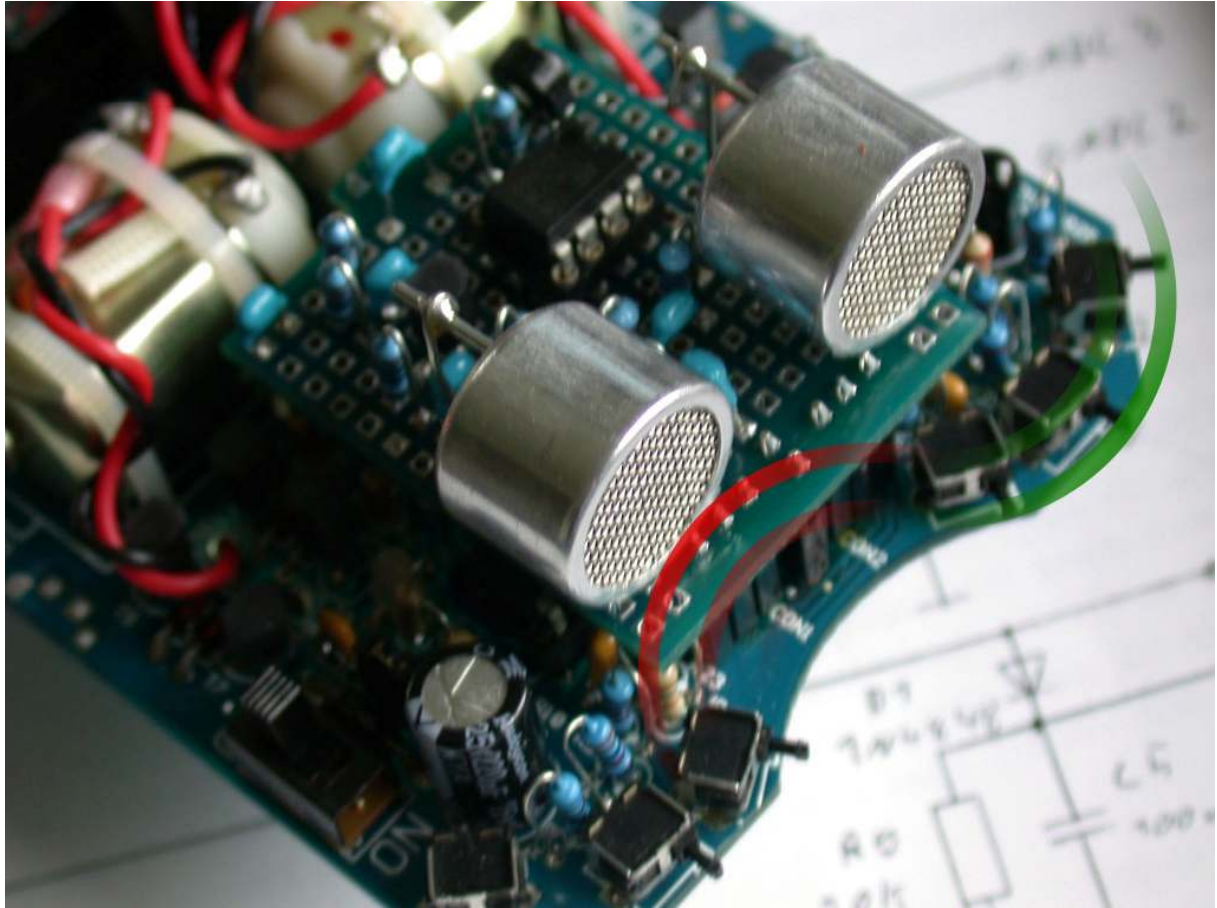


# Ultraschall-Bausatz für den Asuro

*Bauanleitung von Lutz Mager mit Hilfe von  
Robin Gruber*

.....



## Inhalt

1. Vorwort
2. Bauanleitung
3. Test
4. Schlussworte

## 1. Vorwort

Hallo alle zusammen. Anscheinend habt ihr euch also auch den Ultraschall-Bausatz für den Asuro gekauft und kommt mit den (Bau-)Hilfen im Buch „Mehr Spass mit Asuro“ wohl nicht weiter oder habt dieses gar nicht gekauft. Genau deswegen habe ich diese Anleitung geschrieben. Ich hatte genauso harte Startschwierigkeiten. Am Schluss gelang es mir dann doch die Platine fertig zu stellen und erfolgreich zu testen!

An dieser Stelle noch mal vielen Dank an Robin, der wohl allen sehr damit geholfen hat, indem er einen guten Schaltplan im WWW veröffentlichte ([www.roboternetz.de](http://www.roboternetz.de)), auf welchem genau alles eingezeichnet ist wo was auf die Platine hinkommt. Danke!!

Eigentlich sollte dieser Schaltplan als Bauhilfe völlig ausreichen, wenn man dann anfängt zu bauen gibt's allerdings schon eventuell die nächsten Hindernisse. So muss man aus dem Buch ablesen, was der Widerstand R5 ist und wie viel  $\Omega$  der hat. Wenn man allerdings noch nicht gelernt hat wie man die Farbcodes dieser Widerstände zu verstehen hat (und damit nicht auf Anhieb weiß, ob das jetzt ein 10 k $\Omega$  oder 470  $\Omega$  Widerstand, welchen man in der Hand hat, ist), kann man sich da richtig drüber ärgern. Deswegen, diese Bauanleitung. Sie sollte Vollidioten sicher sein ;).

Dann vieeel Spass beim Löten. Grüße Iuma.

Auch in dieser Bauanleitung werden Symbole verwendet:



Bedeutet Achtung. Das solltet ihr lieber 2mal durchlesen.



Bedeutet Hinweis.

## 2. Bauanleitung

### 2.1 Stückliste

Es müssten sich folgende Bauteile in der Packung befinden:

1 x Stiftleiste 10pol. oder mehr (zum Zerteilen)  
1 x Buchsenleiste 10pol. oder mehr (zum Zerteilen)  
6 x Kondensatoren      100 nF keramisch  
4 x Widerstände      10 k $\Omega$       ¼ W 5 %  
1 x Widerstand      100  $\Omega$       ¼ W 5 %  
1 x Widerstand      1k  $\Omega$       ¼ W 5 %  
1 x Widerstand      100 k $\Omega$       ¼ W 5 %  
1 x Widerstand      20 k $\Omega$       ¼ W 5 %  
1 x Widerstand      470 k $\Omega$       ¼ W 5 %  
1 x Trimmer/Poti      1 M $\Omega$   
1 x Ultraschall-Senderkapsel von Polaroid (bei mir 400ST)  
1 x Ultraschall-Empfängerkapsel Polaroid (bei mir 400SR)  
1 x Transistor BC547 (dabei gilt: Entweder B oder C)  
1 x Diode 1N4148  
1 x IC TS912IN oder Vergl.

### 2.2 Wir löten



**Bauteile wie der IC sowie der Transistor sind elektrostatisch gefährdet und hitzeempfindlich und können dadurch sehr leicht zerstört werden!**

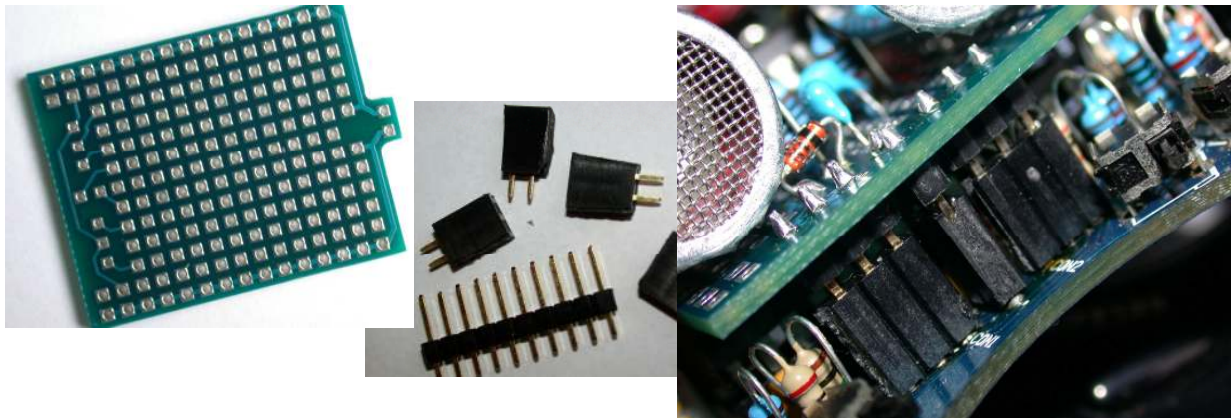
Deswegen **immer** erst an z.Bsp. Heizung **erden** bevor man so ein Bauteil anfasst und den LötKolben an den Transistor nicht Ewigkeiten hinhalten!!

#### Weitere wichtige Hinweise:

- Man lötet erst den Sockel des ICs (IC ist sozusagen der Mikrocontroller) an, später (wenn dann **alles** fertig gelötet ist) wird der IC in den Sockel gesteckt – und zwar vorsichtig!
- Die Widerstände werden (wie beim Asuro) stehend festgelötet. Wer nicht weiß was ich jetzt mein, sollte die Anleitung zur Konstruktion von Asuro noch mal lesen.
- Als erstes werden alle Bauteile festgelötet, anschließend die Kontaktbrücken auf der Unterseite.

### 2.2.1 Die Platine vorbereiten

In diesem Abschnitt wird erklärt wie man die Buchsen- / Steckleisten für die Erweiterungsplatine (richtig) fest lötet.

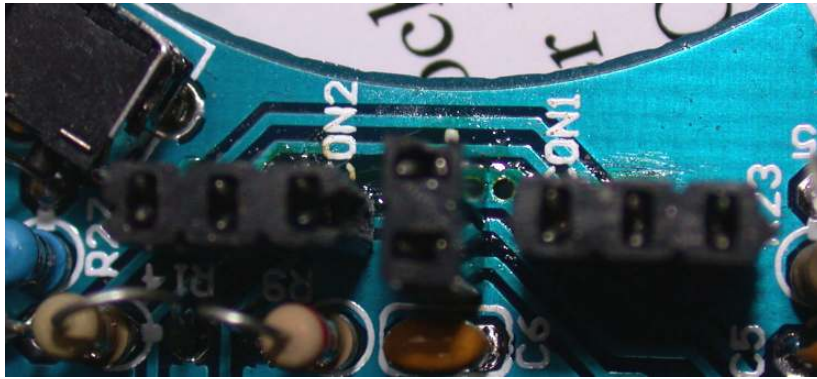


**Lest dieses Kapitel jetzt bitte erst ganz durch bevor ihr euch ans Löten macht!**

Oben links sieht man die noch leere Platine. Diese wird nun mit den Stiften und Buchsen (Mitte) mit der Platine von Asuro verlötet (Rechts). Hier muss man bitte **genau** nach meinen Angaben vorgehen, sonst sieht's nämlich so wie auf meinem Bild rechts aus: Die Stifte stecken nicht richtig in den Buchsen drin und alles ist dann irgendwie schief. Auf die Funktionalität wirkt sich das nicht aus, allerdings pflege ich gerne das Sprichwort „Das Auge isst mit.“ und deswegen versuch ich immer alles so schön wie möglich zu löten, was mir in diesem Fall aber nicht gelungen ist ☹.

Also, man teilt nun die Buchsenleiste in 2 x 2 Pins und 2 x 3 Pins (oder Löcher) auf. Das geht am besten mit nem dicken Küchenmesser. Aber Achtung: Wenn man die Leiste teilt, sollte man auf seine Finger sowie auf evt. auf davon fliegende Teilchen achten (die man dann nicht mehr findet). Das gleiche macht man mit den Buchsenleisten (hier ist der Begriff Pin wohl besser geeignet).

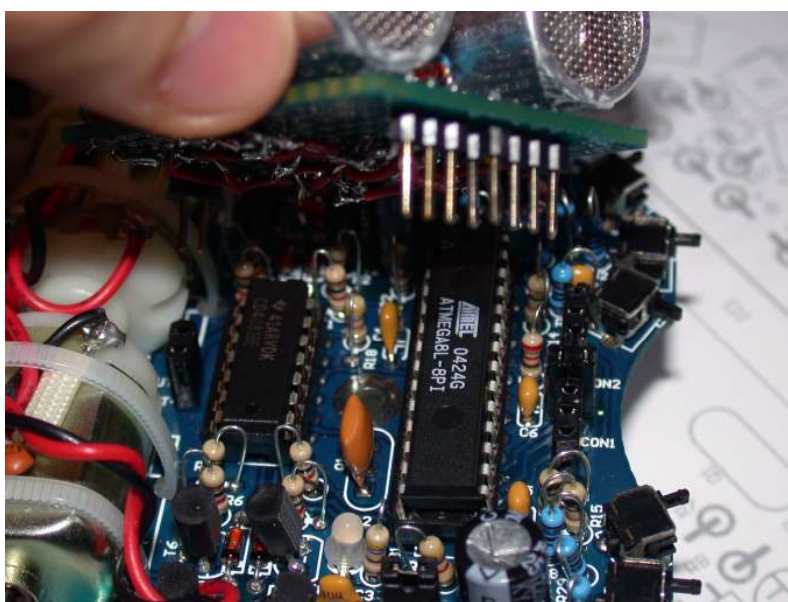
Wenn man das hat, steckt man die einzelnen Stiftleistenstückchen in die entsprechenden Buchsenleistenstückchen (also Stiftleistenstückchen mit 2 Pins in Buchsenleistenstückchen mit 2 Löchern). Anschließend steckt man die Buchsenleisten (**mit den kurzen Enden der Stiftleiste nach oben wegzeigend und mit den langen Enden der Stiftleiste in den Buchsen steckend**) auf die Asuroplatine. Eine 2pol. Leiste kommt in die Lötäugen OUT+ und OUT- die andere 2pol. in die Lötäugen für D11 (rote LED vorne) sowie die zwei 3pol. Buchsenleistenstückchen in die Lötäugen für die Fototransistoren T9 und T10 sowie den beiden Extralötäugen (CON1 und CON2).



Wenn man davor bereits die Bauteile zur Linienverfolgung eingelötet hat, müssen die wieder raus. Mein Tipp: Rackert euch nicht damit ab, die Löcher wieder sauber zu bekommen. Ein Normalsterblicher schafft das eh nicht. Wenn man später einfach die Lötäugen erhitzt und dann im richtigen Moment das Bauteil einsteckt geht das genauso. Das ist zwar ein rechtes Gefummel, weil alles nur Schrittweise geht (da ein Millimeter tiefer, dann dort...) aber es funktioniert bestens. Man sollte dabei auf die Drähte achten. Die können einem leicht ein Brandmal in die Haut rein brennen, wenn man nicht aufpasst. Kurz: Die Dinger werden richtig heiß! Und nicht vergessen die Stifteleitenstückchen müssen schon in den Buchsenleitenstückchen stecken.

Wenn man das hat, steckt man die Erweiterungsplatine auf die kurzen Ende der von der Asuroplatine abstehenden Stifteleiten und verlötet die Platine mit den Enden der Stifte mit der Platine. Dann werden die Buchsenleiten auf der Asuroplatine fest verlötet.

Wenn alles erkaltet ist, sollte sich die Erweiterungsplatine problemlos ab- und anstecken lassen. Wenn nicht hilft ein vorsichtiges biegen.

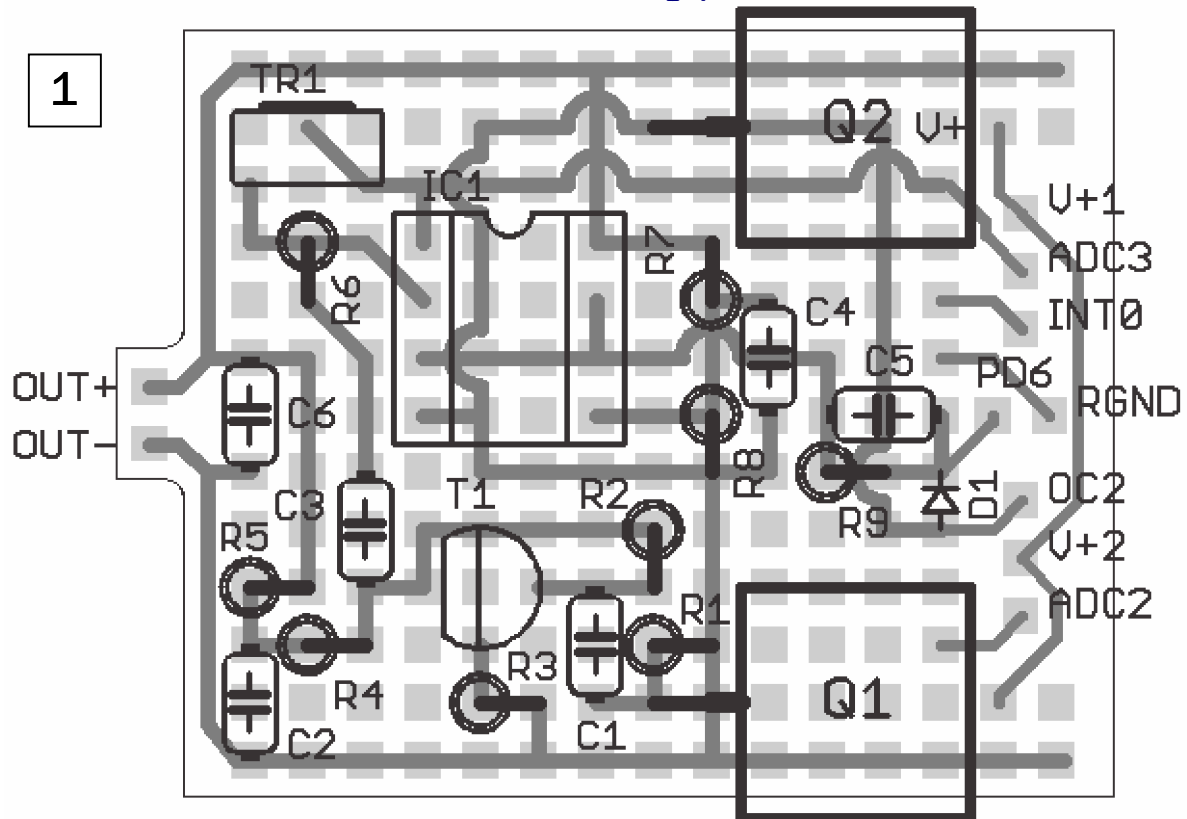


Hier ziehe ich gerade die Erweiterungsplatine (ist schon fertig verlötet) von der Asuroplatine ab.

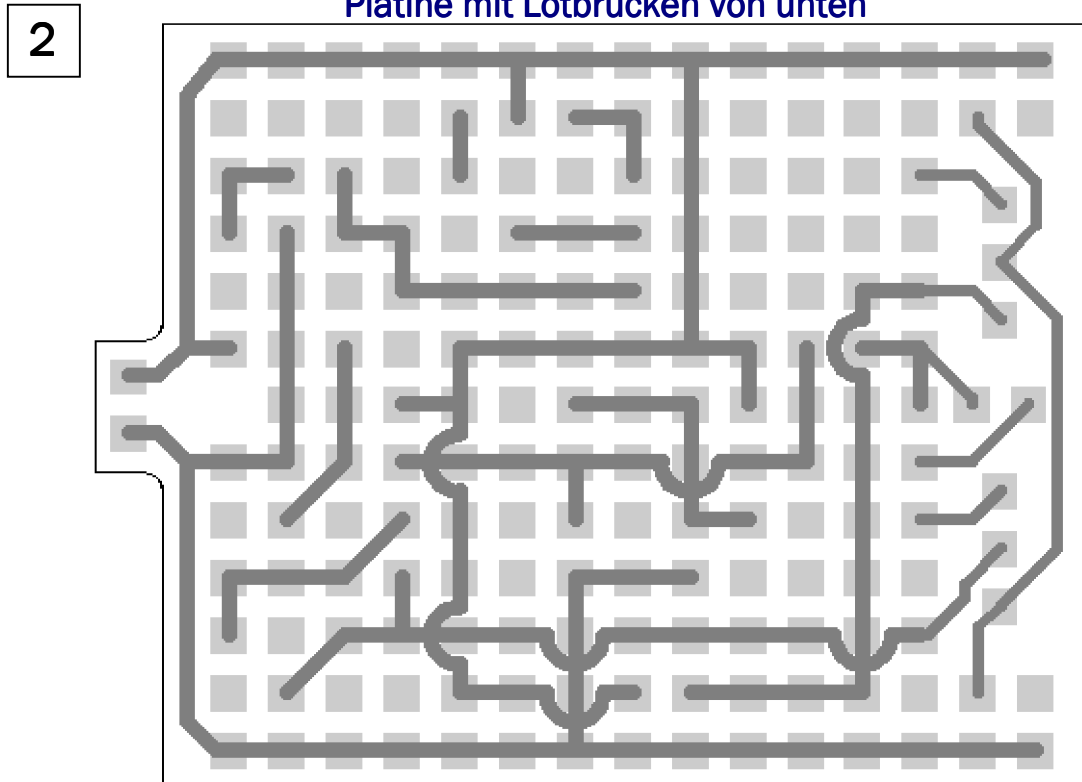
### 2.2.2 Bauteile verlöten

Nun ist alles elementare fertig. Wir legen den Asuro beiseite und die US-Platine wird nun verlötet.

Platine mit Bestückungsplan von oben



Platine mit Lötbrücken von unten



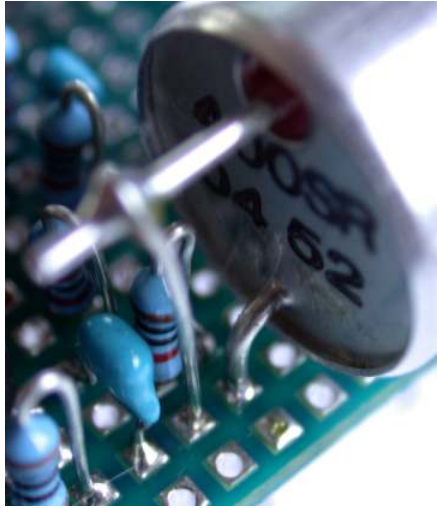
Kurz eine Erklärung zu den Skizzen. Die Skizze 1 zeigt die Platine von oben und wie die Bauteile anzulöten sind. Kleiner Hinweis: Die Kondensatoren sind (bei mir jedenfalls) schön so schön vorgeknickt, damit sie in genau 2 parallele Lötungen passen. In der Schaltung ist allerdings immer ein Lötauge dazwischen. D. h. ihr müsst die Beinchen der Kondensatoren auseinander biegen, aber vorsichtig. **Wichtig:** Isoliert die US-Kapseln, wenn sie evt. Einen Kontakt zu den Pins vorne haben könnten!

#### Verwendete Kurzbezeichnungen:

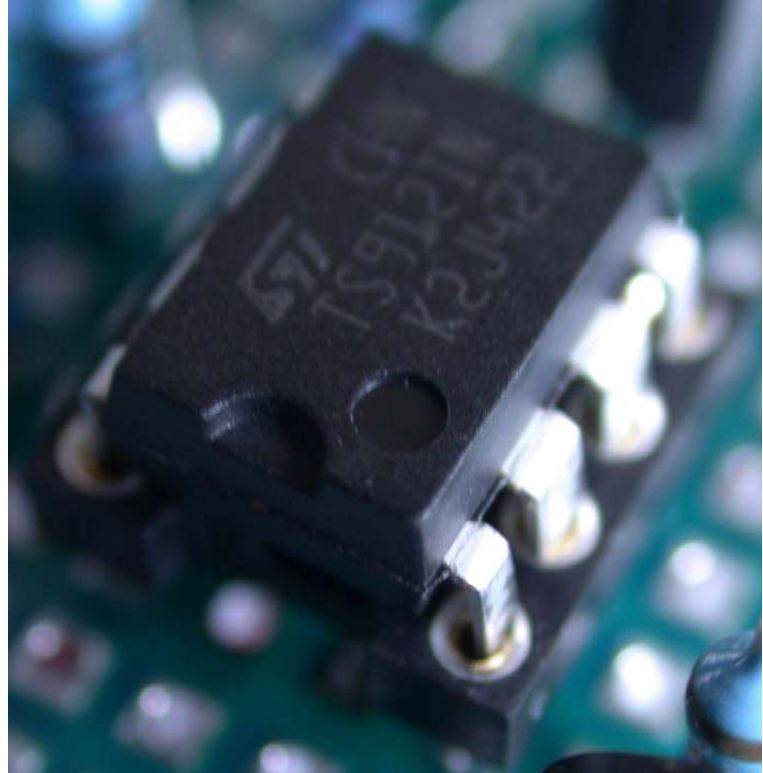
R	Widerstand
T	Transistor
D	Diode
C	Kondensator
TR	Trimmer / Poti
Q	US-Kapsel

#### Bestück wird so:

- R1: 10 k $\Omega$  (braun, schwarz, schwarz, rot, braun)
- R2: 470 k $\Omega$  (gelb, violett, orange, braun)
- R3: 100  $\Omega$  (braun, schwarz, schwarz, schwarz, braun)
- R4: 10 k $\Omega$  (braun, schwarz, schwarz, rot, braun)
- R5: 1 k $\Omega$  (braun, schwarz, schwarz, braun, braun)
- R6: 100 k $\Omega$  (braun, schwarz, schwarz, orange, braun)
- R7: 10 k $\Omega$  (braun, schwarz, schwarz, rot, braun)
- R8: 10 k $\Omega$  (braun, schwarz, schwarz, rot, braun)
- R9: 20 k $\Omega$  (rot, schwarz, schwarz, braun)
- C1 – C6: 100 nF (Polung ist egal)
- TR1: 1 M $\Omega$  (Poti/Trimmer)
- D1: 1N4148 (Auf richtige Polung achten. Strich der Diode auf Strich des eingezeichneten Dreieckes)
- Sockel für IC1
- Q1: Bei mir steht die Bezeichnung 400SR hinten drauf. Das Beinchen mit roter Umrandung ist oben, das andere unten. Das untere wird abgeknickt, das obere mit einem Draht verlängert (siehe Foto unten).
- Q2: Bezeichnung bei mir 400ST. *Siehe Beschreibung für Q1.*
- IC1: Beinchen vorsichtig in die Fassung einstecken **und auf richtige Polung achten: Kerbe nach oben wie auf der Zeichnung!** Davor **unbedingt erden!** Wenn die Beinchen nicht einfach reinpassen, vorsichtig an der Tischkante etwas einbiegen und dann einstecken. Beim Einstecken ist manchmal ein bisschen „zarte“ Gewalt von Nöten, damit der IC richtig drinsteckt.

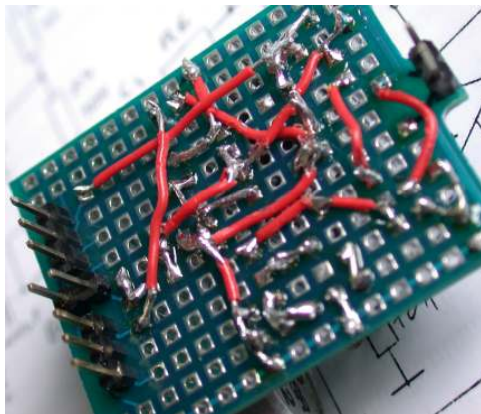


Oben: Eine Ultraschall-Kapsel mit Verlängerung.



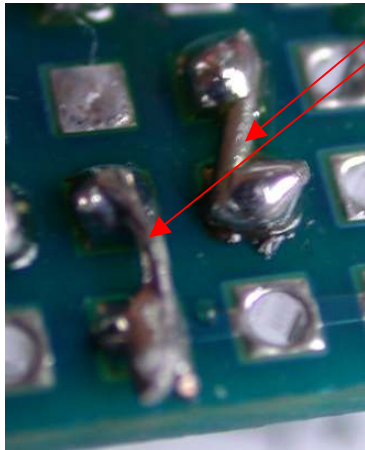
Rechts: Richtig eingesteckter IC (auf Kerbe achten).

Damit hätten wir alle elektr. Bauteile verlötet. Wer jetzt glaubt er sei fast fertig, hat sich allerdings geirrt :p. Jetzt kommt das nervigste. Da die Platine keine vorgefertigten Leiterbahnen integriert hat, müssen wir die nachbauen und zwar in Form von Drahtbrücken. Das sieht dann etwa so aus:



Ziemlich unübersichtlich, was ☺? Und ja, da schleichen sich seeehr gerne Fehler ein.

Jetzt hat es sich gelohnt bis hier her gelesen zu habe, **ohne** bereits den LötKolben in die Hand genommen zu haben. Wie man auf der Schaltplanskizze 2 (siehe Seite 6) sieht, liegen manchmal mehrere Bauteile direkt auf der Platine nebeneinander und müssen mit einer Brücke verbunden werden. Hier ist es auf jeden Fall einfacher, ein Bein des Bauteils einfach etwas länger zu lassen, um es dann umzuknicken und es als Brücke zu verwenden. Ich hab das leider nicht gleich von vorneherein gemacht und mir dadurch viel Ärger eingehandelt. Es ist nämlich nicht gerade einfach ein 1 mm langes Drahtstück zw. 2 Pins zu verlöten...



Hier hätte ich mir viel Arbeit sparen können...

Mehr kann ich euch nicht mehr sagen. Nur: Lötet sauber, baut keine Kurzschlüsse (die gelbe Schlont\*\* die mein Löten gern entsteht (ist Löt fett) leitet nicht, da müsst ihr euch keine Gedanken über Kurzschlüsse machen), verlötet nichts falsches und vergesst keine Bauteile zu verlöten! Wenn ihr fertig seid, **alles noch mal genau überprüfen!**

Wenn dann wirklich alles passt wird die Erweiterungsplatine auf die Asuroplatine gesteckt.

**Achtung: Ihr müsst genau schauen, das es keine Kurzschlüsse zw. Erweiterungsplatine und Asuroplatine mit irgendwelchen kollidieren Bauteilen gibt!! Sonst könnt ihr euren Asuro + US-Erweiterung schrotten!!**

## 2.3 Test

Natürlich müssen wir die Platine jetzt testen! Dazu gibt's folgenden Code. Danke an Robin und Jan, die den Code so geschrieben haben (Nachzulesen im Buch „Mehr Spass mit Asuro“).

```
#include "asuro.h"

void LocalInit(void)
{
    // Change Oscillator-frequency of Timer 2
    // to 40kHz, no toggling of IO-pin:
    TCCR2 = (1 << WGM21) | (1 << CS20);
    OCR2 = 0x64; // 40kHz @8MHz crystal
    ADCSRA = 0x00; // ADC off
    // Analog comparator:
    ACSR = 0x02; // Generate interrupt on falling edge
    ADMUX = 0x03; // Multiplexer for comparator to
    // ADC pin 3

    SFIOR |= (1 << ACME); // Enable multiplexing of comparator
    DDRD &= ~(1 << 6); // Port D Pin 6 is input!
}

void Ping(unsigned char length)
{
    count72kHz = 0;
    TCCR2 = (1 << WGM21) | (1 << COM20) | (1 << CS20);

    // Toggling of IO-Pin on

    // generate the Chirp
    while(count72kHz < length) {
        OCR2 = 0x64 + length / 2 - count72kHz;
    }

    TCCR2 = (1 << WGM21) | (1 << CS20); // Toggling of IO-Pin off
    OCR2 = 0x64; // set frequency to 40kHz
}

int main(void)
{
    int pos, i;
    int posmarker;
    Init();
}
```

```

LocalInit();
while(TRUE) {
    posmarker = 0;
    Ping(20);
    for(pos = 0; pos < 100; pos++) {
        Sleep(10);
        if((ACSR & (1 << ACI)) != 0) {
            if(posmarker == 0) { posmarker = pos; }
        }
        ACSR |= (1 << ACI);
    }
    if(posmarker > 10) {
        StatusLED(GREEN);
        MotorDir(FWD, FWD);
        MotorSpeed(200, 200);
    }
    else {
        StatusLED(RED);
        MotorDir(FWD, RWD);
        MotorSpeed(0, 200);
        for(i = 0; i<100; i++) { Sleep(200); }
    }
}
return 0;
}

```

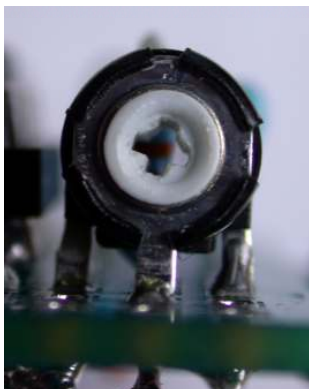
**i**  
**i**

**Hinweis:** Dieser Code findet sich im Archiv mit dem Name test.c!

**Hinweis 2:** Wenn ihr das so kompiliert (asuro.c, asuro.h und makefile müssen im gleichen Verzeichnis liegen) gibt's nen Fehler. Verschiebt deswegen die Variable **volatile unsigned char** count72kHz von asuro.c nach asuro.h! Sonst tut sich gar nichts. Dann wird das ganze mit dem neusten Asuro FLASH (v1.4 ist gerade aktuell) auf den Asuro geflasht und dann ab.

Wenn sich der Asuro beim Einschalten irgendwie komisch verhält, sofort ausschalten und auf Lötfehler genausten überprüfen!! Wenn nicht sollte der Bot im Zimmer rumfahren und nirgends dagegen bockeln. Auf einem Abstand von etwa 20 cm sollte er immer nach

Rechts ausweichen. Wenn ich, ein bisschen am Poti drehen. Ich habe meinen so eingestellt:



← Der Pfeil steht im nach oben, leicht nach rechts. Das entspricht ungefähr einem Widerstand von 430 kΩ.

**Wichtig:** Passt unbedingt auf, dass der Asuro nicht auf irgendwelche Metallteile auffahren kann die das US nicht erkennt. Ist mir nämlich so

ergangen. Zum Glück scheint nichts kaputtgegangen zu sein. Die Status-LED wechselte nur auf Gelb, das war's (zum Glück!). Also, passt gut auf!

**i** Wem es jetzt schon wieder langweilig ist, kann einen sehr interessanten Versuch machen. Normale Menschen hören US nicht, Katzen oder Hunde dagegen schon! D. h. jeder der ein Haustier in Form einer Katze oder Hund hat, setzt sich mal ganz diskret neben das arme Tier und starte den Asuro (Programm so ändern, das sich kein Motor dreht, sondern nur die LED leuchtet). Normalerweise müsste das Versuchsobjekt jetzt irgendeine Reaktion zeigen (außer es pennt so tief, dass es eh nichts mitbekommt). Tiere hören US als sehr hohen Ton (ähnlich wie der Fernseher pfeift). D.h. die Tiere mögen diesen Ton wahrscheinlich nicht sonderlich. Quält sie daher nicht mit ständigem gechirpe, sondern testet euren Bot nicht in Nähe eines solchen Tiers!

## 4. Schlussworte

Ich hoffe euch hat meine Anleitung gefallen. Und verzeiht mir die unzähligen Komma- und Rechtschreibfehler.  
Und jetzt viel Spass beim CHIRPEN!

## X. Anhang

Für alle die verstehen wollen wie man die Widerstandcodes liest, ist hier eine kleine Tabelle. Für weitere Informationen hilft Wikipedia im WWW bestimmt immer aus ☺!

Farbe	Abkürzung	Wert	Multiplikator	Toleranz	Temp.Koeffizient
schwarz	sw	0	1	-	-
braun	br	1	10	1 %	100 ppm
rot	ro	2	100	2 %	50 ppm
orange	or	3	1.000	-	15 ppm
gelb	ge	4	10.000	-	25 ppm
grün	gn	5	100.00	0,5 %	-
blau	bl	6	1.000.000	0,25 %	-
violett	vio	7	10.000.000	0,1 %	-
grau	gr	8	100.000.000	-	-
weiß	we	9	1.000.000.000	-	-
silber	-	-	0,01	10 %	-
gold	-	-	0,1	5 %	-