



Build your Hive

## Inhaltsverzeichnis

Einführung.....	2
Hinweise.....	2
Teilbestückung und Variationen.....	4
Die Reihenfolge.....	4
Schaltplan / Übersicht.....	5
Schritt 1 – Stromversorgung.....	6
Bauelemente.....	6
Ablauf.....	7
Schritt 2 – Hostinterface.....	8
Bauelemente.....	8
Ablauf.....	9
Schritt 3 – Adapterkabel.....	10
Schritt 4 – Bellatrix.....	11
Bauteile.....	11
Ablauf.....	11
Schritt 5 – VGA Interface.....	14
Bauteile.....	14
Ablauf.....	14
Schritt 6 – TV und PS/2 Schnittstellen.....	16
Bauteile.....	18
Ablauf.....	18
Schritt 7 – Administra, Sound und SD-Card.....	20
Bauteile.....	21
Ablauf.....	22
Schritt 8 – Regnatix, Latch und SRAM.....	25
Bauteile.....	26
Ablauf.....	26
Schritt 9 – Netzwerkinterface (offen).....	29
Abschluß.....	32
Zusätzliche Schaltpläne.....	33

# Einführung

## Hinweise

1. Diese Anleitung sollte auch Einsteigern in der Materie dienen einen Hive zu bauen. Profis mögen deshalb über die selbstverständlich erscheinenden Hinweise hinweglesen und ihre eigenen Wege gehen.
2. Diese Anleitung wird sicher in einigen Punkten noch wachsen, wie das Projekt selbst. Also immer mal hier schauen was sich verändert. Wie es die Zeit erlaubt werde ich den einzelnen Schritten kleine Minitutorials zu den verschiedenen Themen (Grafik, Sound, Bus...) beifügen
3. Experimentiere! Ich habe den Hive innerhalb einer Woche ohne Schaltplan aufgebaut und dabei vieles davon gelernt was man über diesen speziellen Microcontroller lernen muß. Der Aufbau des Gerätes ist kein notwendiges Übel, sondern vermittelt im ständig möglichen Experiment alle Grundlagen, um mit dem Propeller und dem Gesamtsystem – dem Hive – umzugehen. Ich habe ein provisorisches Softwarepaket zusammengeschürt, welches in jeder einzelnen Phase eine kleine Möglichkeit bietet um dem Geschaffenen ein wenig "Leben" einzuhauchen. Spin und das Propellertool bietet dabei die Möglichkeit fast interaktiv (der Upload in das System dauert nur Augenblicke) zu experimentieren. Jeder Propeller-Chip ist in sich ein funktionierendes Sytem. So läuft der Testcode für die LAN-Schnittstelle einzig im Administra-Chip, ganz ohne das Mitwirken von Bellatrix oder Regnatix! Andererseits: Falls du ein neues Testprogramme hast, welches bei dir ein Problem beim Aufbau gelöst hat oder einfach nur cool ist, solltest du es uns zusenden, damit wir es in dem Testpaket allen zugänglich machen könne.
4. Hier die grundlegenden Unterlagen, Software und sonstige Dateien die für den Aufbau benötigt werden:
  - Testprogramme <sup>(1)</sup>: Eine Sammlung thematisch geordneter Codeschnipsel um den Hive stückweise in Betrieb zu nehmen, zu testen und zu experimentieren
  - Propeller Manual <sup>(2)</sup>: Die neuste Version ist als PDF frei bei Parallax verfügbar.
  - Software Propeller Tool <sup>(3)</sup>: Ist kostenfrei ebenfalls bei Parallax verfügbar.
  - Pinout <sup>(4)</sup> der drei Propellerchips im Hive
  - Schaltplan 1 <sup>(5)</sup> - Stromversorgung und Hostinterface
  - Schaltplan 2 <sup>(6)</sup> - Bellatrix und Administra
  - Schaltplan 3 <sup>(7)</sup> - Regnatix und externer SRAM

---

1 <http://hive-project.de/downloads/5>

2 <http://www.parallax.com/Portals/0/Downloads/docs/prod/prop/WebPM-v1.1.pdf>

3 <http://www.parallax.com/Portals/0/Downloads/sw/propeller/Setup-Propeller-Tool-v1.2.6.exe>

4 <http://hive-project.de/wp-content/gallery/prototyp-entwurf/pinout.jpg>

5 <http://hive-project.de/wp-content/gallery/prototyp-entwurf/hive-r13-plan1.jpg>

6 <http://hive-project.de/wp-content/gallery/prototyp-entwurf/hive-r13-plan2.jpg>

7 <http://hive-project.de/wp-content/gallery/prototyp-entwurf/hive-r13-plan3.jpg>

- Schaltplan 4 <sup>(8)</sup> - Ethernet-Controller
- Bestückungsplan <sup>(9)</sup>
- Schaltpläne und Layout <sup>(10)</sup> in hoher Auflösung
- Datenblatt 8bit-Latch <sup>(11)</sup>
- Datenblatt RAM 512K x 8Bit <sup>(12)</sup>
- Datenblatt EEPROM <sup>(13)</sup>
- Datenblatt Ethernet-Controller <sup>(14)</sup>
- Datenblatt Gehäuse <sup>(15)</sup>
- Datenblatt RS232-Driver <sup>(16)</sup>
- Datenblatt Propeller <sup>(17)</sup>
- Datenblatt Spannungsregler 5 V <sup>(18)</sup>
- Datenblatt Spannungsregler 3,3 V <sup>(19)</sup>
- Startdateien <sup>(20)</sup> - In diesem Paket sind nur die ganz grundlegenden Dateien enthalten um den Hive zu starten, keine Tools
- SDCard Mini <sup>(21)</sup> - oder Maxi (Diese Zusammenstellungen enthalten zusätzlich zu den nötigen Startdateien eine Zusammenstellung von Tools und Demos)

Weiter führende Informationen zum Thema Elektronik und zu Teilschaltungen:

Zum Thema Löten:

[http://www.youtube.com/watch?v=I\\_NU2ruzyc4](http://www.youtube.com/watch?v=I_NU2ruzyc4)

<http://www.youtube.com/watch?v=AgRo2IJS3Kk>

<http://www.elektronik-kompendium.de/sites/grd/0705261.htm>

Wichtig vor allem für gepolte Bauelemente wie Elektrolytkondensatoren, Dioden, LED's, Transistoren:

<http://www.elektronik-kompendium.de/sites/bau/index.htm>

---

8 <http://hive-project.de/wp-content/gallery/prototyp-entwurf/hive-r13-plan4.jpg>

9 <http://hive-project.de/wp-content/gallery/prototyp-entwurf/hive-r13-best.jpg>

10 <http://hive-project.de/downloads/8>

11 <http://hive-project.de/downloads/9>

12 <http://hive-project.de/downloads/10>

13 <http://hive-project.de/downloads/11>

14 <http://hive-project.de/downloads/12>

15 <http://hive-project.de/downloads/13>

16 <http://hive-project.de/downloads/14>

17 <http://hive-project.de/downloads/15>

18 <http://hive-project.de/downloads/16>

19 <http://hive-project.de/downloads/17>

20 <http://hive-project.de/downloads/3>

21 <http://hive-project.de/downloads/6>

## **Teilbestückung und Variationen**

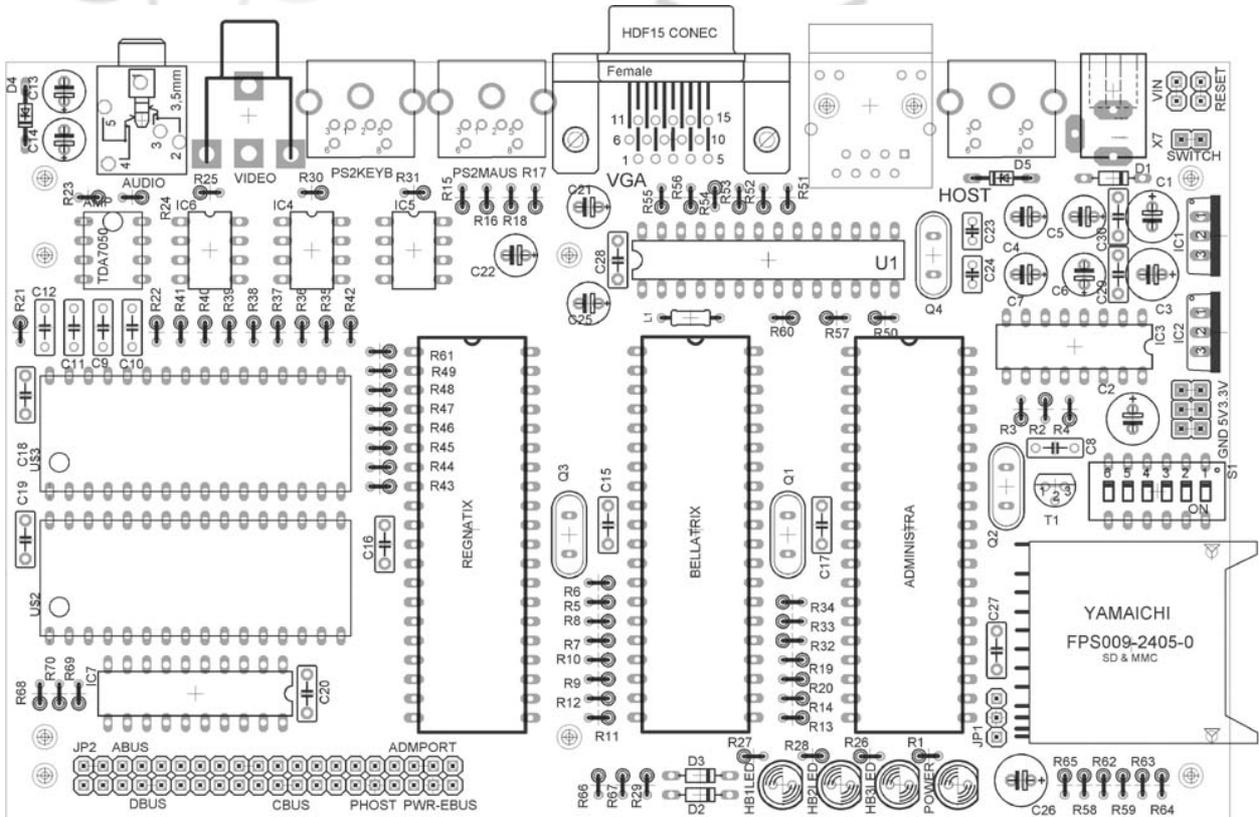
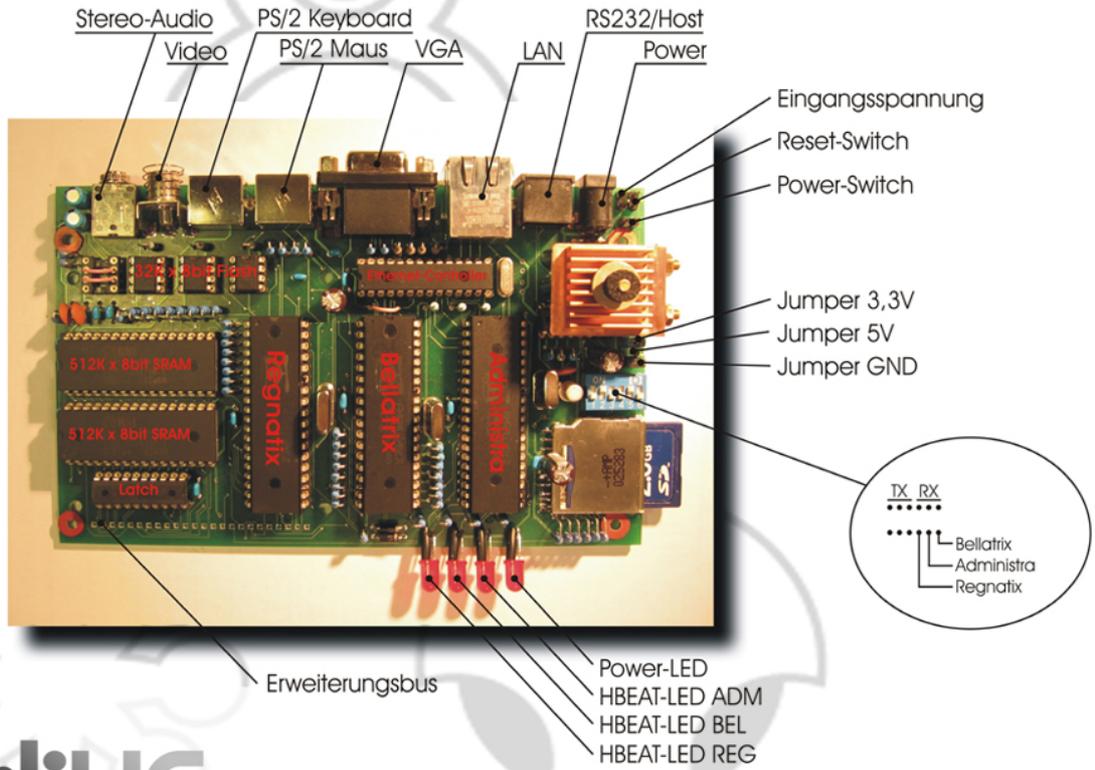
- Es ist möglich das Netzwerkinterface unbestückt zu lassen, sofern man kein Interesse an einer Ethernetschnittstelle hat. Vorteil: Fünf I/O-Ports an Administra werden frei für andere Hardware. Man kann diese Signale komfortabel abgreifen mit einer Stiftleiste welche man an Pin 4,6,7,8,9 statt dem ENC-Chip einlötet.
- Externes SD-Cardlaufwerk: Statt den Widerständen R58/59 und R62/63 ein Pinheader bestücken und die Signale mit einem Kabel abgreifen. Auf der externen Platine mit dem SD-Connector müssen dann natürlich obige Widerstände eingesetzt werden.
- LED's an Frontplatte: Wie beim Kartenlaufwerk auch die LED's nicht bestücken und mit einem Pinheader ersetzen, welcher über ein kleines Kabel die LED's versorgt.
- Mehr I/O-Ports: Wer für Experimente auf VGA, Video oder externen RAM verzichten kann, hat die Möglichkeit die entsprechenden Widerstände oder Schaltkreise unbestückt zu lassen. In die frei werdenden Lötungen kann man problemlos einen entsprechend breiten Pinheader einlöten um folgende Ports abzugreifen: VGA 8 Bit, Video 3 Bit, Externer Ram 11 Bits.
- Modulares System: Durch das Expansionsport ist es in verschiedener Weise möglich ein Backplane für mehrere Steckkarten anzuschließen. Da für sehr umfangreiche Erweiterungen die Stromversorgung auf dem Mainboard nicht reicht, sollte für die Erweiterungskarten eine eigene Stromversorgung realisiert werden. Diese kann auch problemlos das Mainboard mitspeisen, wenn man die Stromversorgung dort nicht bestückt, oder über die Jumper trennt.

## **Die Reihenfolge**

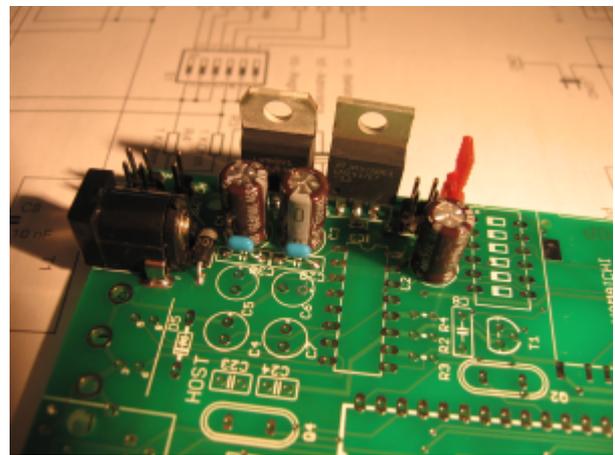
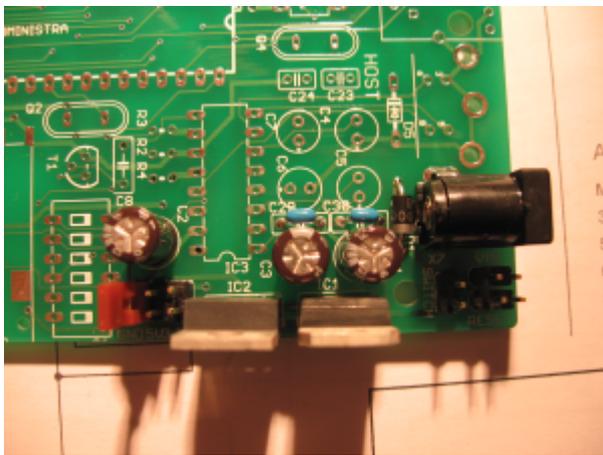
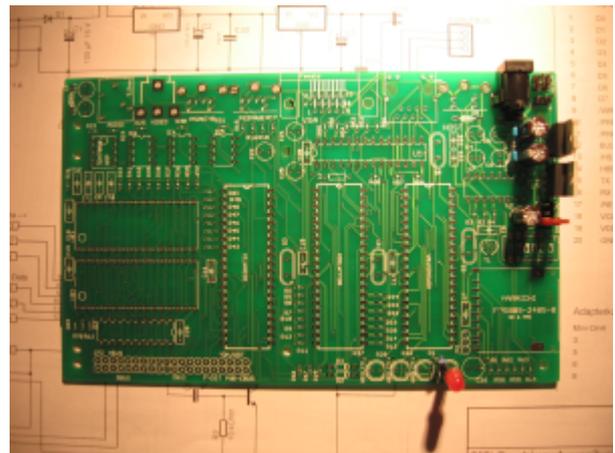
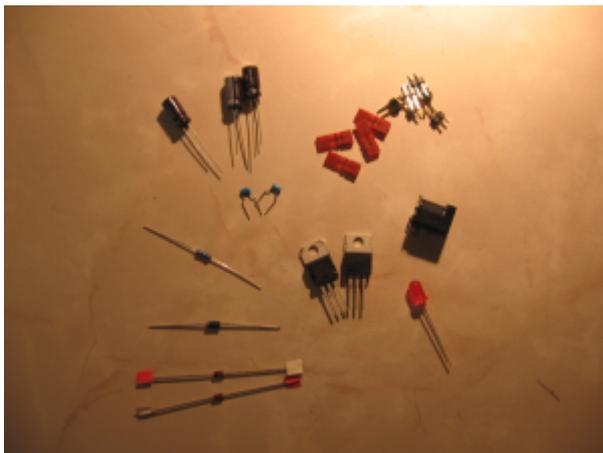
Es ist praktisch, einige flache Bauteile als erstes aufzulöten, um für diese Arbeit genug Baufreiheit zu haben. Dazu gehört als erstes der SD-Connector, welcher erst aufgeklebt und dann angelötet wird. Dieser Connector hat auch die kleinsten Löt pads, was Lötneulingen wahrscheinlich die meisten Schwierigkeiten bereiten wird. Als nächstes könnte man alle IC-Sockel auflöten, was sich auch gut macht ohne die restlichen Bauteile, da man dazu die Platine flach auf den Tisch legen kann.

Sinnvoll erscheint mir für den weiteren Aufbau folgende Reihenfolge: Bellatrix - Administra - Regnatix. Bellatrix macht den Anfang, da man mit diesem Chip sofort Display und Tastatur sowie grundlegende Terminalfunktionen zur Verfügung steht, welche mir für weitere Funktionen und Experimente wichtig erschienen. Administra macht sich akustisch bemerkbar - auch damit kann man problemlos experimentieren. Regnatix aber ist auf ihre Slaves angewiesen, benutzt Bellatrix (die Terminalfunktionen) um ihre Funktionen zu offenbaren und realisiert den Bus, um alle Teile zu einem Ganzen zusammenzufügen - deshalb wird sie in einem letzten Schritt in Betrieb genommen.

# Schaltplan / Übersicht



## Schritt 1 – Stromversorgung



### **Bauelemente**

- X7 Hohlbuchse
- D1 Diode (Belegung Anode/Katode beachten)
- C1..C3 Elektrolytkondensator (Polung beachten)
- C29, C30 Keramikkondensatoren
- IC1/2 Spannungsregler
- POWERLED Leuchtdiode (Polung beachten)
- R1 Vorwiderstand
- Jumper
- D4, D5 Z-Dioden

## **Ablauf**

Als Stromversorgung ist ein unstabiliertes Netzteil mit einer Ausgangsspannung von 6..9 Volt geeignet. Nach der Bestückung der Bauelemente ist als erstes am Pinhead für den Power-Switch mit einem Voltmeter zu überprüfen, ob die Eingangsspannung anliegt. Man kann nun einen Jumper oder gleich einen Schalter auf dem Pinheader "Switch" (dieser trennt die gesamte Schaltung von der Versorgungsspannung) aufstecken und an den entsprechenden unteren Pinheadern messen, ob die 3,3 bzw. die 5 Volt anliegt. Sollte eine der Spannungen nicht korrekt sein, muß die Bestückung genau überprüft und die Lötstellen genau auf Kurzschlüsse untersucht werden, um den Fehler zu finden. Die Power-Led sollte nun auch leuchten, und man sollte nicht vergessen sie zu bestücken für den Test: Damit die zwei Regler arbeiten können, muß es einen Verbraucher geben. Ohne die Power-LED fließt kein Strom und die Regler können auch nicht regeln, was zu dem Ergebnis führt, dass die Ausgangsspannung einen zu hohen Wert hat.

**ACHTUNG:** Eine fehlerhafte Spannung kann die gesamte Schaltung zerstören! Deshalb ist es wichtig, diese Arbeiten sehr gewissenhaft zu prüfen um Schäden zu vermeiden! Erst wenn die Versorgungsspannungen korrekt sind sollte auf den drei Pinheadern "3.3V", "5V" und "GND" je ein Jumper aufgesteckt werden. So wird die eigentliche Schaltung mit den zwei Versorgungsspannungen versorgt.

Nun ist der Zeitpunkt gekommen um die Z-Dioden D4 und D5 einzulöten. Diese dienen als Schutz der Schaltung vor Überspannung.

Zusätzliche Informationen gibt es zu Gleichrichterschaltungen <sup>(22)</sup>, Glättung und Siebung <sup>(23)</sup> und Integrierte Festspannungsregler (78xx/79xx) <sup>(24)</sup>.

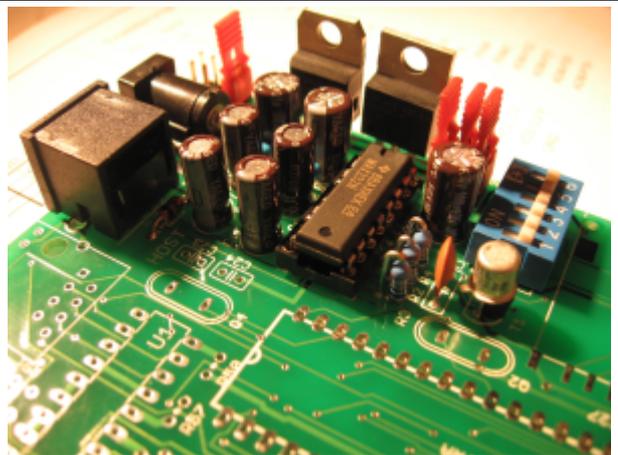
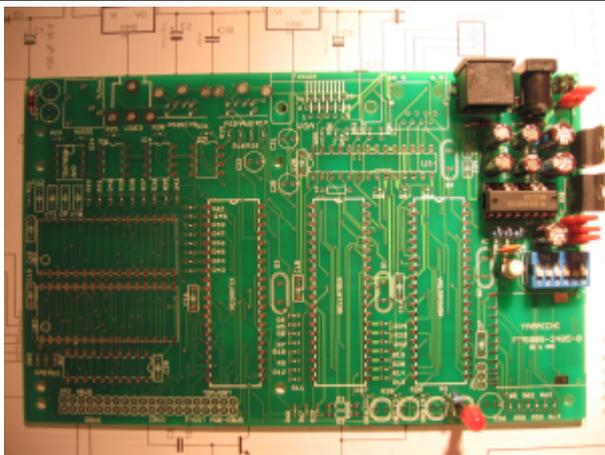
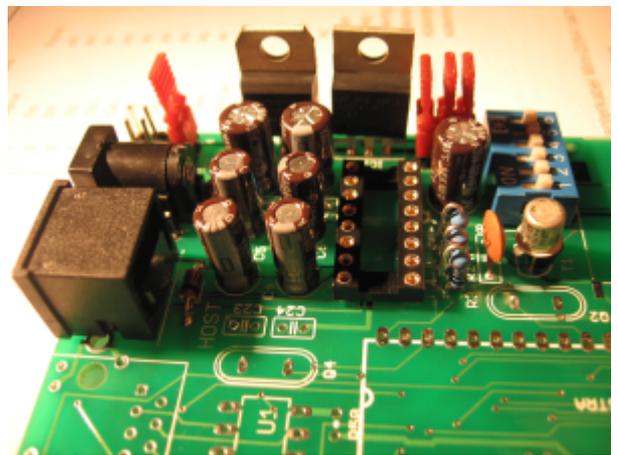
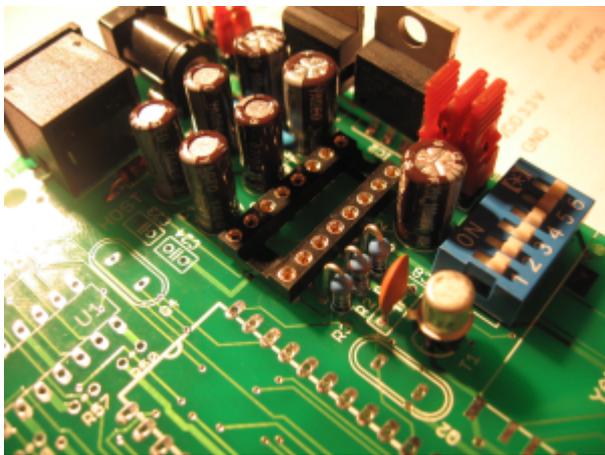
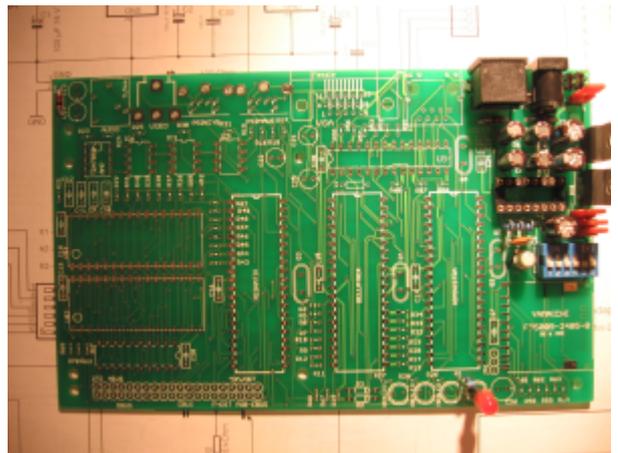
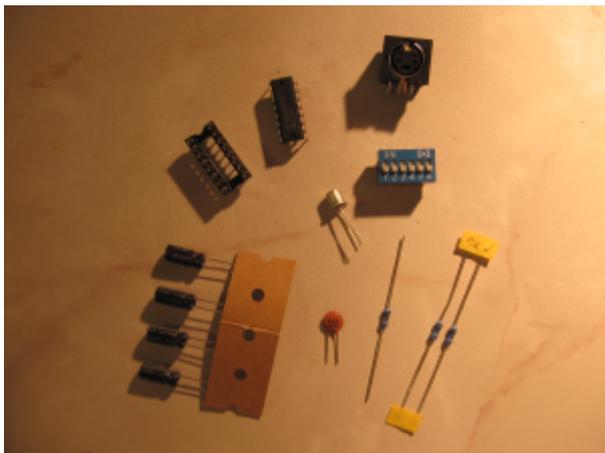
---

22 <http://www.elektronik-kompendium.de/sites/slt/0201071.htm>

23 <http://www.elektronik-kompendium.de/sites/slt/0210251.htm>

24 <http://www.elektronik-kompendium.de/sites/bau/0204301.htm>

## Schritt 2 – Hostinterface



### **Bauelemente**

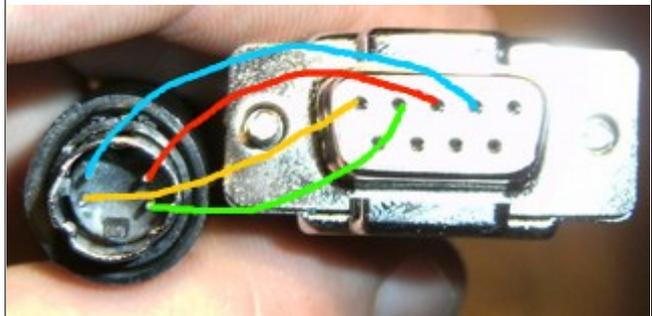
- Sockel für MAX232 (Pin1 beachten)
- C4..8 Kondensatoren (Polung beachten!)

- R2..4 Widerstände
- S1 DIP-Switch
- T1 Transistor (Belegung von Emitter/Basis/Collector beachten!)
- HOST MiniDin-Buchse
- IC3 MAX232 (Pin 1 beachten!)

### ***Ablauf***

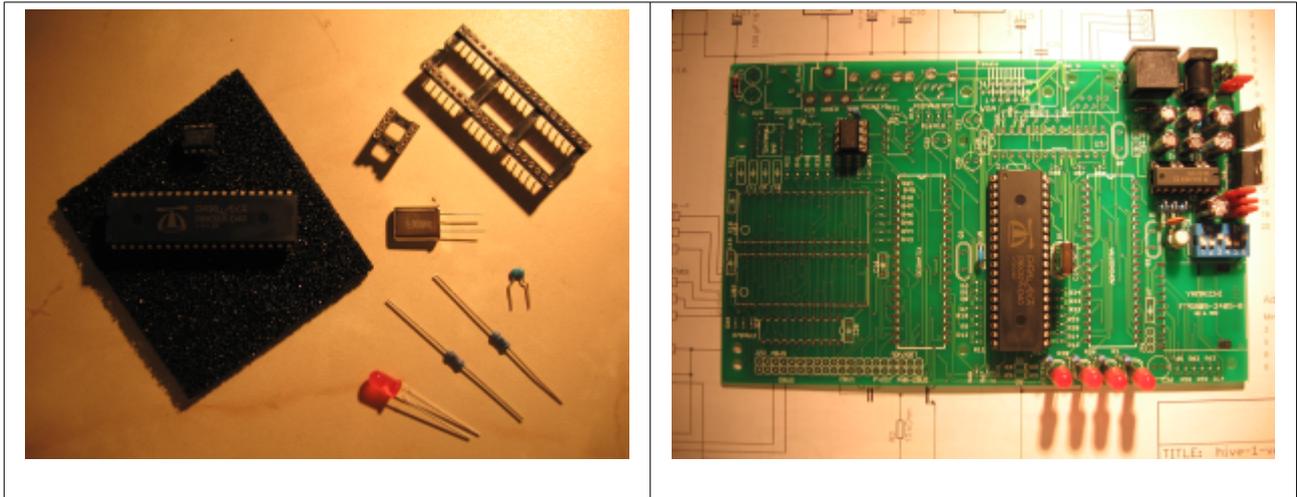
Bis auf den MAX232-Schaltkreis werden alle Bauteile bestückt. Als Test wird gemessen (ohne IC3 in den Sockel zu stecken) ob die Betriebsspannung korrekt anliegt. Zu beachten ist dabei, daß der MAX232 mit 5V Betriebsspannung läuft. Erst wenn dieser Test ok ist kann der Schaltkreis selbst bestückt werden. Damit ist die Versorgungsschaltung komplett.

### Schritt 3 – Adapterkabel



Die Belegung des Adapterkabels ist auf dem Schaltplan 1 verzeichnet.

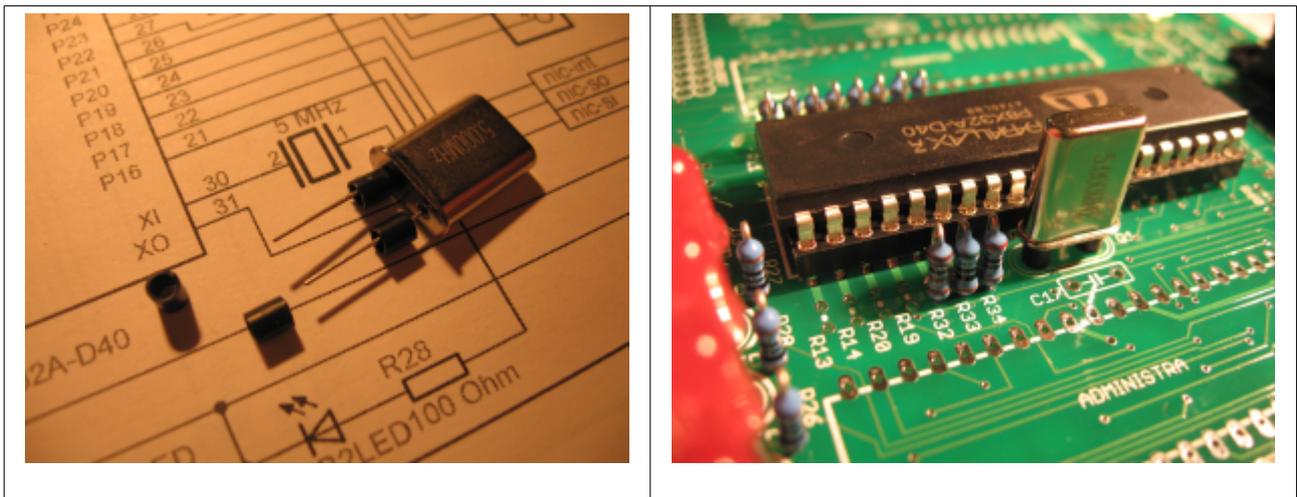
## Schritt 4 – Bellatrix



### Bauteile

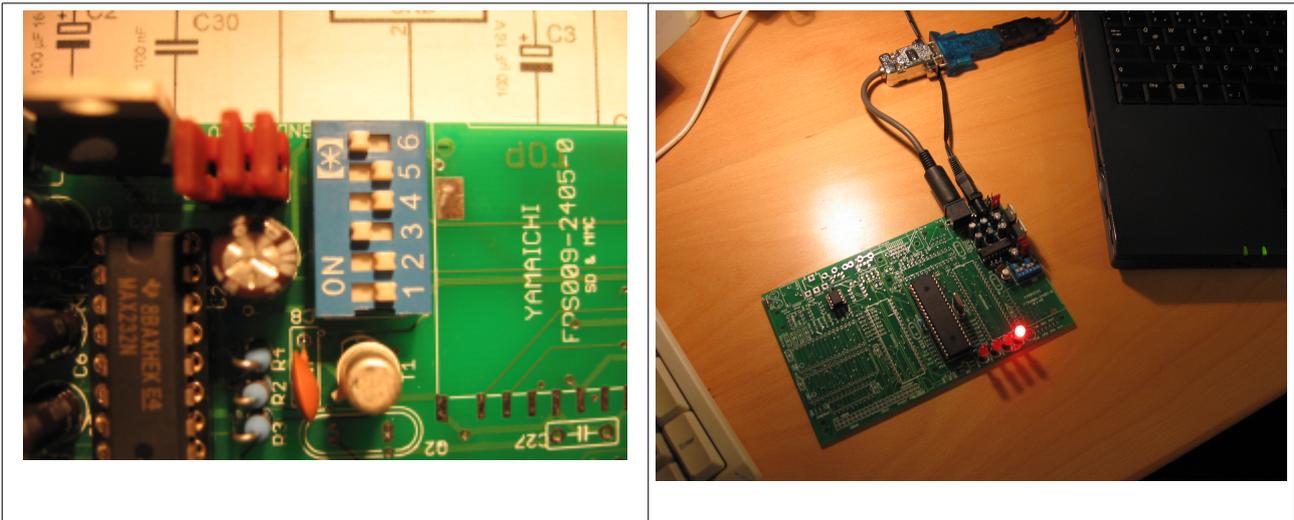
- 1 x 40pol Sockel für Bellatrix (Pin1 beachten)
- 1 x 8pol Sockel für die EEPROMs (Pin1 beachten)
- Q1 5 MHz Quarz
- Cx 100 nF Stützkondensator neben dem Sockel des Chips
- HB1LED Heartbeat-LED's (Polung beachten!)
- R26,R27, R28 Vorwiderstände
- R30 Pullup-Widerstand am EEPROM

### Ablauf



Achtung: Bei der Bestückung des Quarz sollte auf einen ausreichender Abstand des Metallgehäuses zur Leiterplatte geachtet werden, um evtl. Kurzschlüsse zu Durchkontaktierungen zu vermeiden. Ich habe dafür die Pins mit kurzen

Isolierschlauringen versehen, so das der Quarz etwa einen Millimeter über der Leiterplatte steht. Auf obigem Bild sollte man das erkennen können.



Nach der Bestückung sollte als erstes die Betriebsspannung an dem Sockel vom Propellerchip und an den Sockeln der EEPROMs überprüft werden. Beide Schaltkreise werden mit 3,3 Volt gespeist. Erst wenn die Speisespannung überprüft ist, sollten die beiden Schaltkreise gesteckt werden. Die EEPROMs in ihrer Anordnung entsprechen der Anordnung der Propellerchips, das heißt der Flash für Bellatrix ist der mittlere IC. Für den Test selbst ist aber der Flash nicht unbedingt nötig, aber wünschenswert, weil die Programme auch testweise in den hubRAM des Propellers übertragen werden können. Soll der Code dauerhaft gespeichert werden, ist natürlich letztlich der Flash erforderlich.



**Erster Test:** Das Propellertool starten und mit der Taste F7 überprüfen ob der Propellerchip erkannt wird. Dazu müssen die zwei DIP Schalter für den Bellatrix-Chip auf ON, alle anderen Schalter auf OFF geschaltet werden. Wenn das Tool eine entsprechende Meldung bringt, können wir sicher sein, daß die Versorgungsschaltung, das Hostinterface und das Adapterkabel funktioniert. Sollte eine Fehlermeldung erscheinen sind diese Schaltungen genau zu überprüfen. Zur genaueren Betrachtung der Funktion sind auch die Hinweise von Parallax zur Minimalbeschaltung <sup>(25)</sup> hilfreich. Das Signal DTR an der seriellen Schnittstelle löst dabei einen Reset aus um den Bootvorgang des Chips zu starten. In dieser Phase laufen folgende Vorgänge ab:

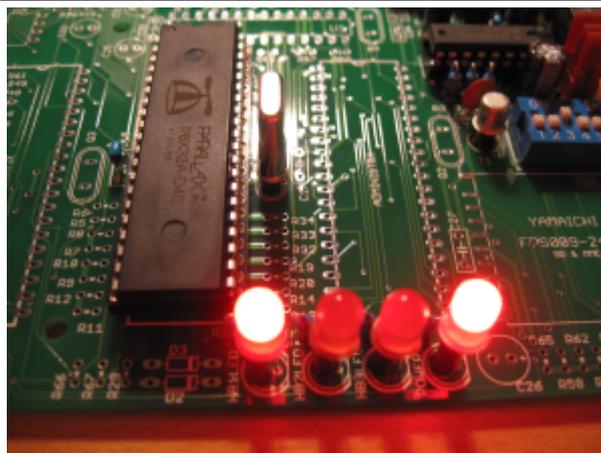
1. Der Propellerchip versucht an den Pins 30/31 eine serielle Verbindung zum Propeller Tool aufzubauen. Ist das erfolgreich wird ein 32kByte Image in den HubRAM geladen und entweder sofort gestartet (F10) oder erst geflashed und dann gestartet (F11).

<sup>25</sup> <http://www.parallax.com/tabid/254/Default.aspx>

2. Konnte keine serielle Verbindung etabliert werden, versucht der Chip über die Pins 28/29 einen EEPROM über I2C auf der untersten Adresse anzusprechen. Bei einem vorhandenen ROM wird nun ebenfalls ein 32K-Image in den HubRAM geladen und gestartet.
3. Sind die ersten beiden Phasen erfolglos verlaufen, taktet der Chip niedriger und geht in einen Shutdown-Modus.

Diesen Ablauf sollte man bei den Experimenten immer im Hinterkopf behalten. Zu beachten ist im Hive folgendes: Die entsprechende Verbindung der Pins 30/31 (RX/TX) sind über den DIP-Switch vom Hostinterface auf den entsprechenden Propellerchip schaltbar. Damit ist wählbar welcher Chip über das Interface mit dem Hostcomputer kommuniziert. Das Resetsignal aber ist für alle Propellerchips gleich.

Als erstes Testprogramm bietet sich "test hbeat-led.spin" an. Dieser Dreizeiler funktioniert übrigens mit jedem Propellerchip im Hive, da die LED's an den gleichen Pins angeschlossen sind. Oder man experimentiert selbst: Im Propeller Manual wird ab Kapitel 3 ganz wild mit LED's geblinkt und dabei die Grundlagen der Programmiersprache Spin erläutert. Im Pinout kann man schauen an welchem Port die LED angeschlossen ist. Mit F10 wird das Programm in den HubRAM des Propellerchips übertragen, F11 flasht es für einen dauerhaften Einsatz. Am besten man probiert beide Varianten, um gleich auch den EEPROM zu testen. Zeigt Bellatrix ihr Leben mit einem blinkenden Heartbeat-LED, an war der Aufbau bisher erfolgreich – das erste Subsystem ist also betriebsbereit!



### **Fehlersuche, wenn der Propellerchip nicht identifiziert wird:**

#### **Kabeltest:**

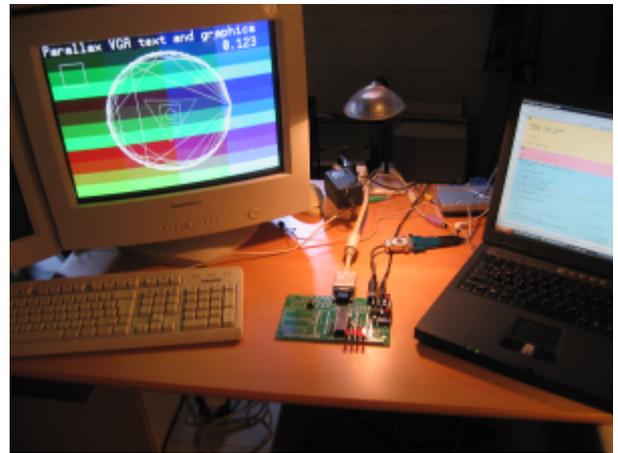
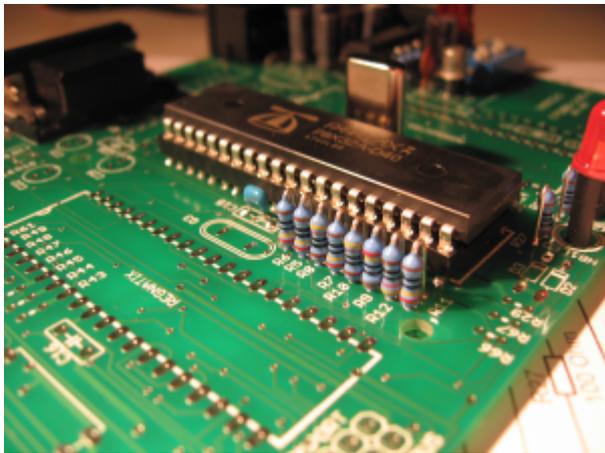
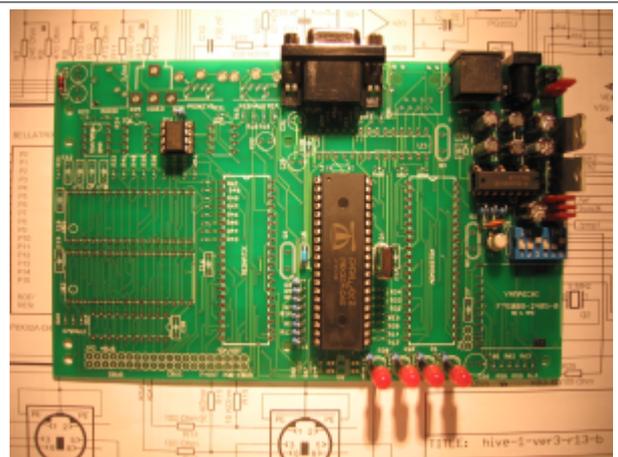
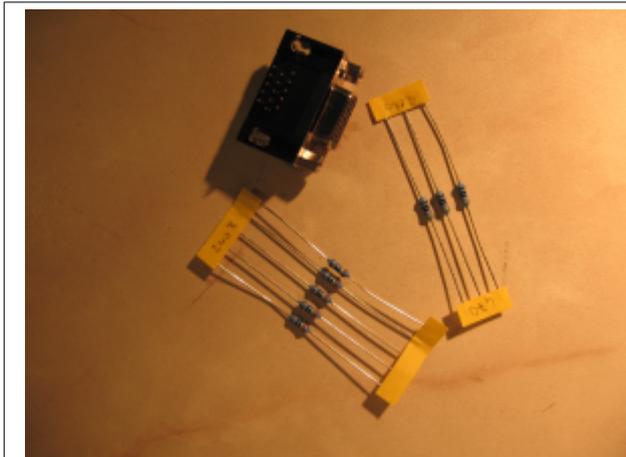
1. MAX232 und Bellatrix aus der Fassung ziehen.
2. Drahtbrücke im Sockel des MAX232 anbringen: max232:pin7 → max232:pin13 (TxD → RxD)
3. Mit einem Terminalprogramm kann man nun Zeichen senden, die durch den Loopback gleich wieder empfangen werden. Entfernt man die Brücke, werden keine Zeichen mehr empfangen.
4. Funktioniert das nicht, liegt der Fehler am MiniDin/DB9-Adapterkabel oder allgemein an der Verkablung zwischen Hive und Host.

#### **MAX232-Test:**

1. MAX232 bestücken, der Propellerchip bleibt draußen.

2. Drahtbrücke im Sockel von Bellatrix anbringen: bel:pin39 → bel:pin40 (Port 30 → 31)
3. Mit einem Terminalprogramm kann man nun wie beim Kabeltest Zeichen senden, die durch die Brücke gleich wieder zurück gesendet werden. Im Unterschied zum Kabeltest wird hierbei aber der MAX232-IC und die Verbindung bis zum Sockel des Propellerchips mitgetestet.
4. Funktioniert die Übertragung nicht, so findet sich der Fehler zwischen Propeller und Adapterkabel (sofern der Kabeltest erfolgreich war).

## Schritt 5 – VGA Interface



### **Bauteile**

- R5...R12
- VGA-Buchse

### **Ablauf**

So, jetzt wird es Zeit ein wenig Leben in die Bude zu bringen und dafür bedarf es beim Propeller keiner großen Anstrengung.

Bei der Bestückung der Widerstände muß peinlich auf die Werte geachtet werden, vertauscht man sie kann es zu Synchronisationsproblemen oder im lustigsten Fall zu abweichenden Farbwerten kommen. Zum experimentieren sind im Testpaket drei Beispielprogramme verfügbar, teilweise Demos von Parallax, welche an die Pinbelegung im Hive angepasst sind:

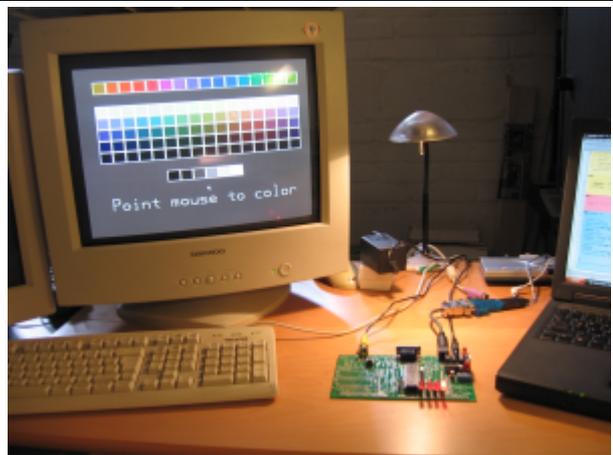
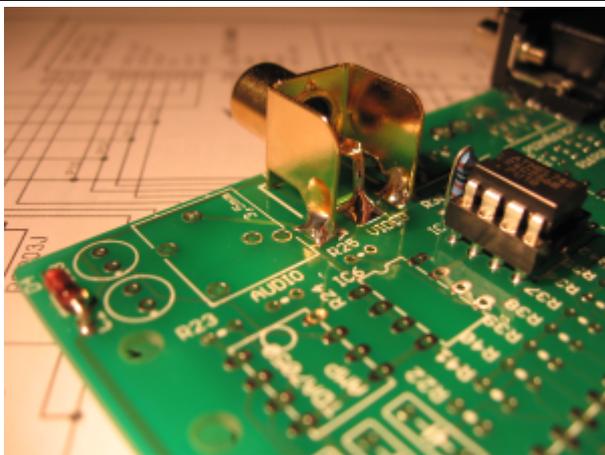
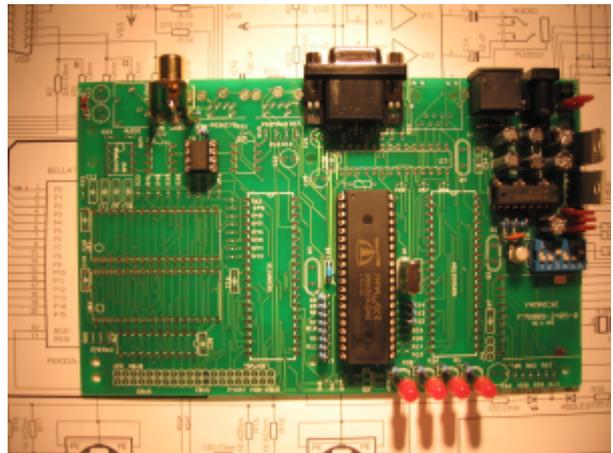
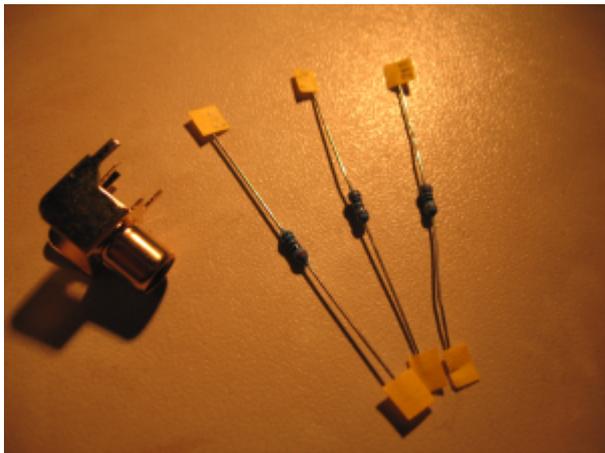
BEL\_VGA\_DEMO

VGA\_640×240\_BITMAP\_DEMO

VGA\_TILE\_DRIVE\_DEMO

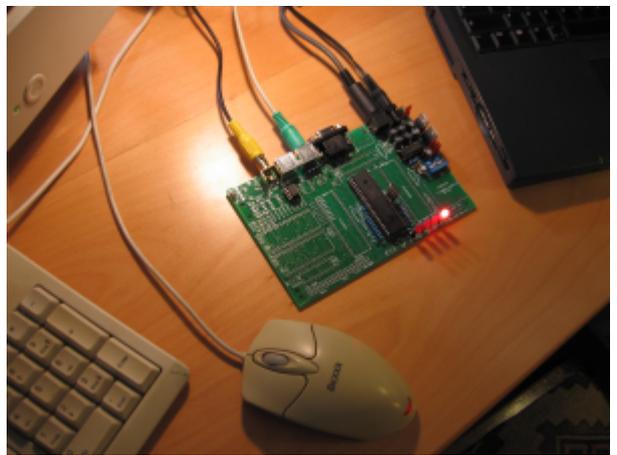
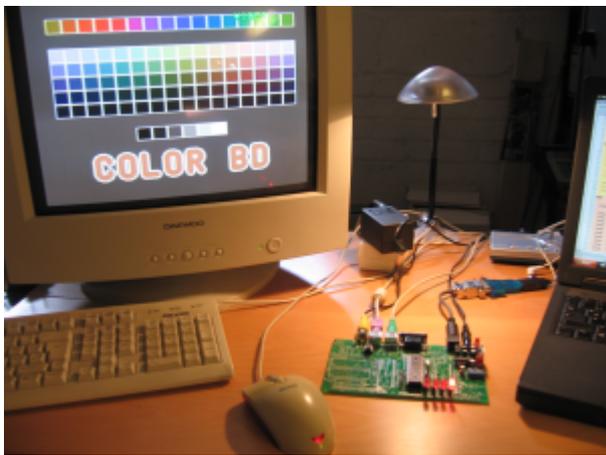
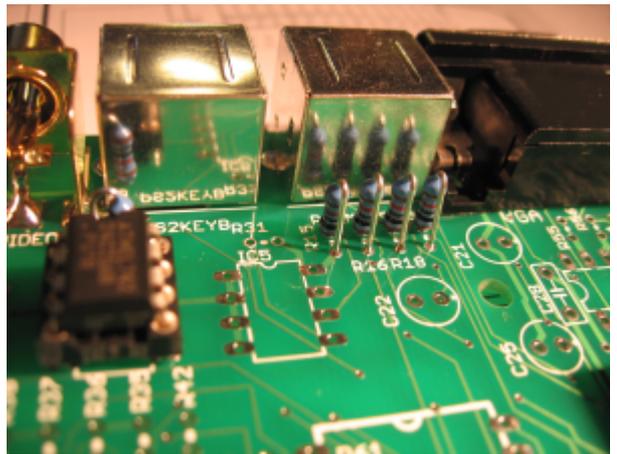
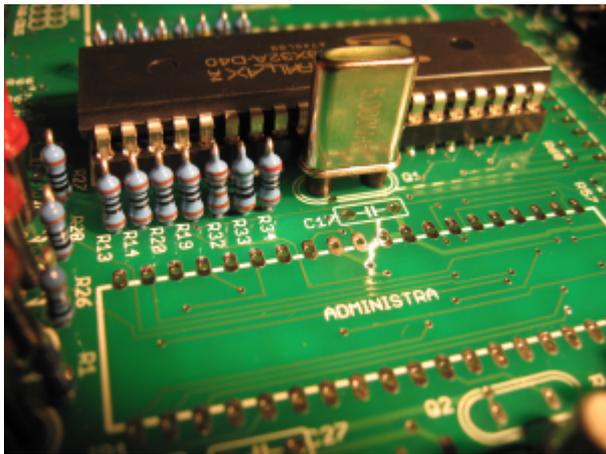
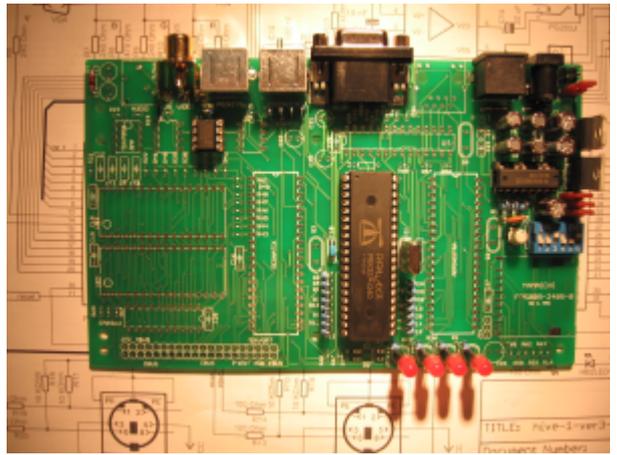
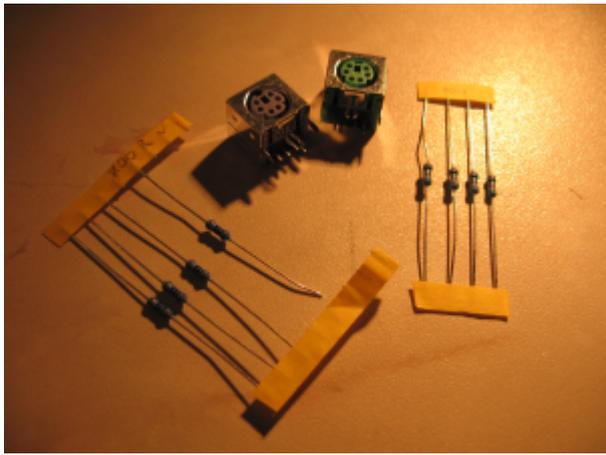
In den meisten Fällen bietet eine Auflösung von 1024 x 768 Pixeln eine gute Arbeitsgrundlage, es ist aber auch problemlos 1280 x 1024 Pixel im Textmodus möglich. Die Textmodis sind tilebasierte Treiber um einen möglichst kleinen Bildwiederholpeicher zu realisieren, Vollgrafik ist in diesen Auflösungen nur mit Bellatrix nicht möglich, bedingt durch den hohen Speicherverbrauch. Für schnelle Grafik und eine entsprechende Farbpalette ist der TV-Modus die erste Wahl, zum Arbeiten aber ist der VGA-Modus eine feine Sache. Ein netter einfacher Textmodus mit dem Feeling von C64 und Atari800, nur mit einer absolut sauberen und flimmerfreien Darstellung.

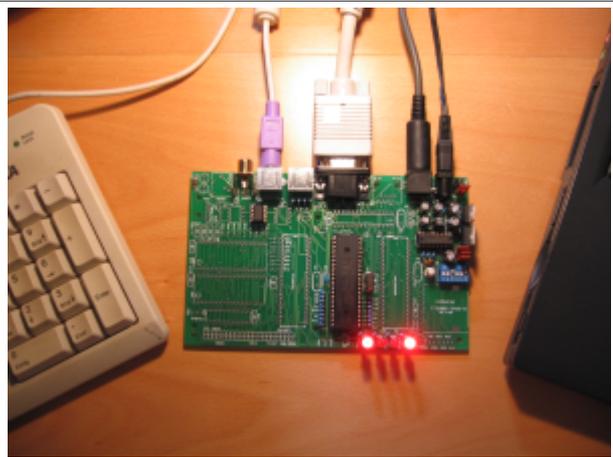
## Schritt 6 – TV und PS/2 Schnittstellen



Wer einen TV zur Verfügung hat sollte den Videoausgang testen. An diesem liefert der Propeller bei entsprechender Programmierung ein Composite-Video-Signal, welches man am Fernseher darstellen kann, sofern dieser über eine SCART-Buchse verfügt. Sollte dabei nur ein monochromes Bild erscheinen, liefert die Software ein NTSC-Signal. Die

meisten heutigen TV-Geräte können aber sowohl PAL als auch NTSC darstellen und erkennen das entsprechende Signal automatisch. Kann der Fernseher aber nur PAL, muß man den entsprechenden Modus im Quelltext des Demos umschalten. Wie man dabei erkennt, erfolgt die Grafikausgabe am Propeller fast vollständig durch Software, einzig ein 32Bit-Videoshifter bildet die minimalistische Hardware im Kern. Damit ist das System sehr flexibel: Was ich in welcher Bildschirmzeile wo, wie und wann darstelle, bestimmt einzig die Software die in einer oder mehreren Cog's läuft. So sind tile- und auch rasterbasierte Darstellungsarten, sowie jede denkbare Mischform realisierbar. Einziges Limit ist der verfügbare hubRAM, in welchem sich der Bildwiderholpeicher befindet. Sprites werden durch Cog's synchron zur Bildausgabe in Echtzeit überblendet und auch echtzeitorientierte Vektormodis ähnlich Vectrex wären denkbar. Für schnelle Darstellungen und viele Farben ist der TV-Videomodus gegenüber dem VGA-Modus die erste Wahl, schon allein bedingt durch den kleinen benötigten Bildschirmpuffer.





### **Bauteile**

- R32,33,34
- Chichbuchse
- R13..20
- PS/2-Buchsen

### **Ablauf**

Drei Widerstände, eine Buchse und schon spuckt Bellatrix Grafikdemos aus – also einfacher geht das nicht mehr. Die Testsoftware dazu:

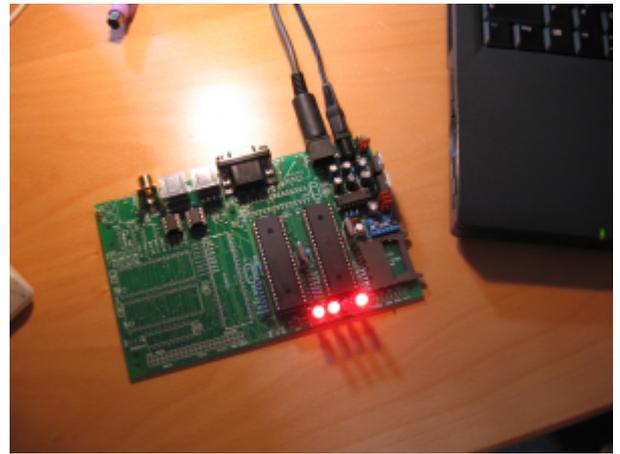
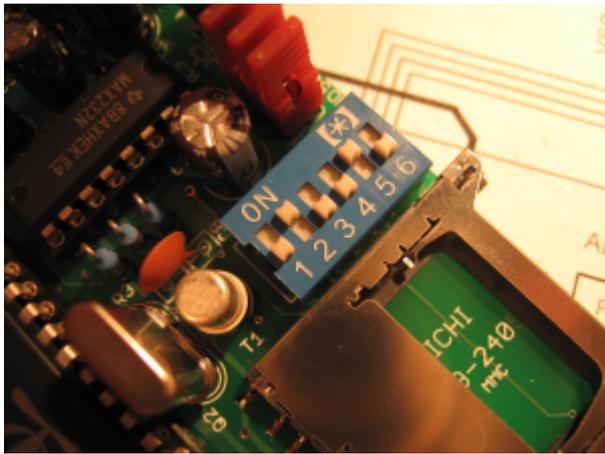
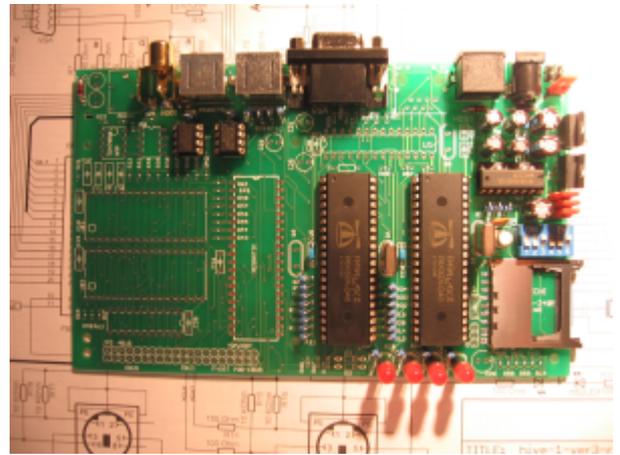
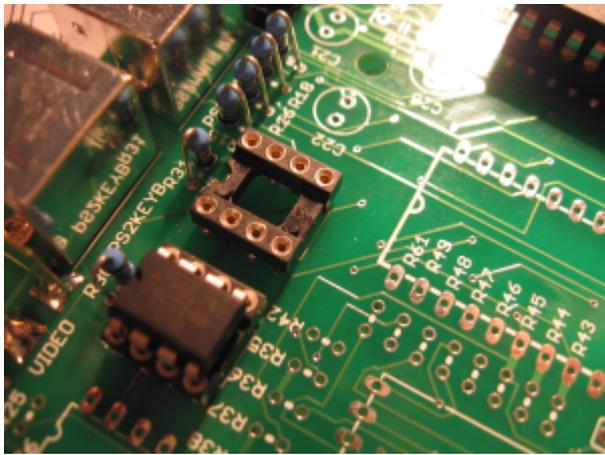
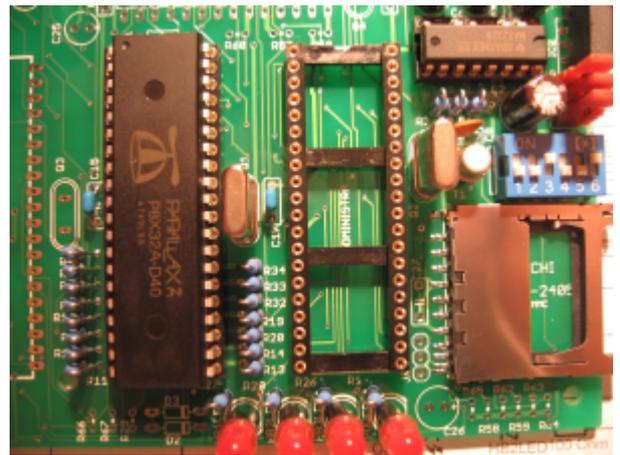
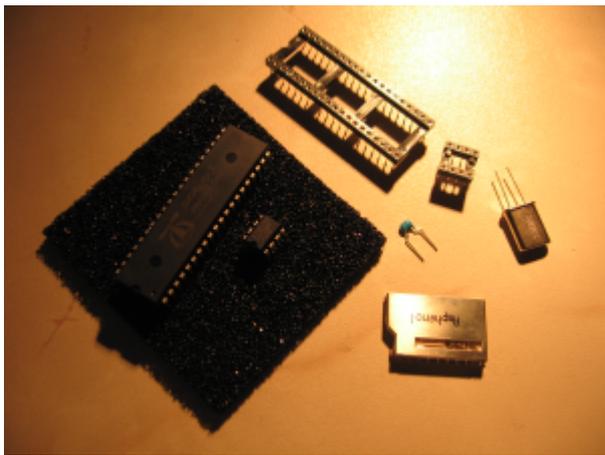
Graphics\_Demo\_Hive

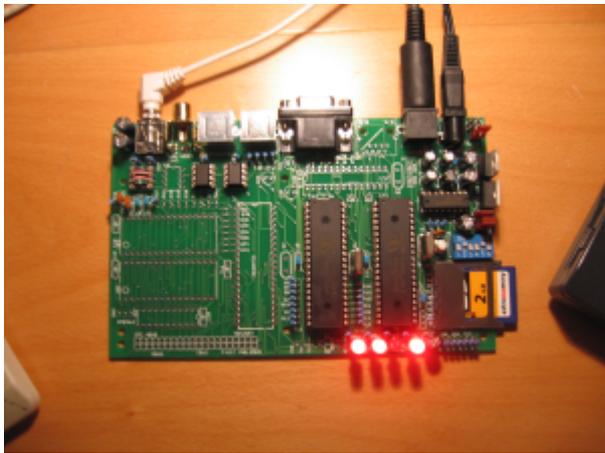
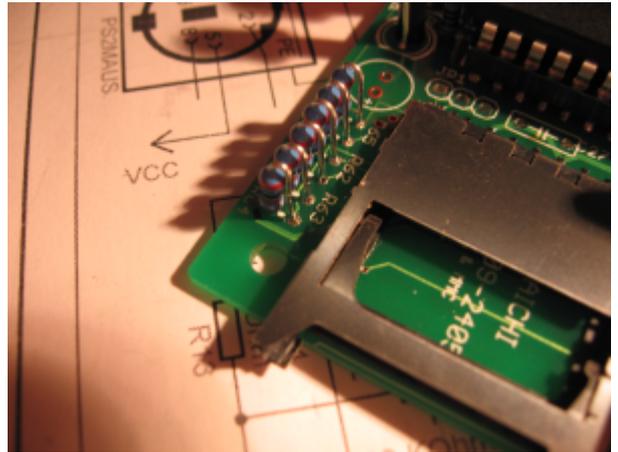
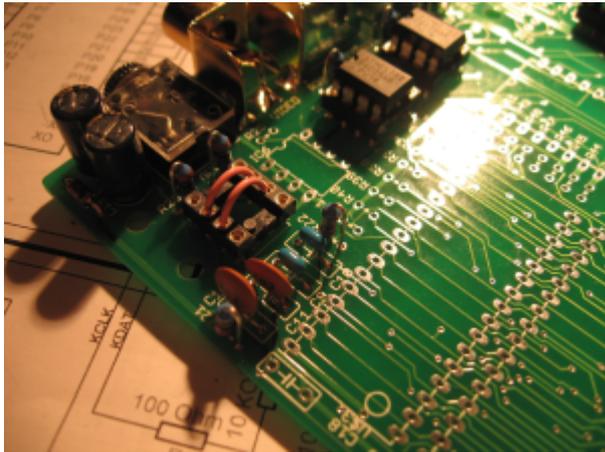
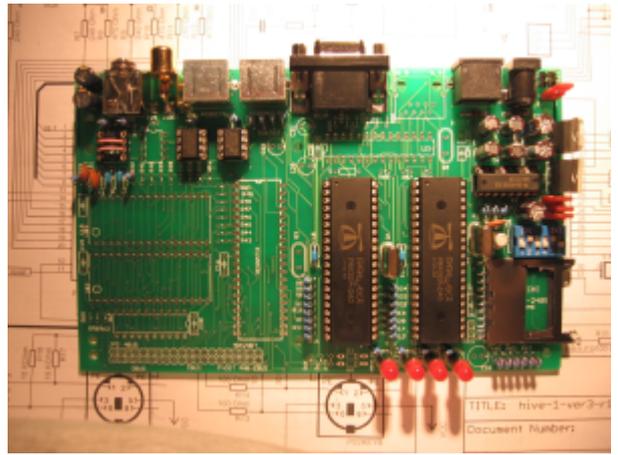
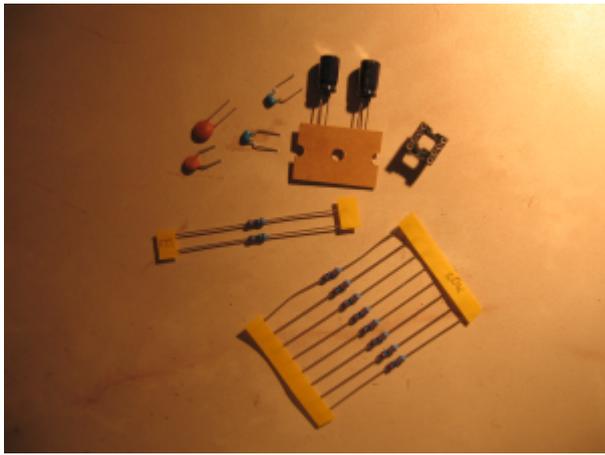
Graphics\_Palette\_Hive

TV\_Text\_Demo\_Hive

Die ersten beiden Programme sind auch geeignet um den Mausanschluß zu testen, da beide entsprechende Mausroutinen enthalten, aber dazu gleich mehr. Bei dieser Gelegenheit können auch die letzten Bauteile im Umkreis von Regnatix bestückt werden, um die PS/2-Schnittstellen in Betrieb zu nehmen. Wie schon erwähnt kann man ihre Funktion mit beiden Graphics-Demos testen, da diese eine Maus benutzen. Für den Test der Tastatur existiert ein Testprogramm im VGA-Modus. Damit ist der erste Propellerchip voll bestückt und wir können uns Administra zuwenden. Eine Zwischenbemerkung sei mir jedoch an dieser Stelle gestattet: Ein solches Tutorial zu schreiben dauert deutlich länger als einen Hive zu bauen und nur weil ich eine außerordentlich gehorsame Drohne bin ziehe ich das jetzt mal durch.

## Schritt 7 – Administra, Sound und SD-Card





### **Bauteile**

- 40pol Socket für Administra (Pin 1 beachten)
- 8pol Socket für TDA
- Klinkenbuchse
- R31, R21..24, R58, R59, R62..65

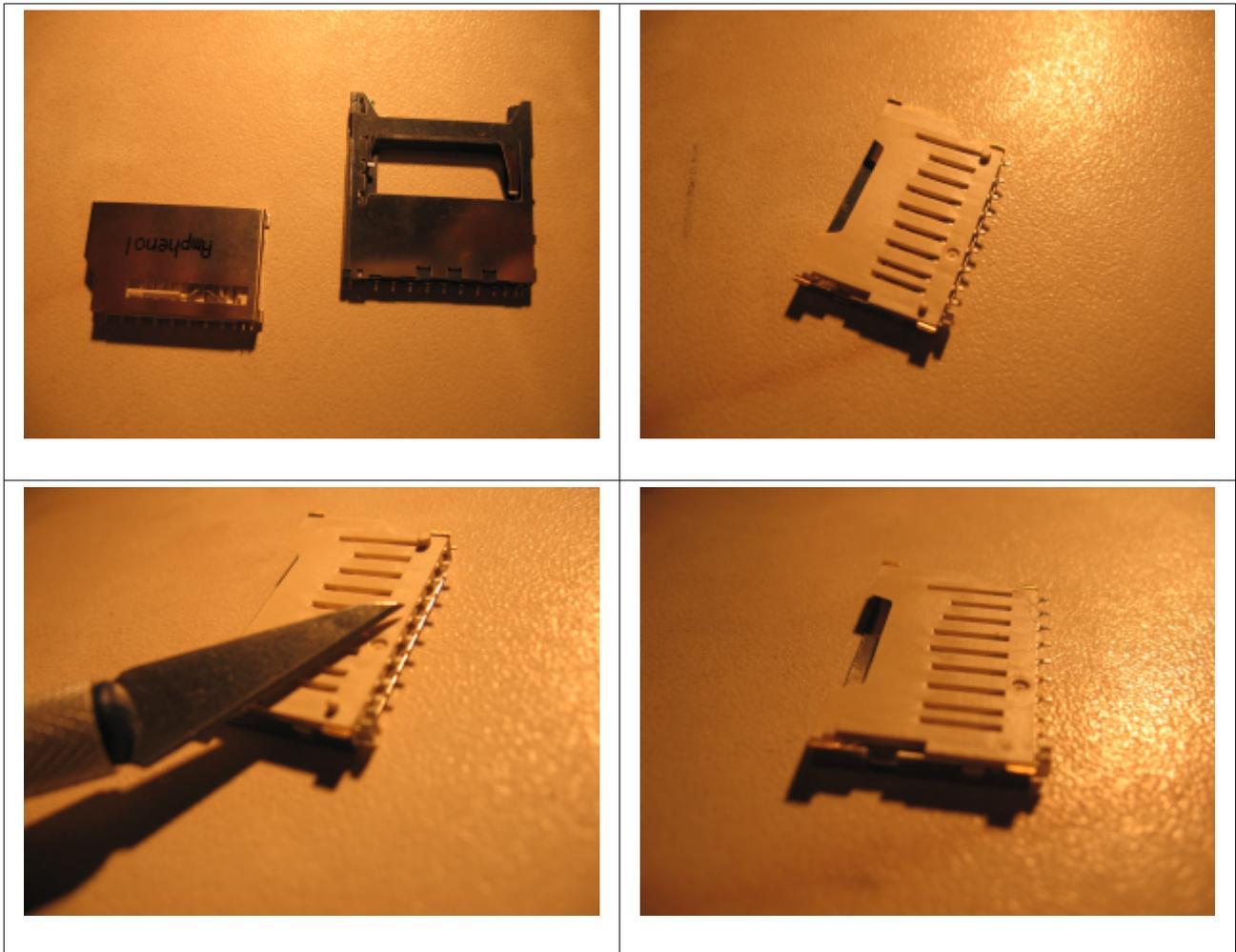
- C9..14 (Polung der Elektrolytkondensatoren beachten)
- C17
- Q2 Quarz

## **Ablauf**

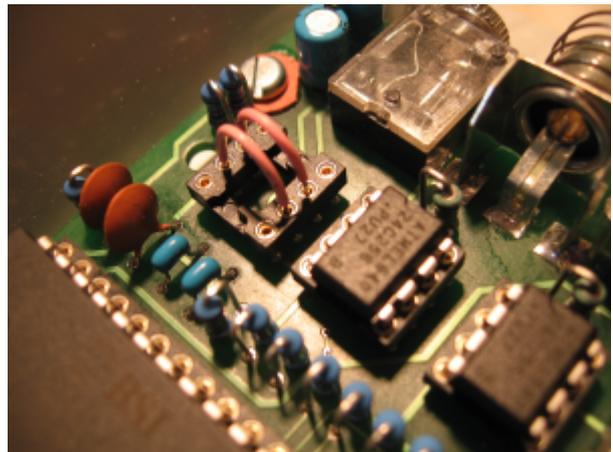
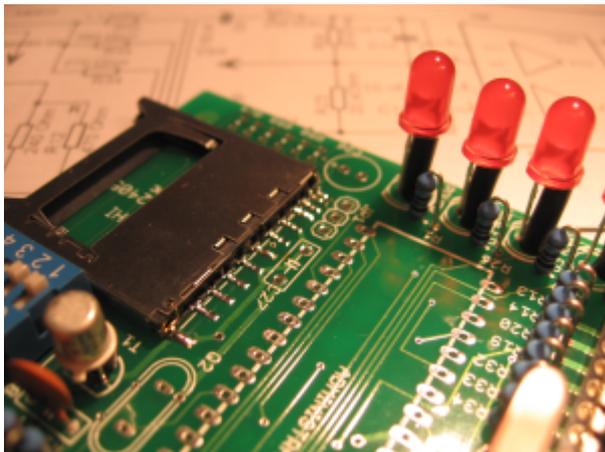
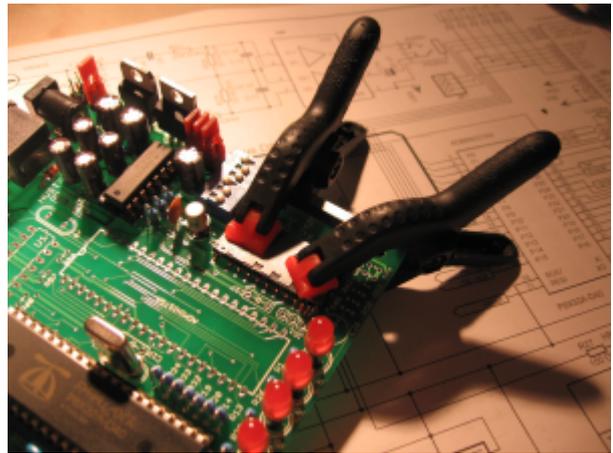
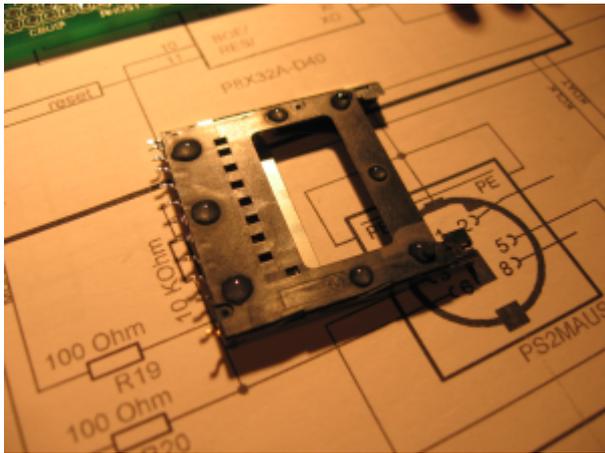
ACHTUNG: "Build your Hive – Audio-Patch" <sup>(26)</sup>

1. Kondensator C9 und C10 durch eine Brücke ersetzen.
2. Widerstand R21 und R22: 200 bis 300 Ohm

Die Bestückung von Administra gestaltet sich analog dem ersten Propellerchip und sollte nun schnell von der Hand gehen. Ebenso kann mit dem gleichen Testprogramm die Funktion des Propellerchips und vom Flash-ROM mit dem Heartbeat-LED überprüft werden. Nicht vergessen am DIP-Schalter die Einstellungen für Administra vorzunehmen, sonst gibt es keine oder eine Verbindung zum falschen Host! Für den SD-Connector gilt das gleiche wie für die Quarze: Bitte beachten das es durch Metallelemente keinen Kurzschluß gibt. Allerdings muß ich sagen, dass ich bei allen drei von mit bisher begutachteten Typen die Unterseite vollständig isolierend aus Plastik gestaltet war.



<sup>26</sup> <http://hive-project.de/build-your-hive-%E2%80%93-audio-patch>



Der SD-Connector wird mit Sekundenkleber auf der Leiterplatte fixiert, um die mechanischen Kräfte abzufangen. Es erscheint sinnvoll den Connector vor dem Lötens aufzukleben. Vorher sind die kleinen Plastiknippel an der Unterseite zu entfernen, sofern diese nicht in die entsprechenden Bohrungen passen. Beim Aufkleben muß aber sehr sorgfältig gearbeitet werden, die Pads müssen exakt sitzen und der Connector grade sein, da es gerade bei Sekundenkleber nach kürzester Zeit unmöglich ist das Bauteil noch zu verschieben. Ist die Klebestelle fest (für einige Minuten mit Klammern fixieren!) kann ganz entspannt gelötet werden. Wenn es sich um eine kurze Version handelt, so kann man überlegen auf den Connector eine Plastikführung zu kleben, wenn das ganze in einem Gehäuse eingebaut wird. So wird verhindert das die Karte beim stecken über dem Connector landet. Die langen Typen ragen zum Teil ein kleines Stück über den Platinenrand und bieten eine ausreichende Führung, allerdings sind sie auch deutlich teurer.

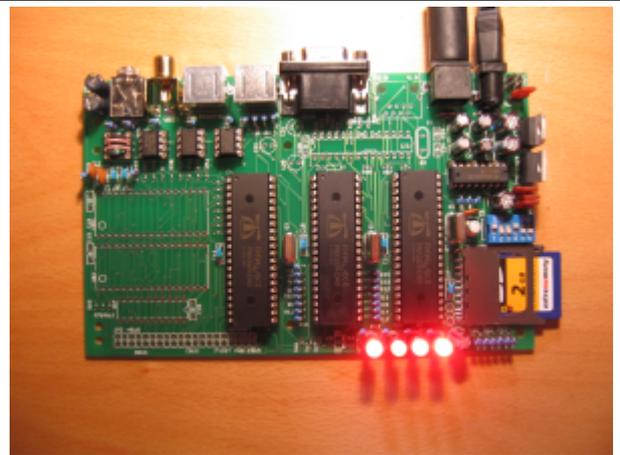
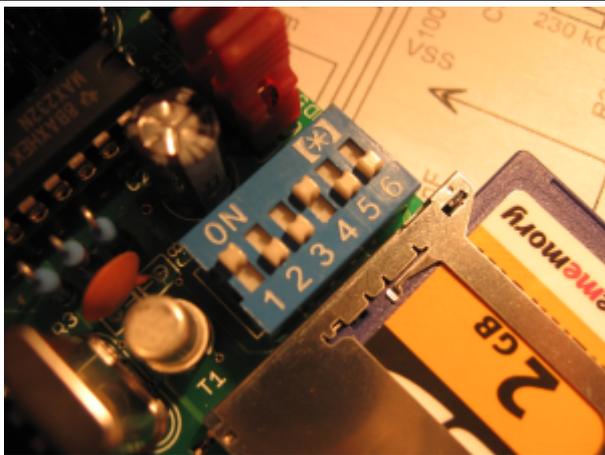
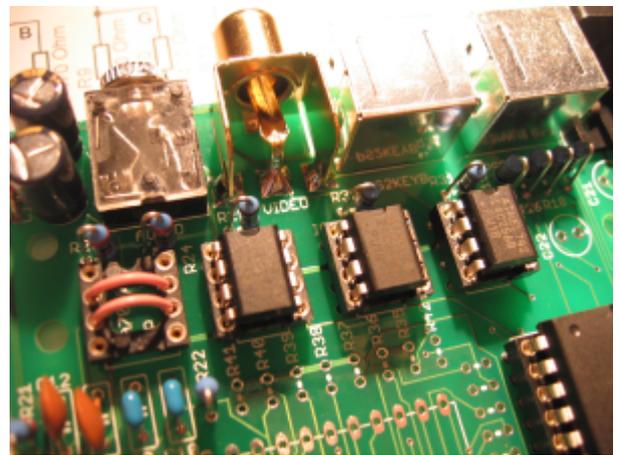
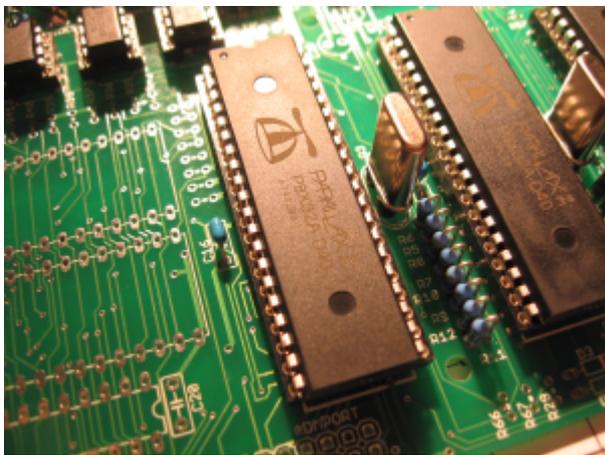
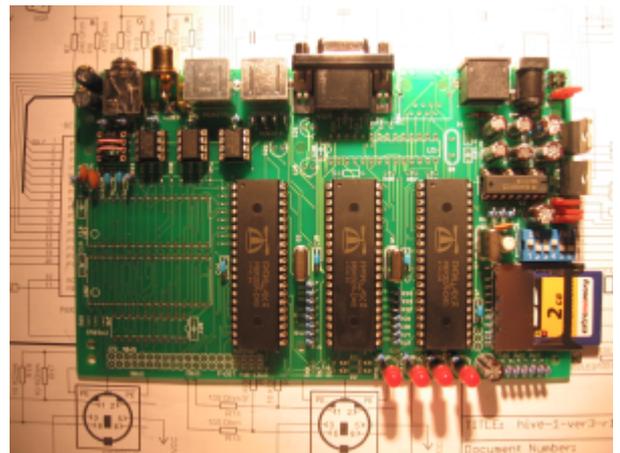
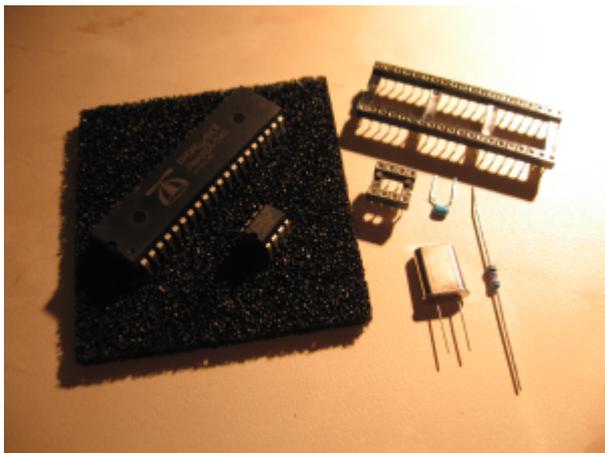
Auch die Anlogschaltung hat eine kleine Änderung erfahren: Der Verstärkerschaltkreis TDA7050 ist nicht nötig wenn man aktive Boxen anschließt, was in den meisten Fällen wohl so sein wird. Es genügen dabei zwei Drahtbrücken zwischen Pin 2/7 und 4/6 wie es in dem Bild zu erkennen ist. Wer möchte kann dennoch den Schaltkreissocket einlöten um eine spätere Änderung leichter zu gestalten, aber prinzipiell ist das nicht nötig. Bei Verwendung eines Sockels können die Drahtbrücken problemlos in diesen eingesteckt werden wie man sieht.

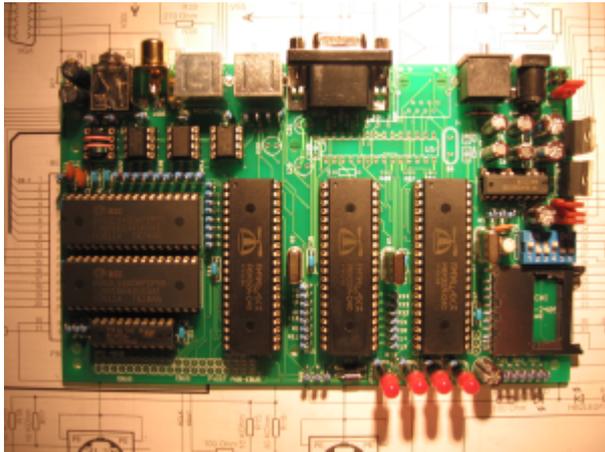
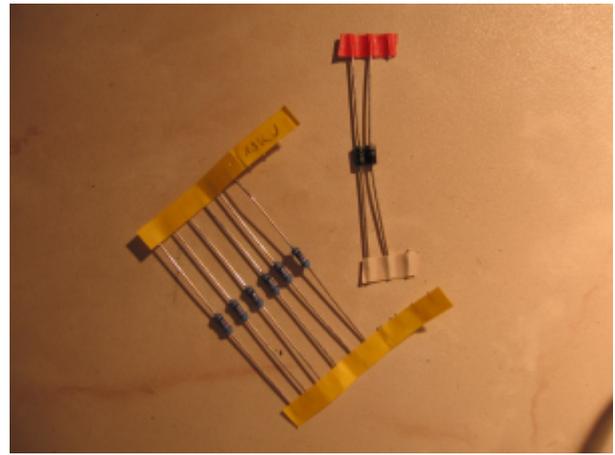
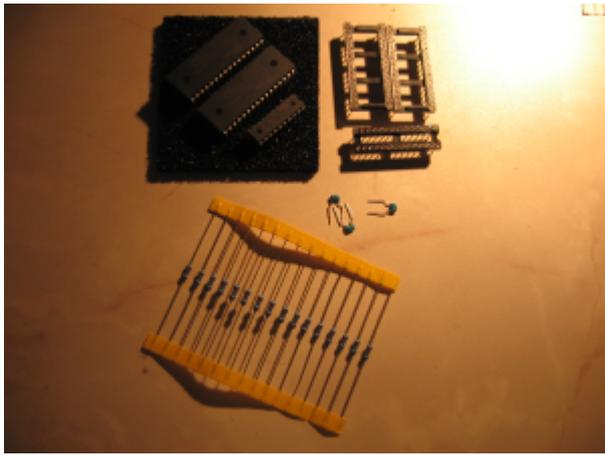
Sind diese Arbeiten erledigt kann der Test beginnen, Administra sollte ja prinzipiell schon laufen wie wir mit dem LED-Test überprüft haben.

soundtest 1 – Spielt eine HSS-Trackerdateien ab. Man kann im Quelltext die Datei auswählen (einfach auskommentieren) die der Player spielt. Dieser Test überprüft einzig die Soundhardware, da die MOD-Datei in den Datenblock compiliert wird – er wird in diesem Demo also nicht zur Laufzeit vom Laufwerk geladen.

midi\_player – Spielt der Reihe nach die Midi-Dateien von der SD-Card ab. Vorher müssen natürlich diese Dateien auf eine maximal 2 GB (FAT16) große SD-Card kopiert werden und diese muß natürlich auch im Connector stecken. Ertönen die Midi's, ist die Soundhardware und die Verbindung auf die SD-Card funktionsbereit.

## Schritt 8 – Regnatix, Latch und SRAM





## **Bauteile**

- 40pol. Sockel für Regnatix
- 20pol Sockel für Latch
- 2 x 32pol. Sockel für die Rams
- R35..50, R25, R29
- Q3 Quarz
- D2,D3 (Polung beachten)

## **Ablauf**

Die Bestückung und der Grundtest (LED) von Regnatix und den externen Ram's muß nun wie ein Kinderspiel erscheinen. Zu beachten ist wie immer das Gehäuse vom Quarz und die korrekte Ausrichtung der Sockelkennzeichnungen von Pin1 und natürlich der Schaltkreise selbst. Bei der Prüfung der Versorgungsspannungen an den Sockeln ist zu bedenken, dass sowohl Latch wie auch die beiden Speicherschaltkreise mit 5V gespeist werden. Der Test selbst aber gestaltet sich etwas komplizierter, denn nun geht es um das Zusammenspiel von mehreren Propellerchips. Im einfachsten Fall geht es erstmal nur um Bellatrix und Regnatix. Sollte für Administra noch Code von älteren Experimenten geflasht

sein, so empfiehlt es sich Administra komplett zu deaktivieren. Das kann man problemlos indem der code "prop-disable.spin" geflasht wird. Dabei reicht es aber nicht den Code in den Ram zu übertragen, da bei jedem weitem Übertragungsvorgang zu den anderen Propellern ein Reset ausgeführt wird und somit der Inhalt des RAM hinfällig ist – also ab jetzt zum testen immer flashen!

Im ersten Fall beschränken wir uns auf Bellatrix und Regnatix. Dabei handelt es sich um das Paket "test regnatix bus1"; in diesem Verzeichnis befindet sich eine entsprechende Datei "bellatrix.spin" und "regnatix.spin". Beide Dateien öffnen wir also im Propellertool und flashen sie in den entsprechenden Propellerchip. Dabei die Umschaltung am DIP nicht vergessen, sonst gibt es Chaos. Die Datei für Bellatrix enthält einen normalen textorientierten Grafiktreiber, welcher grundlegende Terminaldienste auf dem Bus zur Verfügung stellt (Zeichen ausgeben, Tastatur abfragen...). Der Regnatix-Code ist letztlich einfach der minimalistische Kommandozeileninterpreter "Regime", welcher auch später noch eine Rolle spielen wird. Normalerweise startet/bootet man diesen von der SD-Card, aber für unsere momentanen Tests haben wir ja Administra disabled. Das System sollte also in folgender weise geflasht sein:

Administra <— "prop\_disable.spin"

Bellatrix <— "bellatrix.spin"

Regnatix <— "regnatix.spin" (enthält regime.spin)

Nun sollte sich das System nach einem Reset mit der Ausschrift Regime, einer Versionsnummer und einem Prompt melden. Tastatur und Bildschirmausgabe müssen funktionieren, z.b. das Kommando "help". Treten keine Komplikationen auf, ist der Test eigentlich erfolgreich abgeschlossen. Daten können zwischen Regnatix und Bellatrix ausgetauscht werden, der Bus funktioniert und alle damit zusammenhängenden Signale funktionieren. Im konkreten Fall fragt Regnatix über den Bus die Tastaturroutine in Bellatrix ab, wird eine Taste gedrückt landet der entsprechende Tastencode in einem Eingabepuffer in Regnatix und das Zeichen wird folgend über den Bus zu Bellatrix gesendet und auf dem Bildschirm ausgegeben. Man kann nun noch ein wenig mit Regime und/oder seinem Quelltext experimentieren, aber die meisten Kommandos werden nicht funktionieren, da diese auf die SD-Card (hängt an Administra die wir momentan abgeschaltet haben) zugreifen wollen. Bleibt der Kommandozeileninterpreter durch einem Laufwerkszugriff hängen, kann man ihn mit einem Reset wieder zum Leben erwecken da er ja im Flash ist.

Nun kommt der RAM an die Reihe: Für diesen Test müssen wir nichts an Administra und Bellatrix ändern. In dem Paket "regnatix test ram" befinden sich zwei Tools:

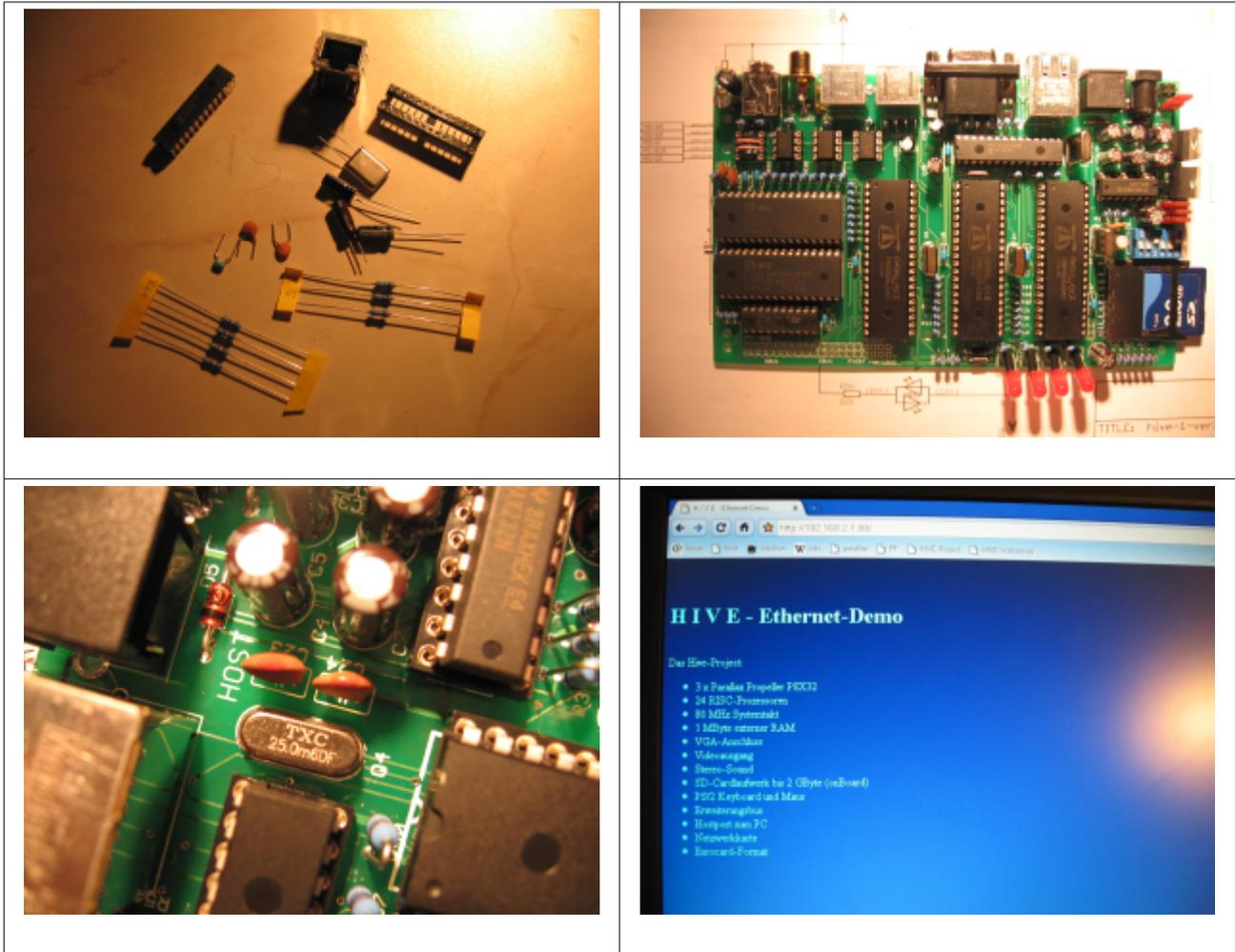
rtools.spin – Ein kleiner RAM-Monitor mit einigen Kommandos um den Speicher anzuzeigen oder zu editieren. Mit "help" gibt es eine Liste der Befehle.

ramtest.spin – Ein kleiner automatischer Ramtest. Nach einer kurzen Sicherheitsabfrage beginnt dieses Programm beide Bänke mit Zufallszahlen zu beschreiben und danach zu vergleichen. Bei einer Differenz gibt das Programm eine Fehlercode aus.

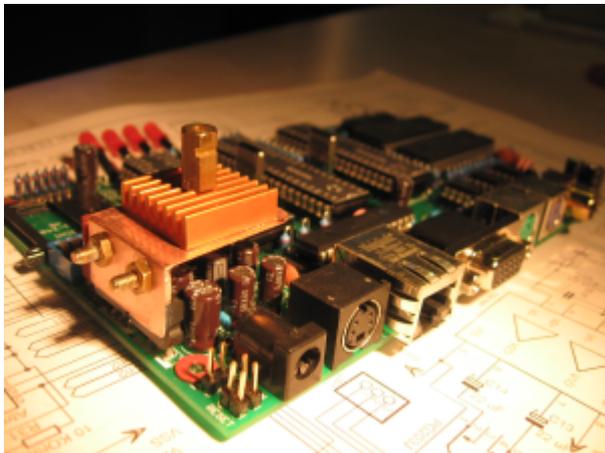
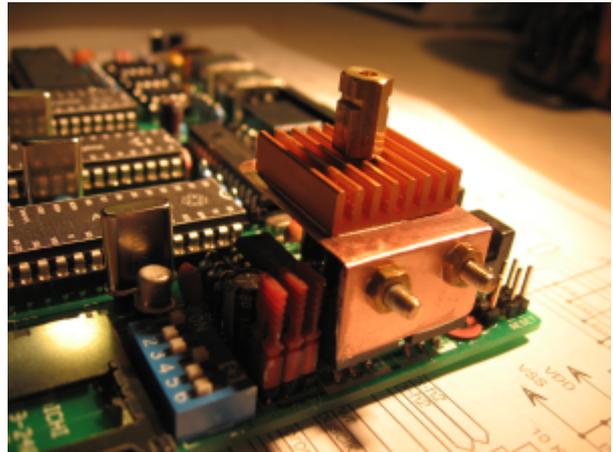
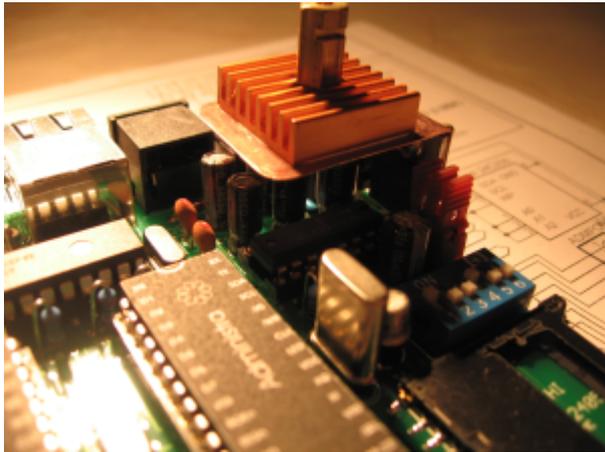
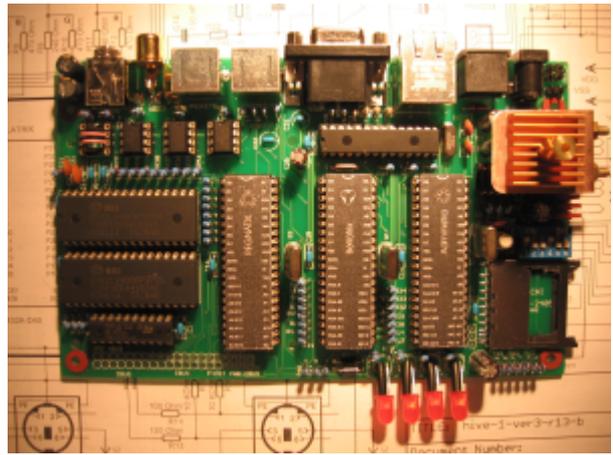
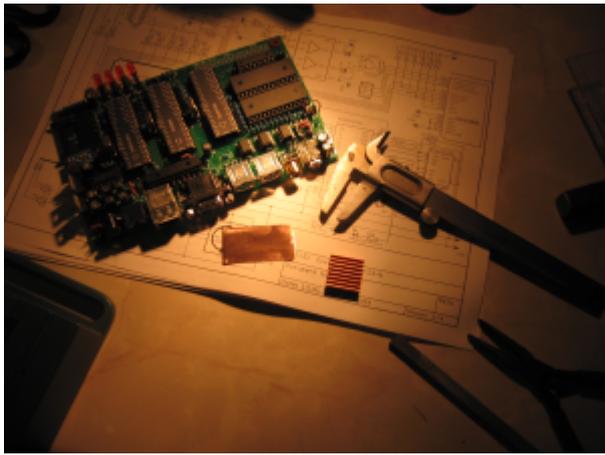


Wenn dieser Test erfolgreich war, kann der Hive in einer minimalen Version in Betrieb genommen werden. Das Paket "Startdateien" aus dem Downloadbereich enthält eine Minimalausstattung von Dateien, um den Hive vollständig zu booten. Es gibt dabei je ein BIOS für die drei Propellerchips, welche geflasht werden müssen. Die restlichen Dateien werden einfach auf eine SD-Card kopiert. Nach einem Reset sollte der Hive nun von der eingelegten Card booten, es sollten im Lautsprecher entsprechende Ladegeräusche (ein Herzschlagsound) hörbar sein und nach wenigen Augenblicken meldet sich der Hive mit seinem Systemprompt im Textmodus.

## Schritt 9 – Netzwerkinterface

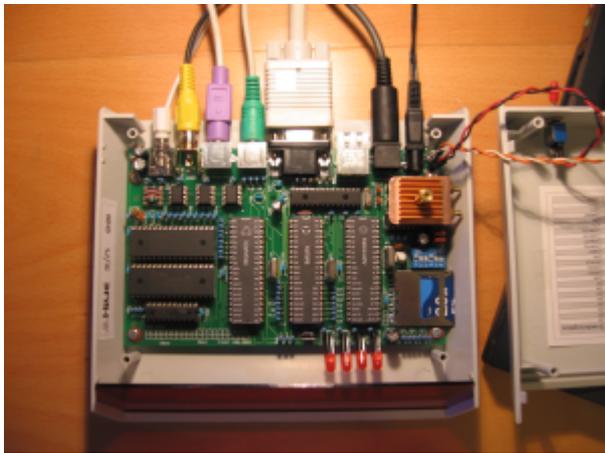
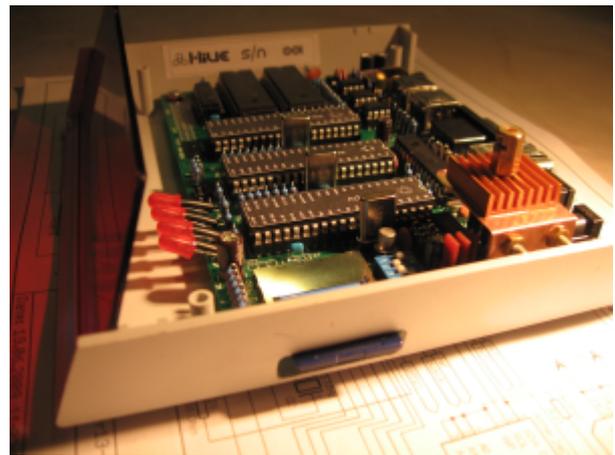


Das Testprogramm für den NIC läuft auf dem Administra-Chip. Die LAN-Schnittstelle ist dafür mit einem normalen LAN Kabel mit einem Switch, oder mit einem Crossover-Kabel direkt mit der Netzwerkbuchse eines anderen Computers zu verbinden. Bei einem erfolgreichen Connect leuchtet die in der Buchse des Hive integrierte LED. Jetzt kann der Testcode in Administra übertragen werden. Dieses kleine Programm enthält einen kompletten TCP/IP-Stack und einen minimalistischen Webserver. Die IP ist im Quelltext veränderbar und steht normalerweise auf dem Wert 192.168.2.1/24. Der Webserver lauscht auf dem Port 88. Nun kann man über einen anderen an das LAN angeschlossenen Computer im Browser unter folgender Adresse <http://192.168.2.1:88/> auf die Test-Webseite zugegriffen werden.



Bitte folgendes beachten: Mit bestücktem NIC sollte man den beiden Spannungsreglern einen Kühlkörper spendieren. Die Kühlfans beider Regler liegen auf dem gleichen Potential (GND) und können deshalb ohne Isolierung auf dem gleichen Kühlkörper verschraubt werden. Ich selbst habe dafür ein Stück verfügbares Kupferblech verwendet, welches ich abgewinkelt habe. Auf der Oberseite habe ich einen kleinen Kühlkörper von einer defekten Grafikkarte aufgeschraubt. Alle Kontaktflächen von dem Kupferblech zu den Reglern bzw. zu dem Kühlkörper habe ich mit ein wenig mit Wärmeleitpaste versehen.

## Abschluß



Als letztes kannst du eine der Arbeitsdisketten downloaden, sie enthalten eine komplette Zusammenstellung an verfügbaren Tools, Demos und den StarTracker. Die größere Version beinhaltet zusätzlich noch einige coole Mediendateien um ein wenig zu experimentieren. Die Arbeiten am Gehäuse überlasse ich jedem Bastler selbst. Kurze Kabel für den Reset-Taster und den Einschalter mit einem zweipoligen Stecker findet man in jedem alten PC-Gehäuse – evtl. hilft da ja der nette Computerhändler um die Ecke aus. Verwendet man das abgebildete Teko-Gehäuse welches man auch bei Reichelt beziehen kann, muß der hintere Durchbruch nach unten ein wenig erweitert werden und es fehlt ein seitlicher Schlitz für die SD-Card. Mit einem Drehmel ist das in kürzester Zeit erledigt, aber man muss sehr vorsichtig sein bei dem weichen Kunststoff.

# Zusätzliche Schaltpläne

