LED, 5mm, orange	Reichelt	LED 5MM R OR	0,12 €	0,12 €	

LED_R	1	LED, 5mm, rot	Reichelt	LED 5MM RT	0,06 €	0,06 €	
LED_Y	1	LED, 5mm, gelb	Reichelt	LED 5MM GE	0,06 €	0,06 €	
LED_G	1	LED, 5mm, grün	Reichelt	LED 5MM GN	0,06 €	0,06 €	
LED1...16	16	LED, 3mm, grün	Reichelt	LED 3MM GN	0,06 €	0,96 €	
				Quarz			
Q1	1	Standardquarz, Grundton, 9,8304 Mhz	Reichelt	9,8304-HC49U-S	0,15 €	0,15 €	
				MOSFETs			
Q2...Q17	16	IRF3708	AliExpress	Link	0,29 €	4,64 €	50er-Pack für 14,32 €
				Widerstände			
R16	1	180	Reichelt	METALL 180	0,08 €	0,08 €	
R17	1	0R22	eBay	221583734560	1,00 €	1,00 €	100er-Pack...
R1...3, R7, R19, R20, R37	7	10k	Reichelt	METALL 10,0K	0,08 €	0,57 €	10 billiger als 7!
R18	1	150k	Reichelt	METALL 150K	0,08 €	0,08 €	
R4...6, R8...11	7	1k	Reichelt	METALL 1,00K	0,08 €	0,57 €	10 billiger als 7!
R_Z	1	2R2-11W	Reichelt	11W VERT. 2,2	0,60 €	0,60 €	
R12, R14, R38	1	3k3	Reichelt	METALL 3,30K	0,08 €	0,08 €	
R15	1	56k	Reichelt	METALL 56K	0,08 €	0,08 €	
R21...36	18	6k8	Reichelt	1/4W 6,8K	0,03 €	0,59 €	
RN1, RN2	2	Network, 9Pin, 10k	Reichelt	SIL 9-8 10K	0,11 €	0,22 €	
				HF-Komponenten			
	1	SMA-Kabel Funkmodul-Gehäuse	eBay	371168735628	1,60 €	1,60 €	
	1	SMA-Kabel Gehäuse-Antenne	eBay	361188461828	2,20 €	2,20 €	

	1	Antenne 868 MHz	eBay	331449006319	3,65 €	3,65 €	
	1	SMA-Platinenbuchse, 1,6mm	eBay	220952712009	1,10 €	1,10 €	
				Mechanische Bauteile			
X1	1	Sub-D-Buchse, 9-pol	Reichelt	D-SUB BU 09US	0,23 €	0,23 €	
SV2	1	Wannenstecker, 10-pol	Reichelt	WSL 10G	0,08 €	0,08 €	
	1	Wippschalter	Pollin	420 697	0,35 €	0,35 €	
	1	Miniatur-Schlüsselschalter	Pollin	420 664	0,75 €	0,75 €	
	1	Kunststoffgehäuse 021-002-084	Pollin	460 001	7,95 €	7,95 €	
	8	Lautsprecher-klemmen	Reichelt	PT 932	0,29 €	2,32 €	
	1	Koffer	Amazon	Bilora 545	19,89 €	19,89 €	
	1	Akku	Reichelt	WP 1,2-12	7,80 €	7,80 €	
	4	Schrauben M3x6	Reichelt	SZK M3X6-200	0,01 €	0,03 €	200er-Pack für 1,70€
	16	Schrauben M3x10	Reichelt	SKL M3X10-50	0,02 €	0,34 €	50er-Pack für 1,05€
	16	Muttern M3	Reichelt	SK-E M3-100	0,02 €	0,35 €	100er-Pack für 2,20€
	2	Akku-Flachstecker	Reichelt	FSH-M1 4,75	0,14 €	0,28 €	
	1	Stiftleiste	Reichelt	SL 1X36G 2,54	0,15 €	0,15 €	Benötigt werden 13
	1	Buchsenleiste mit 6 Plätzen	Reichelt	MPE 094-1-006	0,25 €	0,25 €	
	1	Buchsenleiste mit 7 Plätzen	Reichelt	MPE 094-1-007	0,31 €	0,31 €	
	1	Deans-T-Plugs-Paar	Pollin	820 129	0,59 €	0,59 €	5er-Pack für 2,95€
				Kabel			
	1	Flachbandkabel für LED+Schlüsselsch.	Reichelt	AWG 28-10F 3M	1,65 €	1,65 €	

1	Litzen-Sortiment, 0,5 mm^2, 5x 5 m	Pollin	800 024	6,25 €	6,25 €
1	Schrumpf-schläuche 1,6mm	Reichelt	SDH 1,6 SW	0,25 €	0,25 €
1	Schrumpf-schläuche 3,2mm	Reichelt	SDH 3,2 SW	0,26 €	0,26 €
				83,55 €	

Tabelle 8.2: Materialliste für die Zündbox

9 Platinenherstellung

Wer zur Platinenherstellung nicht auf die Dienste eines PCB-Herstellers zurückgreifen will, findet in Abbildung 9.1 die Platine des Transmitters als Druckvorlage für den Tonertransfer[1] bzw. als Belichtungsvorlage, in Abbildung 9.2 die der Zündbox und in Abbildung 9.3 die Adapterplatine zum Auflöten des Funkmoduls.

Die Oberseite ist hierbei jeweils schon gespiegelt, die Ausdrücke können für einen Tonertransfer also einfach ausgedruckt, im Zwischenraum gefaltet werden, wobei auf möglichst exakte Deckung zu achten ist.

[1] Eine Einführung zum Ätzen mit dieser Methode gibt es HIER

Abbildung 9.1: Ober- und Unterseite des Transmitters für Toner-Transfer-Verfahren/Belichtung

Abbildung 9.2: Ober- und Unterseite der Zündbox für Toner-Transfer-Verfahren/Belichtung

Abbildung 9.3: Adapterplatine für Funkmodule (8 Stück) für Toner-Transfer-Verfahren/Belichtung

10 Aufbau

Der Aufbau umfasst alle Schritte von der blanken Platine hin zum fertigen Gehäuse/Koffer. Hierfür sind verschiedene handwerkliche Tätigkeiten, vor allem das Elektroniklöten[1], aber auch das Bohren, Schneiden, Kleben und evtl. Trennen mittels Trennscheibe und Feilen nötig.

Im eigenen Interesse ist darauf zu achten, diese Arbeiten sorgfältig und unter Einhaltung der gängigen Sicherheitsregeln durchzuführen. **Beim Bohren und Trennen Schutzbrille tragen!** Beim Löten ist auf richtige Orientierung aktiver Bauteile (Dioden, Elektrolytkondensatoren, Temperatursensor, Integrierte Schaltungen) sowie Anschlusskabel zu achten, vor dem Einschalten soll die eigene Arbeit auch kritisch auf beim Löten entstandene Kurzschlüsse getestet werden[2]. Zunächst eine Laborspannungsquelle mit einstellbarer Strombegrenzung – oder ein Steckernetzteil mit maximal 2 A Ausgangsstrom zu Testzwecken tun hier gute Dienste, wobei auf die Einhaltung der zulässigen Betriebsspannungen (Transmitter $5-7$ V, Zündbox $8-15$ V) zu achten ist.

10.1 Kabel

Zur Verbindung der Platinen mit der Peripherie wird bei *El Fueradoro* eine Vielzahl von Kabeln benötigt, die grob in drei Kategorien unterteilt werden können:

1. Flachbandkabel bzw. Flachbandkabel-Adern zum Anschluss von LCD (außer Hintergrundbeleuchtung), LEDs und Schlüsselschalter

2. Litze mit einer Querschnittsfläche von mindestens 0,5 mm^2, d.h. einem Mindestdurchmesser von 0,8 mm, zum Anschluss von LCD-Hintergrundbeleuchtung, Netzschalter und Zündklemmen

3. 50 Ω-Koaxialkabel als Antennenkabel

Die Koaxialkabel werden in diesem Abschnitt nicht behandelt, da dafür oft Spezialwerkzeug notwendig ist und davon ausgegangen wird, dass diese Kabel bereits fertig konfektioniert erworben werden. Für die Konfektionierung der anderen Kabel gilt, dass diese so kurz wie möglich aber gleichzeitig auch so lang wie nötig sein sollten, um das Verlöten/Verkleben annehmbar zu gestalten und das Gehäuse später noch einmal öffnen zu können, ohne gleich alles abzureißen.

[1] Ein gutes – wenn auch englischsprachiges – Löt-Tutorial mit wichtigen Grundlagen gibt es HIER

[2] Hier klicken, um zu sehen, was ein kurzgeschlossener Blei-Gel-Akku mit Drähten/Leiterbahnen anstellt, sofern er nicht direkt explodiert!

Device (T/Z)	Art	Adern	Länge	Anzahl	Verwendungszweck
Z	Litze	–	210 mm	4	Rote Klemmen (spaltenweise)
Z	Litze	–	210 mm	16	Schwarze Klemmen
Z	Litze	–	160 mm	2	Netzschalter
Z	Litze	–	200 mm	4	Batterie (2x rot, 2x schwarz)
Z	Flachband	2	150 mm	16	Kanal-LEDs
Z	Flachband	2	200 mm	4	Status-LEDs
Z	Flachband	2	200 mm	1	Schlüsselschalter
T	Litze	–	100 mm	2	LCD-Hintergrundbeleuchtung
T	Flachband	2	100 mm	4	Status-LEDs
T	Flachband	10	100 mm	1	LCD

Tabelle 10.1: Übersicht über benötigte Kabelverbindungen

In Tabelle 10.1 sind die benötigten Abschnitte aufgelistet. Alle Kabel sollten am einen Ende jeweils auf einer Länge von 4 mm zum Festlöten an der Platine, am anderen auf einer Länge von 8 mm abisoliert und verzinnt werden. Zur Vereinfachung der Arbeit ist es ratsam, die Adern des Flachbandkabels erst danach zu trennen, so dass nicht jeder „Zweierverbund“ einzeln abisoliert und verzinnt werden muss.

Es dient der Übersichtlichkeit und dem späteren Verständnis ungemein, wenn man verschiedene Kabelfarben verwendet und sich dabei an gängige Konventionen hält (Akkuspannung rot, Masse schwarz).

10.2 Platinen

Ausgangspunkt der Bestückung ist die geätzte und gebohrte Platine mit allen Leiterbahnen auf der Ober- und Unterseite. Anschließend sollten die Bauteile in folgender Reihenfolge eingelötet werden:

Widerstände mit Ausnahme von R_Z

Dioden mit korrekter Polung

Keramikkondensatoren

Quarz

Buchsenleisten für Funkmoduladapter

Elektrolytkondensatoren Polung beachten und C14 beim Transmitter waagrecht legen!

Leistungswiderstand R_Z

Temperatursensor mit korrekter Orientierung

Integrierte Schaltungen mit korrekter Orientierung

Anschluss-Kabel für Klemmen (Zündbox) bzw. LCD (Transmitter)

Steckerkabel für Netz- und Schlüsselschalter

LED-Kabel für Status-LEDs und Kanal-LEDs. Um den Überblick zu behalten sollte dabei jeweils das Kabel für den GND-Anschluss vor dem Einlöten markiert werden

MOSFETs mit korrekter Orientierung (Metallplatte sitzt auf der Seite der großen Elkos)

Batterie- bzw. USB-Versorgungs-Kabel Falls bei der Zündbox fertig konfektionierte Kabel mit Anschluss verwendet werden, vor dem Löten die Seitenwand auf der Schalterseite durchbohren und beide Kabel durchfädeln. Beim Trasmitter wird das Kabel zwischen zwei miteinander verschraubten Teilen verlegt.

10.3 Funkmodul-Adapter

Nachdem die Platinen nun fertig bestückt sind, folgen die Schritte für die Fertigstellung des Funkmodul-Adapters:

Funkmodul mit allen 14 Anschlüssen (Orientierung beachten) auf der Adapterplatine festlöten. Dabei beachten, dass auf die SMD-Pads auf beiden Seiten etwa gleich weit unter den Anschlüssen hervorstehen und das Modul nicht nach oben oder unten verschoben ist. Am besten mit dem mittleren Pin einer Seite beginnen und das Modul beim Löten korrekt positionieren, dann alle anderen 13 Anschlüsse löten.

SMA-Buchse anlöten, so dass sie direkt an der Platine anliegt. Falls Anschlüsse zu lang sind und am Modul anstoßen, vorsichtig mit Trennscheibe (Außenleiter) oder Seitenschneider (Innenleiter) kürzen.

Stiftleiste in einen 6-poligen und einen 7-poligen Abschnitt teilen und anlöten.

10.4 Peripherie

10.4.1 Transmitter

Als Gehäuse für den Transmitter wird das schwarze Kunststoffgehäuse `GEH KS 50` aus dem Sortiment von Reichelt verwendet. Es besteht aus zwei miteinander zu verschraubenden Teilen, wobei der dünne Teil, auf dem später die Platine befestigt wird, als Rückwand dient. Dementsprechend sind die Bezeichnungen in der folgenden Beschreibung zu verstehen.

Abbildung 10.1: Abmessungen (in mm) des LCD

Die Vorderseite des Gehäuses muss zunächst mit einem Ausschnitt von 97 x 39,5 mm und vier 3mm-Bohrungen für die Anbringung des LCD versehen werden. Die Abstände zueinander sind in Abbildung 10.1 verdeutlicht. Wer nicht auf seine Messkünste vertrauen möchte oder ein LCD mit abweichenden Abmessungen besitzt, sollte zunächst einen passenden Ausschnitt für den Bildschirm mittig in der Fläche anbringen, anschließend können das LCD aufgelegt und die vier Bohrlöcher markiert und gebohrt werden.

Bevor man das LCD einschraubt, muss noch eine Aussparung für den Sub-D-Anschluss, ein Loch für den Antennenanschluss (6,5 mm), vier Löcher für die Status-LEDs (5 mm)und eine Kerbe für die Versorgungskabeldurchführung in die Wand, welche später als Oberseite des Gehäuses dient, eingebracht werden.

Hierfür sollte man zunächst die fertig bestückte Platine mittels zweier Schrauben in ihrer finalen Position festschrauben und das Oberteil so anlegen, dass die Position der Aussparung für die Sub-D-Buchse angezeichnet werden kann. Die Abmessung der Aussparung sollte 31 x 12,5 mm betragen. Die Positionen der restlichen Löcher sind aufgrund der Kabelverbindung mit der Platine unkritisch, die ungefähre Lage kann Abbildung 1.2 auf Seite 17 entnommen werden. Das Versorgungskabel mit USB-Stecker wird zwecks Zugentlastung zwischen den beiden zu verschraubenden Teilen eingeklemmt, die Kerbe sollte daher nicht allzu groß ausgeführt werden. Die Bohrarbeiten am Transmitter sind damit erledigt!

Nun müssen noch die elektrischen Verbindungen zwischen Platine und Peripherie hergestellt und die Peripherieteile anschließend befestigt werden – zuerst die vier Status-LEDs:

1. Die Verbindung der Adern des Flachbandkabels auf einer Länge von etwa 35 mm auftrennen

2. Auf jede Ader einen dünnen Schrumpfschlauch der Länge 15 mm stecken (noch nicht erhitzen!)

3. Das kürzere Anschlussbein der LED (Kathode) mit dem Seitenschneider auf eine Länge von 8 mm trimmen und die mit GND verbundene Ader anlöten

4. Das längere Anschlussbein der LED (Anode) mit dem Seitenschneider auf eine Länge von 8 mm trimmen und die andere Ader anlöten

5. Schrumpfschläuche bis ans LED-Gehäuse vorschieben und per Heißluft schrumpfen

Anschließend das Flachbandkabel für das LCD vorbereiten, d.h. abisolieren, verzinnen und Verbindungen soweit lösen, dass alle Anschlüsse bequem erreicht werden können. Da das LCD im 4-Bit-Modus betrieben wird, werden nur die Pins 1-6 sowie 11-16 angeschlossen, 7-10 bleiben offen. Bei Anschluss der Pins 15 und 16 darauf achten, Anode und Kathode nicht zu vertauschen; die Belegung ist in der Regel so, dass die Anode an Pin 15 herausgeführt ist, kann aber von LCD zu LCD variieren. Der Lötkolben kann danach ausgeschaltet werden, jetzt geht es an die Befestigung.

Zunächst wird das LCD am Gehäuse festgeschraubt, wobei darauf zu achten ist, dass die Oberseite auch in die Richtung von Antennen- und Sub-D-Anschluss zeigt. Anschließend die LEDs um den Gehäusering mit Sekunden- oder Heißkleber bestreichen und anschließend für einige Sekunden fest in eines der dafür vorgesehenen Löcher pressen. Nun die SMA-Buchse fest am Gehäuse anschrauben und das andere Kabelende mit der Buchse am RFM-Adapter verbinden. Unter möglichst geringer Torsion sollte dann der Adapter in die Buchsenleisten auf der Platine gesteckt werden.

Nun noch die Antenne anschrauben. Wenn der Bootloader sich bereits auf dem Controller befindet, kann man das Gehäuse zuschrauben. Ansonsten den Transmitter mit Energie versorgen und den Bootloader wie in Abschnitt 11 beschrieben flashen. Jetzt ist der Transmitter fertig aufgebaut und kann zugeschraubt werden! Die Firmware sollte kann wie in Abschnitt 4 beschrieben über die serielle Schnittstelle aufgespielt werden, wobei beim ersten Mal noch die Angabe des Dateinamens nötig ist.

10.4.2 Zündbox

Abbildung 10.2 zeigt die Bohrschablone für die Oberseite des Kunststoffgehäuses der Zündbox (Kunststoffgehäuse 021-002-084 von Pollin). Diese kann dazu verwendet werden, eine Schablone aus Sperrholz oder Metall anzufertigen, welche später auf die Boxenoberseite gelegt wird, um die nötigen Bohrungen vorzunehmen.

Es ist beim Bohren der Box auf die richtige Orientierung der Schablone zu achten, da die

Befestigungsbohrungen für die Platine nicht symmetrisch sind (Schrauben auf der Seite der seriellen Schnittstelle sind weiter außen als die auf der Kanal-LED-Seite) und die Gehäuseteile durch ein Nut-Feder-System nur in einer Kombination aufeinander gesteckt werden können.

Damit die Löcher nicht mit Schraubenhalterungen in den Ecken interferieren, sollte zunächst im Gehäuseinneren der passende Ort für die Kanal-LEDs von Kanal 4 und 16 (äußerste grüne LEDs in der obersten und untersten Reihe, Abstand 75 mm) gesucht, die beiden Löcher mit einem Bohrer < 3 mm vorgebohrt und die Schablone nach diesen beiden Bohrungen ausgerichtet werden.

Die Bohrlöcher für die 16 Kanal-LEDs sollten mit einem 3-mm-Bohrer, die der vier Status-LEDs mit einem 5-mm-Bohrer ausgeführt werden. Die LEDs werden später (nach dem Verkabeln und Festlöten des Kabels auf der Platine) seitlich mit Heißkleber bestrichen von unten bis zum Anschlag in diese Löcher eingeschoben.

Da die Lautsprecherklemmen mit M3-Gewindeschrauben befestigt werden, wäre der ideale Bohrdurchmesser für die 16 Schraubenlöcher 3,2 mm. Sollte dieser Durchmesser nicht vorhanden sein, kann aber auch mit 3,5 mm gearbeitet werden. Für die Lötfahnen der Klemmen ist eine rechteckige Aussparung von $4{,}5x2$ mm nötig, als schnelle Lösung kann auch jeweils ein 5-mm-Loch durch den Mittelpunkt (Diagonalenschnittpunkt) dieser Flächen gebohrt werden.

Die rechteckige Aussparung für den Netzschalter sollte die Größe $19x13$ mm besitzen, der Schlüsselschalter hat einen Einbaudurchmesser von 12 mm und der Durchsteckplatz für die SMA-Buchse sollte mit 6,5 mm vorgebohrt werden. Im nächsten Schritt werden Schlüsselschalter, Klemmen und Netzschalter am Gehäuse verschraubt bzw. eingeklickt.

In die linke Einschub-Seitenwand muss, sofern noch nicht im Rahmen des Lötens geschehen, ein Loch für die Durchführung der Batteriekabel mit einer Größe je nach Kabeldurchmesser.

Um die Position der Sub-D-Buchse zu finden, muss die Platine möglichst identisch zu ihrer späteren Position ins Gehäuse eingelegt werden, die Aussparung beträgt wie beim Transmitter 31 x 12,5 mm. Wenn die Aussparung fertig ist, kann die Platine in ihre endgültige Position gebracht und mit vier Schrauben befestigt werden. Die linke Seitenwand bleibt noch ausgesteckt. Jetzt ist wieder der Lötkolben dran!

Zunächst vier Drahtstücke á 800 mm abschneiden und durch die übereinander liegenden Anschlüsse für die roten Klemmen ziehen und mit den Klemmen verlöten.

Die Kanal- und Status-LEDs wie auf Seite 62 beschrieben mit den zugehörigen Flachbandkabeln verbinden, anschließend den Schlüssel- und den Netzschalter mit den zugehörigen Anschlusskabeln.

Bei den Schritten in den folgenden beiden Absätzen ist Sorgfalt geboten, da die Kanäle und LEDs richtig zugeordnet werden müssen, um später mit dem entsprechenden Befehl auch den richtigen Kanal zu zünden!

Zunächst müssen die 16 Kabel mit den zugehörigen schwarzen Klemmen verbunden werden,

Abbildung 10.2: Bohrschablone für Zündboxoberseite

was anhand des Layouts in Abbildung 5.4 erklärt werden soll. Kanal 1 wird vom Transistor Q2 gesteuert, Kanal 2 von Q3, Kanal 3 von Q4 und allgemein Kanal x-1 von Qx . Entsprechend ist das Kabel, welches an derjenigen Lötstelle angelötet ist, die durch die dicke blaue Linie unmittelbar mit dem mittleren Pin (Drain) von Q2 verbunden ist, an der Lötfahne der schwarzen Klemme ganz unten links – bezogen auf Abbildung 1.3 – anzubringen. Das Kabel an der Drain von Q3 (Zick-Zack-Anordnung der Transistoren beachten!) wird mit der schwarzen Klemme daneben verbunden, das an Q4 mit der dritten und das an Q5 schließlich mit der letzten Klemme in der Reihe, die unmittelbar neben der Viererreihe für die Kanal-LEDs liegt. Man sollte sich nicht dadurch verwirren lassen, dass aufgrund des umgedrehten Deckels alles seitenverkehrt ist, man die Klemmen also beim Löten von rechts nach links belegt. Analog zum bisherigen Vorgehen verfährt man in der Reihe darüber und den beiden anderen Reihen.

Nun werden mit Heiß- oder Sekundenkleber die Kanal-LEDs in die richtige Position gesteckt. Für Kanal n ist dabei immer auch LEDn zuständig, für Kanal 1 also LED1, die über R21 mit der Drain von Q2 verbunden ist, für Kanal 2 LED2, usw. Sinnvollerweise ist LED1 in das Loch zu kleben, welches in der untersten Reihe direkt neben den Klemmen liegt, LED8 dementsprechend in der zweituntersten Reihe ganz außen usw.

Danach werden die vier Kabel am Leistungswiderstand jeweils mit einem der gespannten Drähte an den roten Klemmen verbunden. Hierbei spielt die Zuordnung (welches Kabel an welchen Draht?) keine Rolle. Damit sind sämtliche Kabel im inneren der Box nun verlötet und die Arbeit nähert sich langsam dem Ende!

Nun die SMA-Buchse fest am Gehäuse anschrauben, die Antenne vorerst dort befestigen und das andere Kabelende mit der Buchse am RFM-Adapter verbinden. Unter möglichst geringer Torsion sollte dann der RFM-Adapter mit dem aufgelöteten Funkmodul in die Buchsenleisten auf der Platine gesteckt werden.

Wenn der Bootloader bereits auf den Controller geflasht wurde, kann man jetzt die beiden Seitenwände einstecken und die Box zuschrauben (wenn man sich sicher ist, dass sie funktioniert. . .). Ansonsten muss die Box noch offen bleiben.

Den männlichen Teil des Steckerpaars mit den aus der Box kommenden Batteriekabeln verlöten (Schrumpfschlauch nicht vergessen!), den weiblichen über zwei Kabel an der Batterie, wobei jeweils auf die korrekte Polung zu achten ist[3].

Für die erste Inbetriebnahme sollte, wie zu Beginn des Kapitels empfohlen, idealerweise eine Laborspannungsquelle, ein kurzsschlussfestes Steckernetzteil oder aber eine träge 3 A-Sicherung in der Zuleitung verwendet werden. Nun die Box über den Netzschalter einschalten und, falls noch nicht geschehen, den Bootloader via ISP flashen. Nun kann die Box zugeschraubt und die Firmware eingespielt werden.

[3] Bei Deans-T-Steckern wird der obere Balken des T mit + verbunden, bei Tamiya-Steckern das eckige Profil

10.4.3 Koffer

Kommt noch…

11 Aufspielen des Bootloaders

Bevor Firmwareupdates über die serielle Schnittstelle eingespielt werden können, muss zunächst ein Programm auf den Controller gespielt werden, dessen Aufgabe es ist, die eigentliche Firmware in den Speicher zu laden und zu starten. Dieses Programm ist der so genannte Bootloader, welcher beim Start des Devices für eine Sekunde überprüft, ob ein Firmwareupdate vorgenommen werden oder die *El Fueradoro*-Firmware normal ausgeführt werden soll.

Abbildung 11.1: Pinbelegung des ISP-Platinensteckers: Ansicht von oben, Aussparung an Pin 5. Quelle: `mikrocontroller.net`

Um den Bootloader auf den Controller zu brennen und einige Grundeinstellungen des Controllers, die so genannten Fuses, welche neben den Einstellungen, welche die Verwendung eines Bootloaders ermöglichen, auch Funktionen wie die Brown-Out-Detektion oder die Taktquelle regeln, wird eine spezielles Programmiergerät zur In-System-Programmierung (ISP) benötigt, welche den Controller in den Resetzustand versetzt und anschließend den Bootloader über die SPI-Schnittstelle an eine festgelegte Stelle im Flash-Speicher schreibt.

Auf der Platine jedes Devices ist für ISP ein zehnpoliger zweireihiger Wannenstecker angebracht, an den gängige Programmiergeräte wie der weit verbreitete *AVRISP mkII* angeschlossen werden. Seine Pinbelegung ist in Abbildung 11.1 gezeigt.

Zum Brennen des Bootloaders gibt es ein Kommandozeilentool namens *btldflsh.exe* für den *AVRISP mkII*, welches auf *AVRDUDE* basiert. Dem Tool muss als Parameter die iHex-Datei des controllertyp- und frequenzspezifischen Bootloaders übergeben werden.

Für einen `ATmega328P` mit einer Taktfrequenz von 9,8304 MHz lautet das Kommando:

btldflsh.exe bootload_m328p_9830400.hex

Die für *El Fueradoro* benötigten Einstellungen für Fuses und die Datenübertragung werden auf

diese Weise automatisch angepasst. Nach einmaligem Flashen des Bootloaders wird die ISP-Schnittstelle nicht wieder benötigt.

12 Tipps und Tricks

12.1 LCD an 3,3 V

Zwar gibt es inzwischen auch LCDs, welche für eine Versorgung mit 3,3 V ausgelegt sind, viele Displays jedoch sind noch für 5 V ausgelegt. Der interne Controller funktioniert ohne Probleme auch bei geringerer Spannung, Knackpunkte sind jedoch die Kontrastspannung für das LCD sowie die Hintergrundbeleuchtung.

Die Kontrastspannung wird zwischen Pin 3 und Pin 2 gemessen und ist verantwortlich für die Lesbarkeit der Schrift auf dem Display. Im Normalfall wird Pin 3 auf 0 V gelegt, so dass eine Differenz von 5 V entsteht. Liegen an Pin 2 nur 3,3 V an, muss an Pin 3 folglich eine negative Spannung anliegen, um die nötige Kontrastspannung zu erreichen. Um diese Spannung zu erreichen, wird der RS232-Treiberbaustein „angezapft“, der am Pin V- eine Spannung von −5,5 V zur Verfügung stellt. Mit einem Spannungsteiler zwischen V- und GND wird daher Pin 3 des LCD auf −1,7 V gelegt.

Manche LCDs torpedieren diesen Versuch, indem Pin 3 relativ niederohmig mit GND verbunden wird. Man sollte also im abgeklemmten Zustand den Widerstand zwischen Pin 1 und Pin 3 messen und bei Bedarf den eventuell auf der LCD-Platine befindlichen Widerstand zwischen den beiden Pins auslöten.

Die Hintergrundbeleuchtung wird über die Pins 15 und 16 versorgt. In Reihe zu den LEDs befindet sich zwischen diesen Pins in aller Regel noch ein Widerstand, der den Strom durch die LEDs begrenzt und auf 5 V zwischen den Pins ausgelegt ist. Bei 3,3 V zwischen Pin 15 & 16 erscheint das LCD daher möglicherweise zu dunkel, so dass man den Widerstand durch einen kleineren Wert ersetzen kann, um den Strom zu erhöhen.

12.2 Antennenbau

Antennen für die verwendete Übertragungsfrequenz von 868 MHz gibt es in großer Auswahl zu kaufen, eine einfache, omnidirektionale Antenne, welche ein sehr gutes Stehwellenverhältnis von <1,3:1 erzielt, kann aber auch relativ schnell selbst gebaut werden. Auf möglichst exakte Einhaltung der Abmessungen ist dabei zu achten:

- Koaxialkabel RG316

Abbildung 12.1: Antenne ohne Schrumpfschlauch (links) und komplett fertig (rechts)

- SMA-Steckverbinder (üblicherweise männliche Ausführung)
- Kupfer- oder Messingröhrchen mit 8 mm Durchmesser und 66,5 mm Länge
- Distanzhülse aus Kunststoff mit 5 mm Länge, 7 mm Außendurchmesser und 3,6 mm Innendurchmesser
- Schrumpfschläuche mit 1,2 mm, 2,4 mm, 4,8 mm und 9,5 mm Durchmesser vor dem Schrumpfen

Die Antenne wird als Sperrtopfantenne bezeichnet und besitzt den in Abbildung 12.1 gezeigten Aufbau. Der oberste Teil ist 88 mm lang und besteht aus dem Innenleiter des Koaxialkabels mit dem ihn umgebenden Dielektrikum. Im mittleren Teil befindet sich ein 66,5 mm langes Röhrchen, welches an seinem oberen Ende mit möglichst kurzer Verbindung an das Schirmgeflecht des Koaxialkabels angelötet wird. Anschließend folgt eine beliebige Länge Koaxialkabel, am Ende schließlich der Steckverbinder zum Anschluss an das Funkmodul.

Der Aufbau der Antenne erfolgt folgendermaßen:

1. Von der Rolle RG316 ein Stück der Länge abschneiden, welche später der Gesamtlänge von Antennenspitze bis zum Anschluss an das Funkmodul bzw. einen Adapterstecker entspricht
2. Entfernung des Kunststoffmantels auf einer Länge von 88 mm

3. Entfernung des nun freiliegenden Schirmgeflechts auf einer Länge von 84 mm, so dass noch 4 mm des Schirmgeflechts verbleiben.

4. Auftrennen und Verdrillen des Schirmgeflechts

5. Verdicken des Koaxialkabels mit einem 10 mm langen Stück Schrumpfschlauch, dessen Mitte 66,5 mm vom oberen Ende der Ummantelung entfernt sein sollte.

6. Aufschieben der Distanzhülse in die Mitte des soeben verstärkten Teils. Gegen Abrutschen nach unten ggf. mit weiterem Schrumpfschlauch unterhalb der Hülse sichern.

7. Großzügiges Vorverzinnen des Röhrchens an einer Stelle der Innenwand

8. Überstülpen des Röhrchens und Anlöten des verdrillten Schirmgeflechts an der Innenwand

9. Gesamte Konstruktion mit Schrumpfschlauch stabilisieren (nach jedem Schritt schrumpfen!):

   a) Ein 88 mm langes Stück Schrumpfschlauch 1,2 über den obersten Teil der Antenne

   b) Ein 90 mm langes Stück Schrumpfschlauch 2,4 über den obersten Teil der Antenne, unmittelbar nach dem Schrumpfen die noch heißen oben überstehenden 2 mm Schlauch durch Zusammendrücken verschmelzen

   c) Ein 2,5 mm langes Stück Schrumpfschlauch 9,5 über den aus dem Röhrchen herausstehenden Teil der Distanzhülse

   d) Ein 80 mm langes Stück Schrumpfschlauch 9,5 über Röhrchen und Distanzhülse, so dass auf beiden Seiten etwa 5 mm überstehen

   e) Mit einem 10 mm langen Stück Schrumpfschlauch 4,8 den Übergang zwischen oberem Antennenteil und Röhrchen versiegeln

10. Falls nötig: Kabel in Endposition einfädeln, bevor SMA-Steckverbinder am intakten Ende angebracht wird

Um den SMA-Steckverbinder anzubringen, müssen vom intakten Ende aus gemessen zunächst 10 mm des Mantels entfernt werden, anschließend 5 mm des Schirmgeflechts und zuletzt auch 2,5 mm des Dielektrikums. Nun werden Schrumpfschlauch und Crimpröhrchen auf das Kabel geschoben, anschließend der kleine Stecker am Innenleiter angelötet. Das Schirmgeflecht wird aufgefächert, das Gehäuse aufs Kabel geschoben und das Crimpröhrchen aufgesteckt, vercrimpt und Schrumpfschlauch darüber angebracht. Wenn gewünscht kann – wie in 12.1 rechts zu sehen – Schrumpfschlauch 9,5 als Witterungsschutz über dem gesamten Steckverbinder angebracht werden.

Die in Abbildung 12.2 dargestellte Messung am Netzwerkanalysator zeigt eine sehr gute Anpassung dieser Antenne an 50 Ω im Bereich um 868 MHz.

Abbildung 12.2: VSWR-Messung der gefertigten Antenne am Netzwerkanalysator

12.3 Kompilieren der Firmware

Falls keine der im Repository zur Verfügung gestellten Hex-Files verwendet werden soll, besteht auch die Möglichkeit, seine eigene Firmware aus den Quellcodedateien zu kompilieren. Hierfür wird der die AVR-GCC-Toolchain benötigt, welche es für Windows z.B. als fertiges Paket `WinAVR`[1] gibt.

`WinAVR` entspricht leider schon länger nicht mehr dem aktuellen Stand des GCC-Compilers, man kann entweder bei Atmel[2] oder bei im Sourceforge-Repository[3] von Georg-Johann Lay aktuellere Versionen ziehen und diese über die alten `WinAVR`-Verzeichnisse kopieren.

Anschließend ist darauf zu achten, dass das Unterverzeichnis „bin“ in der PATH-Variable hinterlegt ist, so dass man die dort befindlichen Anwendungen aus jedem anderen Ordner unmittelbar aufrufen kann.

Der Herstellungsprozess gliedert sich in folgende Schritte:

1. Kompilieren der Quellcodedateien (.c) in noch nicht gelinkte Dateien (.o)

2. Linken zu einer Gesamtdatei (.elf)

3. Umwandeln in eine Hexdatei (.hex), welche mit dem Bootloader über die serielle Schnittstelle auf den Controller geschrieben werden kann

[1] Download WinAVR
[2] Download Atmel-Toolchain (kostenlose Registrierung notwendig)
[3] Downloadlink: Sourceforge

Der gesamte Ablauf kann transparent in der Datei „build_hexfiles.bat“ im Verzeichnis „Hexfiles“ nachvollzogen werden.