

Tetris-Display mit Handy-Steuerung

Raspberry Pi steuert LED-Matrix

Die auf den LEDs mit integriertem Controller basierenden LED-Matrixanordnungen erfreuen sich stetig steigender Beliebtheit, machen sie doch den Aufbau selbst sehr großer LED-Displays einfach. Ansteuerungen wie die Diamex-TPM2-Umsetzer und Freeware-Steuerprogramme wie Jinx! oder Glediator erleichtern dazu die Steuerung dieser Displays. Folker Stange und Daniel Wiese [1] haben ein Stand-alone-Projekt entwickelt, das es ermöglicht, eine 16x32-Matrix per WLAN vom Handy aus zu steuern, und als Applikationen die beliebten Spiele Tetris und Snake implementiert. Wir fanden das Projekt so spannend und vor allem im Endeffekt für jeden, freilich mit seinen eigenen Möglichkeiten, nachvollziehbar, dass wir die beiden gebeten haben, es hier im Rahmen unseres Leserwettbewerbs vorzustellen.



Folker Stange und Daniel Wiese
haben für ihren Beitrag zum Leserwettbewerb
einen 200-Euro-Warengutschein erhalten!

Raspberry Pi an Handy

Die Grundidee für dieses Projekt entstand, als ich im Netz auf die Tetris-Konstruktion in [2] stieß. Diese entstand auf Arduino-Basis, wird mit einem Spielcontroller per Kabel gesteuert und ist im Gesamtergebnis genial, da hier mit minimalistischen Mitteln eine funktionierende Lösung erzielt wurde. Freilich stößt man mit dem Arduino irgendwann an seine Grenzen, was in [2] auch nachvollziehbar diskutiert wurde. Interessant fand ich die Lösung, die bei Tetris störenden runden und gestreuten Lichtpunkte der LEDs mit einem selbst gebauten Raster aus Pappe in quasi quadratische Pixel umzuwandeln. Derartige Raster in ähnlicher Form haben sich ja auch schon bei anderen Eigenbau-Lösungen bewährt, wie z. B. [3] und [4] zeigen. Die Anregung war da, also ging ich an das eigene Hardware-Projekt, während Daniel sich der Software-Entwicklung widmete. Als Hardware-Basis wurde der Raspberry Pi ausgewählt, er erhält mit dem Betriebssystem einen eigenen Web-Server, der via WLAN-Modul ein eigenes WLAN aufspannt, in

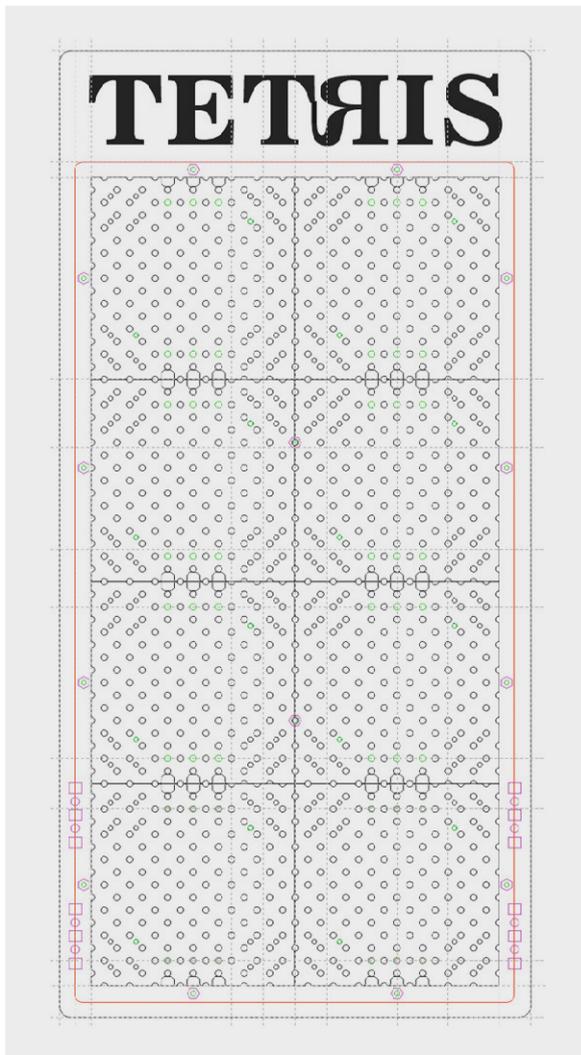


Bild 1: Die Konstruktionszeichnung der Panel-Grundplatte

das man sich mit Handy, Tablet etc. einloggen und dann über eine eigene Oberfläche spielen kann. Dazu ist letztlich nicht einmal eine App auf dem mobilen Gerät notwendig.

Die Hardware

Zunächst wählte ich die Komponenten aus. Mit acht Stück 8x8-WS2812-Panels kommt man auf die Kantenlänge von 16 x 32 Pixel. Das ist für Tetris mehr als ausreichend. Als Rechenkern ist der Raspberry Pi eine interessante Wahl. Mit Tricks und zusätzlichen

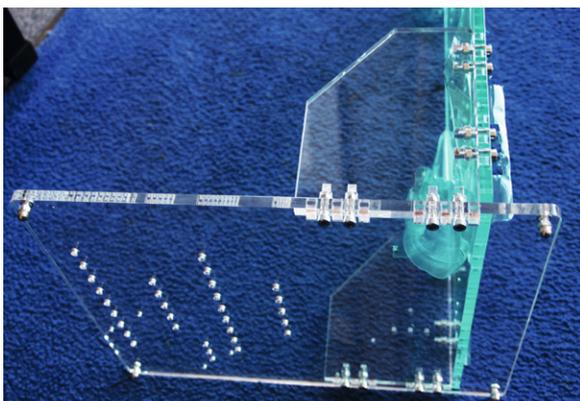


Bild 3: So werden Displayplatte und Standfuß stabil miteinander verbunden.

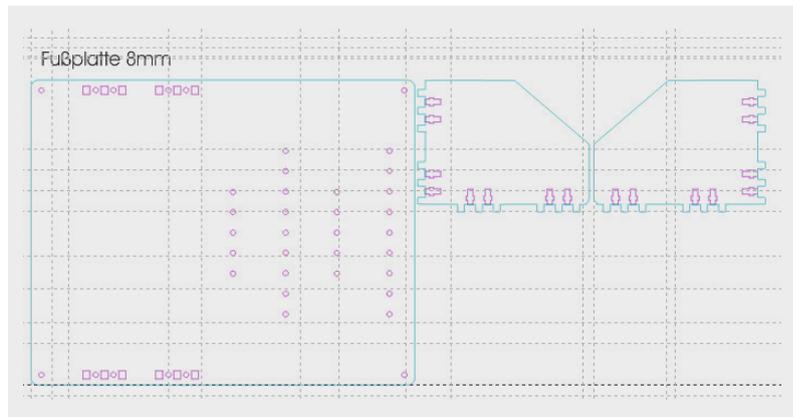


Bild 2: So sieht die Konstruktionszeichnung für die Fußplatte aus.

Bibliotheken kann die Ansteuerung der LEDs mit dem WS2812-Protokoll direkt über GPIO-Pins erfolgen, das Ganze ist jedoch sehr zeitkritisch und umständlich zu verwenden, einfacher geht es mit einem LED-Controller, der via TPM2 angesteuert wird. Das geschieht über USB, ist also recht einfach anzubinden. Meine Wahl ist daher auf den Matrix-Player als Umsetzer gefallen. Der kann bis zu 1024 LEDs ansteuern, ist also gut geeignet und kann direkt auf das Panel geschraubt werden.

Die Mechanik – per Lasercutter „gebaut“

Der nächste Schritt sind die mechanischen Komponenten und die mechanische Konstruktion. Als langjähriger Lasertechniker besteht für mich kein Zweifel am Material, stylisches Plexiglas lässt sich hochpräzise auf einem CO₂-Lasercutter verarbeiten. Die Konstruktion wird in CorelDraw gezeichnet, das ist ein tolles Programm, um 2D-Vektorgrafiken hochpräzise zu erstellen. Allerdings gehört schon eine ordentliche Portion Erfahrung dazu, in Corel auch zum gewünschten Ergebnis zu kommen. Wie gesagt, Corel ist 2D-basierend und der Lasercutter ist das ebenfalls – passt also wunderbar zusammen, zumal der Lasercutter als Drucker direkt aus Corel eingesetzt wird. Man lässt also die Konstruktion einfach auf Plexiglas ausdrucken.

Zuerst konstruiert man die Display-Grundplatte, die ist abhängig von der Größe und Anordnung der Panels. Ich machte mir es einfach und importierte die DXF-Daten der Panels (das sind die schwarzen Quadrate mit den vielen Bohrungen in Bild 1) und kann diese so in CorelDraw wie gewünscht anordnen, in diesem Falle im 2x4-Raster. Toll an den hier eingesetzten Matrix-Panels (Bild 2) ist, dass sie sich wirklich simpel montieren und nahtlos anordnen lassen. Das ist bei keinem mir bekannten, vergleichbaren Produkt möglich. Bei Vollast werden die Panels schon mitunter recht warm, Kleben fällt so als Montagehilfe aus. Die Panels haben M3-Montagelöcher – sogar 16 Stück –, man ist sehr variabel. Dazu kommt, dass die Strom- und Steueranschlüsse großzügig in M4 ausgelegt sind und die Verdrahtung auf dem Panel eine sehr große Fläche hat. Auch die Systembohrungen haben einen Sinn, da kann man ganz prima eine Wabenstruktur verankern, die sollte am Ende exakt aussehen, was sich am Ende tatsächlich so gezeigt hat – die Teile rasten nahtlos mit einem Click wie Lego-Bausteine ein.

Aufgrund der DXF-Importe kann ich ganz genau in CorelDraw die Panels anordnen und dann auch ganz genau die Befestigungsbohrungen (grün, Bild 1) setzen, das geht auf den 100stel Millimeter genau.

Um die Matrixstruktur lege ich einen Rand und darüber hinaus sehe ich noch eine Fläche vor, wo ich Platz für den Schriftzug *TETRIS* habe – wie in Bild 1 zu sehen. Den graviert der Laser gleich mit ein. Die rote Linie markiert die zum Schluss aufmontierte Streuscheibe. Nun muss der Fuß erstellt werden – das Display steht senkrecht und hochkant – es braucht also eine massive Fußplatte. Display und Fuß werden im Endeffekt aus 8-mm-XT-Plexiglas geschnitten, haben also ein respektables Eigengewicht. Mittels gleich mitkonstruierter Haltewinkel wird das

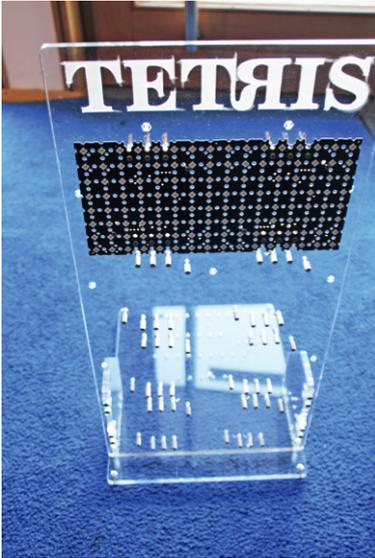


Bild 4: Der Lasercutter hat genau gearbeitet – die Panels passen perfekt!

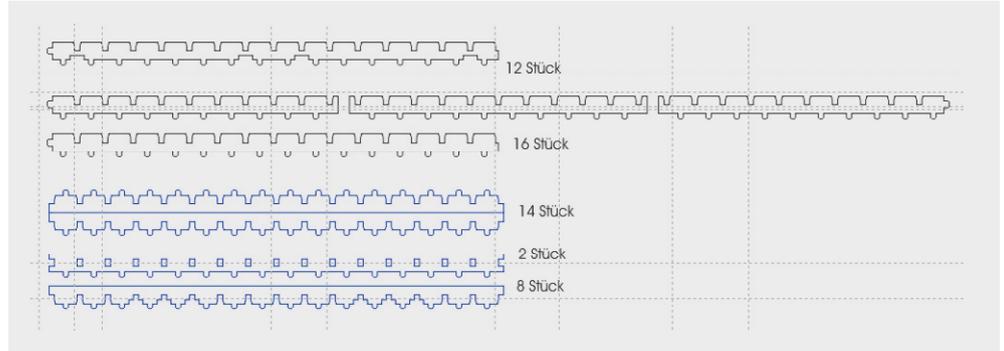


Bild 5: Die Einzelteile für das Wabengitter

GANZE mit acht M5-Schrauben für die feste Verbindung zu 90 Grad verschraubt, wie das aussieht ist in Bild 3 gezeigt. Bild 4 zeigt schon einmal die Anordnung mit zwei testweise montierten 8x8-Matrix-Displays.

Die Konstruktion der Wabenstruktur ist etwas trickreicher, da braucht man schon Erfahrung und Vorstellungsvermögen, wie alles räumlich zusammengesteckt wird. In der Mitte sind es Waben, am Rand ein Streifen, um alles schön aussehen zu lassen. Die Wabenstruktur (Bild 5) geht immer nur über zwei Panels, Plexiglas hat beim Schneiden über große Längen und schmale Breiten die unangenehme Eigenschaft, sich zu verziehen. Man muss fragmentieren. Auf die Wabenstruktur wird eine True-LED-Plexiglasscheibe aufgesetzt, die das LED-Licht sehr gleichmäßig verteilt. Diese Scheibe muss auch Befestigungslöcher haben, um am Ende plan aufzuliegen, sonst gibt es verwischte Kanten. Das Schneiden der beiden Platten dauert bei dieser Stärke und einem 60-Watt-Lasercutter eine gute Stunde. Das Ergebnis sieht dafür aber schon beeindruckend aus. Selbst lose zusammengesteckt macht es einen sehr wertigen und stabilen Eindruck.

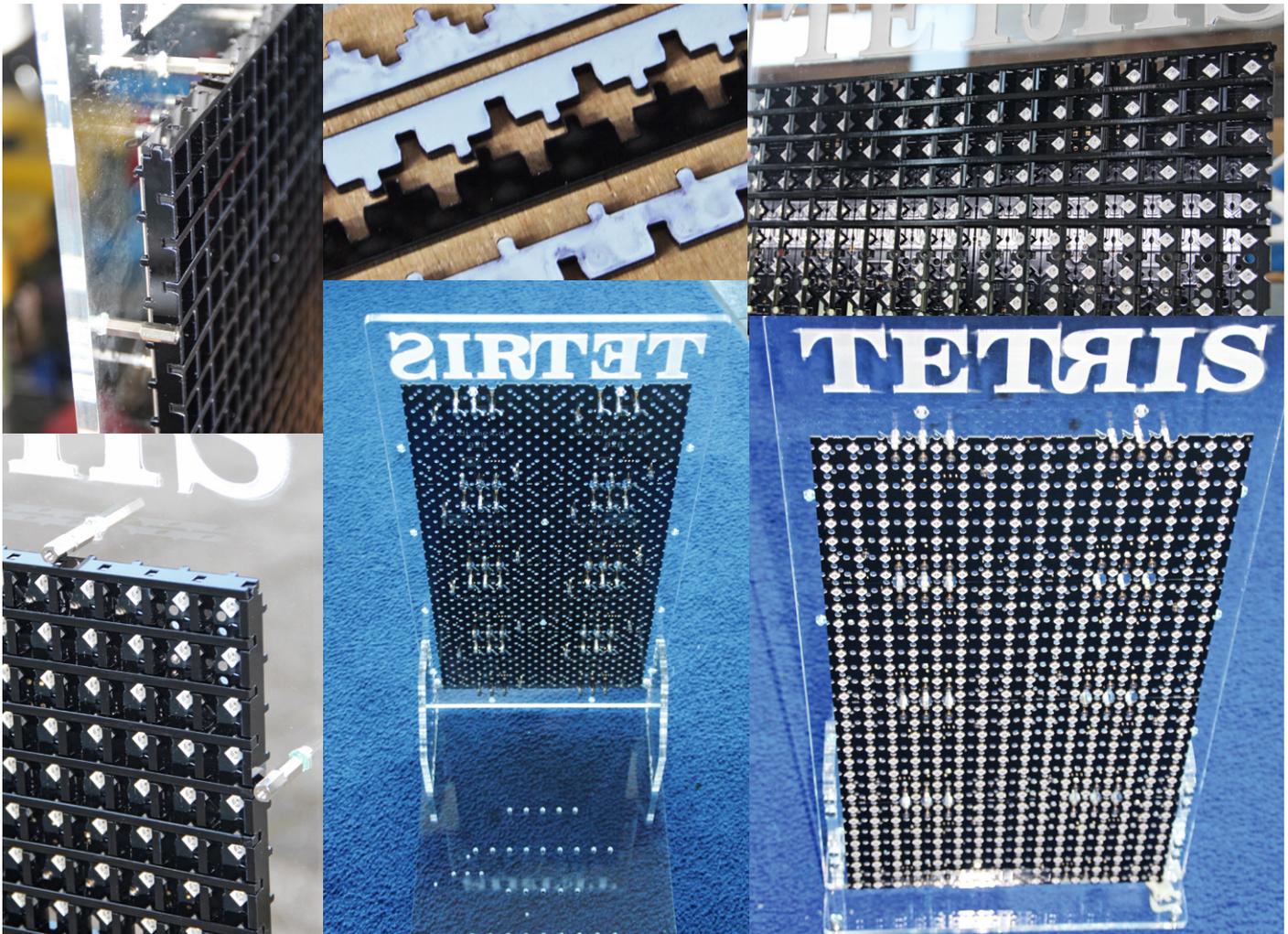


Bild 6: Die Montage des Wabengitters und der Panels erfordert etwas Zeit, aber das Ergebnis kann sich sehen lassen.



Bild 7: Bei der Montage der LED-Panels werden gleich noch die Lötösen für die Verdrahtung mit montiert.

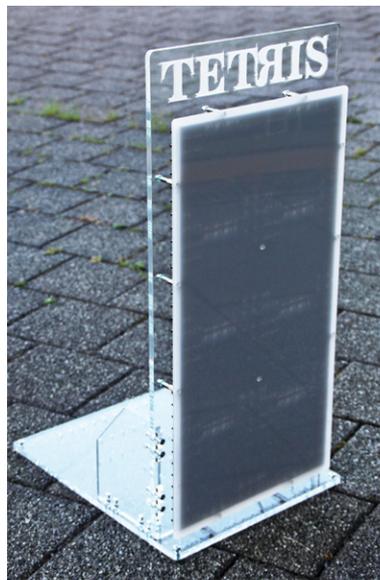


Bild 8: Zum Abschluss wird die Streuscheibe aufgesetzt und mit Kunststoffschrauben befestigt.



Bild 9: Das Testprogramm läuft – hier kann man die Wirkung des Wabengitters und der Streuscheibe bereits gut erkennen.

Fußplatte, Haltewinkel und Displayplatte passen (wie bei einem professionellen Lasercutter zu erwarten) perfekt zusammen. Auch die Panels sind in relativ kurzer Zeit montiert. Die Bilderstrecke in Bild 6 zeigt die Montage der Wabenstruktur und der LED-Panels.

Die Montage dauert eine gute Stunde, wegen der schier Masse an Abstandhaltern und wegen der im nächsten Schritt erforderlichen elektrischen Verbindungen. Es müssen also gleich Lötflächen untergeschraubt werden, ohne diese Anschlusspunkte (siehe Bild 7) gibt es am Ende kein Tetris. Wenn jetzt die Schraubverbindungen nicht gewissenhaft angezogen worden sind, gibt es Probleme bei der Inbetriebnahme und im schlimmsten Falle die komplette Demontage.

Die Wabensegmente sind nun montiert, alles passt prima. Zum Schluss wird die aus speziellem Plexiglas

gefertigte Streuscheibe (True-LED) plan auf die Wabenstruktur aufgesetzt und mit durchsichtigen Plastikschrauben fixiert (Bild 8). Metallschrauben würden hier einfach die schöne Optik stören!

Mit montierter Streuscheibe sieht das „Monster“ weniger bedrohlich aus, doch nun wird die Elektronik erstellt und montiert und nach deren Inbetriebnahme hoffen wir auf sehr helle, schön kontrastreiche Klötzchen, um das perfekte Tetris-Erlebnis zu genießen. Vorab schon einmal in Bild 9 das im Matrix-Player ab Werk integrierte LED-Testprogramm in Aktion ...

Wichtig ist zum Abschluss des Hardware-Kapitels noch, ein Wort zum Strombedarf des Displays zu verlieren. Eine LED nimmt im Extremfall bis zu 60 mA (weiß, maximaler Strom) auf. Der Fall tritt bei unserem Tetris quasi nie ein, zumal die LED-Helligkeit per Fernsteuerung einstellbar ist. Dennoch sollte das eingesetzte Netzgerät (siehe Stückliste im Anhang) leistungsfähig genug sein, um nicht an seine Grenzen zu geraten.

Anmerkung der Redaktion: Unser Nachbau musste sich vier Tage im Dauerbetrieb auf einem Technik-Festival bewähren, ein 16-A-Netzteil hat die Aufgabe bei nur mäßiger Wärmeentwicklung bewältigt.

Fernsteuerung per Handy

Daniel hatte sich derweil der Steuerungssoftware angenommen und in den letzten Wochen für den Raspberry Pi ein Betriebssystem aufgesetzt, das ein WLAN-Netz mit dem Namen „Games“ aufspannt und natürlich auch Tetris enthält.

Mit dem Smartphone kann man sich ins offene WLAN „Games“ einloggen und einen Browser starten. Über die IP 192.168.0.1 sind die Richtungstasten sowie



Bild 10: Die WLAN-Verbindung ist hergestellt, der Web-Server des Raspberry Pi quittiert dies, indem die Bildfläche des Handys grün gefärbt erscheint. Mit den Tastenflächen +/- ist die Helligkeit des Displays einstellbar.



die Starttaste verfügbar. Erfolgreiches Einloggen wird mit grünem Hintergrund quittiert (Bild 10). Manche Browser unterstützen auch eine Tastenvibration. Anmerkung der Redaktion: sehr nützliches Feature, das wir bei unserem Test mit einem LG Nexus nachvollziehen konnten. Die Vibrationsmeldung unterstützt das Finden der Tastenfelder auf dem Touchdisplay sehr, und nach kurzer Zeit hat man sich gut daran gewöhnt. Zusätzlich wird auch das Xbox-360-USB-Gamepad zur Steuerung unterstützt. Andere Gamepads sollten auch funktionieren – an der Konfigurationsmöglichkeit der Tasten wird jedoch noch gearbeitet. Nachsehen im Github [5] lohnt sich also, zumal sicher auch noch einige Spiele hinzukommen werden. Derzeit sind Tetris und Snake dabei.

Bild 11 zeigt noch einmal eine Ansicht des betriebsbereit aufgebauten Displays.

Die Software – Daniels Part

Entwickelt und getestet wurde mit einem Raspberry Pi B. Die beiden USB-Ports haben gerade gereicht, um ein USB-WLAN-Modul und den Diamex-Matrix-Player anzuschließen.

Als WLAN-Modul wurde TP-WN725N verwendet, welches auf dem rtl8188eu-Chip basiert und nach dem Start den Raspberry Pi in einen Access-Point mit dem Netz „Games“ verwandelt. Da hier ein sehr minimalistisches Linux-System verwendet wird, reicht auch eine kleine SD-Karte aus. Im Prinzip könnte alles auf eine 128-MB-Karte passen.

Als Betriebssystem kommt Linux zum Einsatz und zwar ein minimalistisches, zugeschnitten auf eigene Bedürfnisse. Buildroot passt für diesen Zweck sehr gut. Zusätzliche Anpassungen wurden dann an der Konfiguration gemacht, damit das WLAN aufgespannt wird und die „Game engine“ nach dem Booten automatisch gestartet wird. Dabei gibt es auch ein paar Hürden: der eingesetzte rtl8188eu-Chip funktioniert nicht auf Anhieb mit dem Standard Linux hostapd Daemon im Access-Point-Modus, daher musste dafür eine angepasste Version von hostapd gebaut werden. Bei anderen Chips kann dieser Schritt ggf. entfallen und hostapd.conf muss entsprechend angepasst werden.

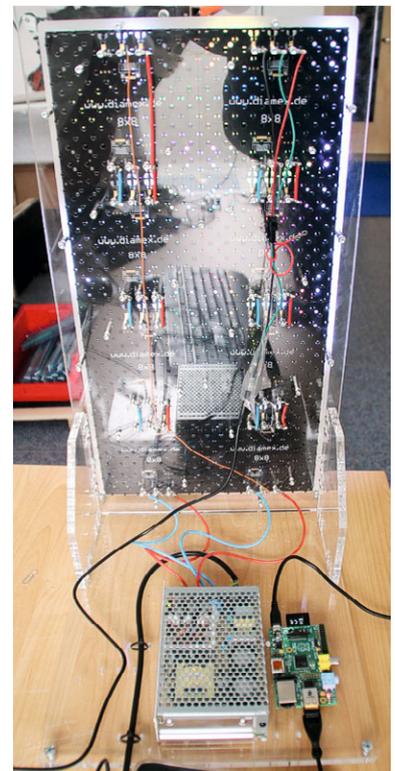


Bild 11: Das komplett mit Netzteil und Raspberry Pi aufgebaute Display

Kurz vorgestellt

LED-Hutschienenplayer

Will man eine große Anzahl von WS2812-LEDs und diese womöglich ortsfest betreiben, z. B. als Reklameschild, so bietet sich auch der stationäre Einbau der Steuerung an. Der neue Hutschienenplayer S erweist sich in vielerlei Hinsicht als Lösung. Er ist in der Funktion weitgehend kompatibel zum Matrix-Player, kann bis zu 512 LEDs ansteuern und besitzt einige Zusatzfunktionen, die den Einsatz noch vielseitiger machen. So ist über den mitgelieferten Busempfänger auch eine größere Leitungslänge zwischen Player und LEDs realisierbar. Der Busempfänger wird unmittelbar an der LED-Anordnung installiert, er sorgt für eine Aufbereitung des ankommenden Signals. Damit sind Leitungslängen bis über 100 m realisierbar. Schraubklemmen am Player und am Busempfänger machen die Installation einfach.

Die nächste Besonderheit ist die mögliche Konfiguration für die Ansteuerung von RGBW-LEDs, also z. B. SK6812.

Eine interne Testfunktion macht das Testen von LED-Anordnungen auch ohne PC-Anschluss möglich, dies ist z. B. beim Aufbau von Displays sehr hilfreich.



LED-Player-S
Best.-Nr.: CJ-12 56 57, € 39,90

Beispielanordnung mit einem im Hutschienengehäuse untergebrachten Raspberry Pi



Der Hutschienenplayer empfängt via USB serielle Daten im TPM2-Format und ist über eine Datei auf der einsteckbaren SD-Karte umfangreich konfigurierbar.



Bild 13: Variable Konfiguration macht es möglich: Das Tetris-Display ist auch als normales Anzeigedisplay einsetzbar. Hier unser Nachbau als Großuhr mit animiertem Hintergrund

Als Software für die Spiele und die Display-Ansteuerung wurde ein C++-Programm geschrieben. Nach der Erfahrung mit der Poco-Bibliothek bei der Portierung von hyperion für OpenWRT habe ich mich gleich dafür entschieden. Die Spiele sollten nicht fest kodiert werden und nach Möglichkeit erweiterbar sein, daher fiel die Wahl auf LUA – eine Programmiersprache, die oft im Embedded-Bereich verwendet wird, eine C-Anbindung sowie keine komplexe Abhängigkeiten hat und je nach Konfiguration auch kompakt ist. Ich habe versucht, die Software modular umzusetzen, sodass künftige Erweiterungen möglich sind. Die Erweiterung der LED-Ansteuerung um ein neues Protokoll sollte mit wenig Aufwand möglich sein – in der Hinsicht habe ich mich ebenfalls an hyperion gehalten. Hart kodiert ist aktuell die Displaygröße von 16×32 Pixel, das muss auf jeden Fall noch flexibler gemacht werden.

Die Spiele-Steuerung ist aktuell nach der Anmeldung im WLAN über die lokale Adresse des Systems (192.168.0.1) im Browser möglich. Ein auf dem Raspberry Pi laufender HTTP-Server stellt eine kleine Seite zur Verfügung, die über WebSockets mit der Spiele-Engine kommuniziert und die Button-Klicks an das LUA-Spiel weiterleitet. Auf diese Weise

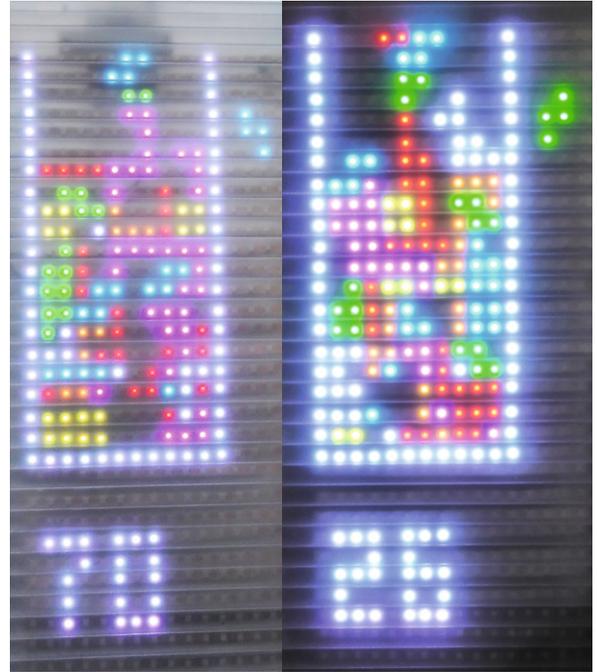


Bild 12: Das mit Doppelstegplatten und LED-Stripes bestückte Display unseres Nachbaus, links am Tage, rechts bei Dunkelheit

kann man die Spiele mit eigenem Handy als Gamepad ausprobieren, ohne Zugang zum Aufbau zu haben und ohne zusätzliche Software installieren zu müssen, wie es bei Bluetooth z. B. der Fall wäre.

Zum Kompilieren wurde eine Debian-Jessie-x64-Installation verwendet. GCC Compiler und andere build tools wie git müssen evtl. über apt-get nachinstalliert werden:

Die wichtigsten Parameter der Konfigurationsdatei

type = tpm2ser

Bestimmt den Typen des Ausgabegeräts/Protokolls. Für den Matrix-Player ist es das TPM2-Protokoll über die serielle Schnittstelle.

output = /dev/ttyACMO

Pfad zu dem seriellen Gerät im System.

baudrate = 115200

Baudrate für die Kommunikation mit dem seriellen Gerät.

color_order = grb

Farbenreihenfolge der LEDs. Bei WS2812-LEDs ist es grb.

origin = TopLeft

Legt die Position der ersten LED auf dem LED-Display fest. Kann Werte TopLeft, TopRight, BottomLeft oder BottomRight haben.

direction = Right

Legt die Flussrichtung der nachfolgenden LEDs fest. Kann Werte Right, Left, Down oder Up haben.

flow = Snake

Legt fest, wie die einzelnen Zeilen miteinander verbunden sind. Kann Werte Line oder Snake haben.

segment_width = 16

Falls das Display aus mehreren Segmenten besteht, die nicht in der Flußrichtung der LEDs miteinander verbunden sind, kann hier die Breite eines Segments festgelegt werden.

[settings]

brightness = 20

screen_width = 16

screen_height = 32

Diese Parameter legen die Breite und die Höhe des LED-Displays sowie die initiale Helligkeit der LEDs fest.

[games]

path = /boot/games

start = menu

Die Parameter der [games] Sektion legen den Pfad zu dem Ordner mit Spielen fest sowie das Spiel, welches beim Start ausgeführt werden soll. Der Wert „menu“ startet das Start-Menü, aus dem andere Spiele gestartet werden können. Die Spiele (Tetris und Snake) sind auf die Displaygröße 16×32 ausgelegt, für andere Größen müssen die Spiele evtl. angepasst werden.



```
git clone https://github.com/Gamadril/led-matrix-games.git
cd led-matrix-games
./build.sh
```

Der Buildprozess dauert sehr lange und generiert dabei einen Cross-Compiler, den Linux Kernel und das Root-Dateisystem. Gleichzeitig wird daraus ein fertiges SD-Karten-Image erzeugt, welches mit den üblichen Tools auf die SD-Karte installiert werden kann. Unter Linux wäre es

```
dd if=sdcard.img of=/dev/sdX
```

sdX muss durch den richtigen Gerätenamen des SD-Kartenlesers ersetzt werden. Das ist sehr wichtig, da man ansonsten auch die eigene Fest-

platte oder andere Datenträger wie z. B. USB-Sticks überschreiben kann.

Im Github befindet sich auch ein fertig erstelltes SD-Karten-Image, das man entweder wie oben aufgeführt unter Linux oder unter Windows per Win32DiskImager auf eine SD-Karte installieren kann. Im dort installierten Programmpaket findet man in der Boot-Partition die Datei *led_games.ini*, die man am PC oder direkt am Raspberry Pi (Login: root:games) bearbeiten und an die eigene LED-Matrix anpassen kann. Ebenfalls auf der Boot-Partition werden nach dem ersten Start die Spiele im „games“-Ordner abgelegt. In der Übersicht in [Tabelle 1](#) sind die entsprechenden Parameter zu sehen.

Anmerkung der Redaktion: Wir haben das Projekt in einer mechanisch einfacheren Version nachgebaut und dort WS2812-Stripes verwendet, die einfach ohne Raster in einer Doppelstegplatte platziert wurden. Durch den etwas größeren LED-Abstand auf den Stripen erwies sich auch diese einfache Lösung als durchaus spielbar, wenn man eine gewisse Entfernung zum Display einnimmt ([Bild 12](#)).

Das Display wird auf diversen Veranstaltungen hauptsächlich als Anzeigedisplay für Laufschriften und Lichtdekoration eingesetzt und dort quer betrieben. Möglich ist dieser flexible Betrieb durch die Konfigurationsmöglichkeiten der jeweiligen Software, z. B. Jinx!, die eine nahezu beliebige Lage des Displays zulassen. [Bild 13](#) zeigt das gleiche Display im Querformat im Betrieb unter Jinx! So kann das Tetris-Display quasi täglich nützlich eingesetzt werden.

Viel Spaß beim Nachbauen!



Weitere Infos:

- [1] Der Blog „LED’sWork“:
<http://www.ledswork.de/wp/2015/09/27/tetris-display-mit-handy-steuerung/>
- [2] Tetris-Projekt:
https://www.blafusel.de/misc/tetris_led-matrix_ws2812b.html
- [3] Wortuhr: <http://www.zabex.de>
- [4] Word Clock:
<http://www.fingers-welt.de/galerie/eigen/impro/klein2/klein2.htm>
- [5] Github zum Projekt:
<http://gamadril.github.io/led-matrix-games/>

Empfohlene Produkte	Benötigte Anzahl	Best.-Nr.	Einzelpreis
DIGI-DOT-Panel 8 x 8	8	CJ-12 09 65	€ 39,90
Matrix-Player für 1024 WS2812-LEDs	1	CJ-12 28 04	€ 19,90
Schaltnetzteil Mean Well RS 100-5, 5 V/16 A	1	CJ-11 90 27	€ 24,95
Raspberry Pi, Typ B	1	CJ-11 18 21	€ 34,95
Digitus-USB-WLAN-Adapter DN-70542	1	CJ-10 98 32	€ 9,95
Alternativ zum DIGI-DOT-Panel einsetzbar:			
4-m-LED-Streifen mit WS2812-LEDs, 60 LED/m	1	CJ-12 56 59	€ 59,95

Preisstellung August 2016 – aktuelle Preise im Web-Shop



Machen Sie mit!

Jede veröffentlichte Anwendung wird mit einem Warengutschein in Höhe von 200 Euro belohnt.

Wir wollen es wissen – Ihre Anwendungen und Applikationen!

Welche eigenen kreativen Anwendungen und Applikationen haben Sie mit den ELV-Haustechnik-Systemen, aber auch mit anderen Produkten und Bausätzen realisiert – ob mit Standard-Bausteinen oder eingebunden in eigene Applikationen? Alles, was nicht gegen Gesetze oder z. B. VDE-Vorschriften verstößt, ist interessant. Denn viele Applikationen verhalfen sicher anderen zum Aha-Erlebnis und zur eigenen Lösung.

Schreiben Sie uns, fotografieren Sie Ihre Applikation, berichten Sie von Ihren Erfahrungen und Lösungen. Die interessantesten Anwendungen werden redaktionell bearbeitet und im ELVjournal mit Nennung des Namens vorgestellt.



Per E-Mail
leserwettbewerb@elv.de



Per Post
ELV Elektronik AG, Leserwettbewerb, 26787 Leer

Die Auswahl der Veröffentlichungen wird allein durch die ELV-Redaktion ausschließlich nach Originalität, praktischem Nutzen und realisierter bzw. dokumentierter Ausführung vorgenommen, es besteht kein Anspruch auf Veröffentlichung, auch bei themengleichen Lösungen. **Der Rechtsweg ist ausgeschlossen.** Für Ansprüche Dritter, Beschädigung und Verlust der Einsendungen wird keine Haftung übernommen. Alle Rechte an Fotos, Unterlagen usw. müssen beim Einsender liegen. Die eingesandten Unterlagen und Aufnahmen verbleiben bei der ELV Elektronik AG und können von dieser für Veröffentlichungen und zu Werbezwecken genutzt werden.