Der Logik auf der Spur...

Um dem Logikprint seine Geheimnisse zu entlocken, gibt es generell zwei Wege:

1. Den Speicher der Maschine direkt auf der Platine auslesen
2. Die Serviceschnittstelle zu nutzen.

Der erste Weg

Wie funktioniert's generell?

Die Logikeinheiten der Jura, älteren AEG CaFamosas und die Krups Orchestro basieren auf einem gleichen Grundprinzip: In allen Maschinen tut ein Mikroprozessor 64732(µP) von Renesas (ehemals Hitachi) seinen Dienst. Er regelt die Programmabläufe für Bezug, Reinigung, Entkalkungsvorgänge usw. Dieser µP ist ein so genannter „maskenprogrammierbarer" Prozessor, der einmal „eingeschossen", also programmiert wird. Einmal programmiert, lässt sich das Programm auch nie wieder ändern. Um dennoch den Ablauf der Maschinen ändern zu können, werden Parameter vorgegeben und einem Speicher abgelegt.
Ein Beispiel mag das illustrieren: Die Temperatur des Heizelementes wird auf einen Sollwert von 95°C eingestellt. Dieser Wert wird (verschlüsselt) im Speicher an der Adresse ‚aaab' abgelegt. Das Programm sagt dann nicht: "Heizstrom einschalten bis 95°C erreicht werden", sondern: „Heizstrom einschalten, bis der Wert in Speicherzelle ‚aaab' erreicht wird".
Unterschied verstanden? Der Prozessor bestimmt den generellen Ablauf des Programms, die Speicherwerte bestimmen, mit welchen Parametern der Ablauf gesteuert wird.

Der Speicher

Da das generelle Programm im Prozessor nicht verändert werden kann, schauen wir uns also den Speicher der Geräte an. Dieser Speicher ist ein so genanntes „EEPROM“ („electrically erasable and programmable read only memory“, also ein durch elektrische Signale lösch- und programmierbarer Lesespeicher). Man kann also in so ein EEPROM Werte einschreiben, löschen und wieder auslesen. So ein EEPROM verliert auch bei Stromausfall seine Werte nicht, ist recht robust und deshalb besser geeignet als RAM in einem Computer.

Jura speichert im EEPROM nicht nur Werte zur Programmablaufsteuerung, sondern auch viele andere Werte. Gesamtbezüge, Bezüge seit dem letzten Reinigungszyklus, Bezüge einzelner Tassen, Doppelbezüge und viele, viele mehr.

Das EEPROM

Das in den Jura- Maschinen verwendete EEPROM ist ein „93C56“, ein achtbeiniger Chip von gängiger Bauart. Nichts Besonderes und in jedem Elektronikladen erhältlich. Um mehr darüber zu erfahren, besorgt man sich von so einem Chip zuerst mal das Datenblatt. Das steht fast alles drin was man braucht, um etwas mehr über den Chip zu erfahren. Man googelt „93C56“ und findet schnell ein Datenblatt. Von welchem Hersteller ist egal – alle 93C56 funktionieren gleich. Das Datenblatt sagt uns dann, dass das 93C56 insgesamt 256 Byte Speichervermögen hat (=256 x 8 bit).

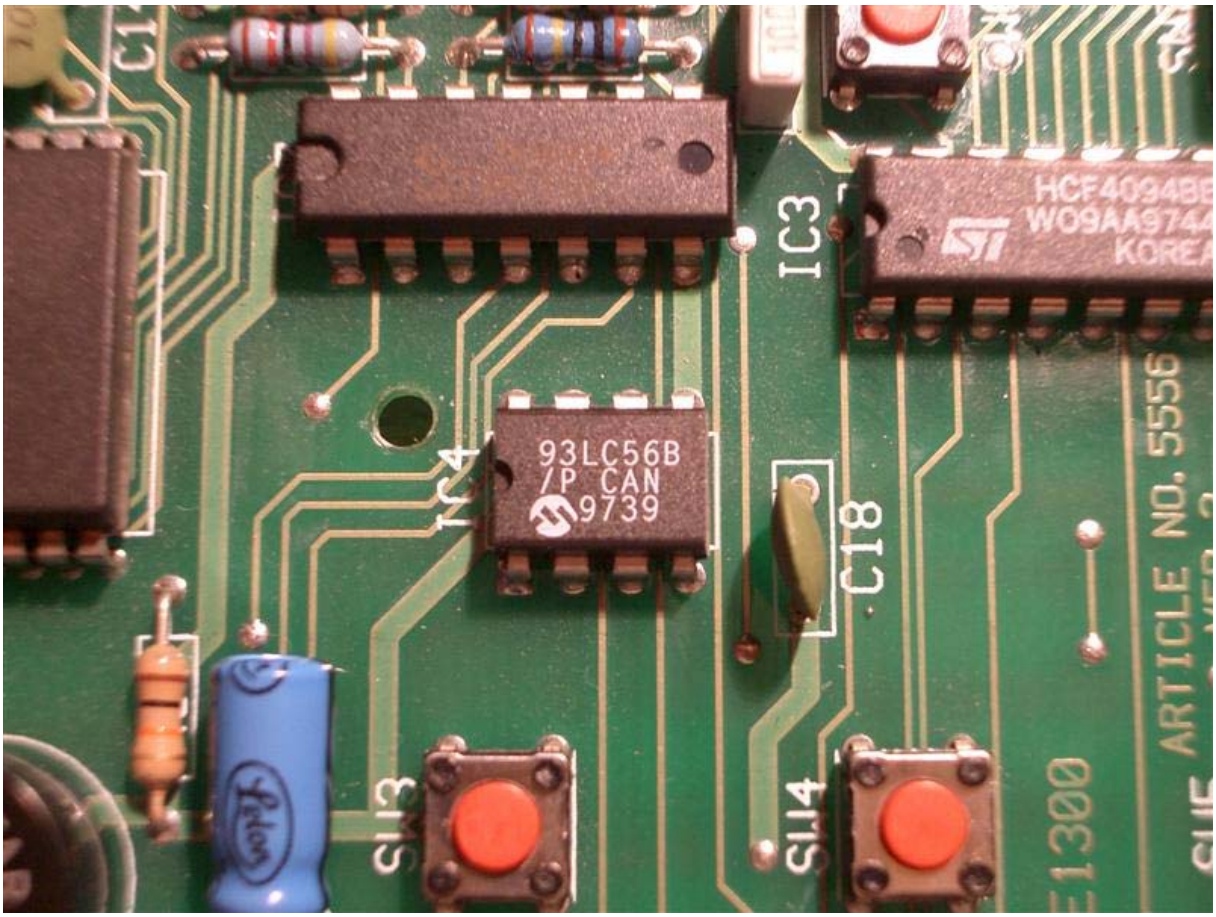
In den 256 Byte Speicherplatz können entweder 256 größere Zahlen (Mit einem Byte kann man von 0 (0000 0000) bis 255 (1111 1111) darstellen. Nimmt man 2 Byte , kann man Zahlen von 0 bis 65535 (genug, um Bezüge zu zählen) darstellen.

Das Auslesen und Programmieren des EEPROMS

Um EEPROMS lesen, löschen und beschreiben zu können, benötigt man einen „EEPROM- Programmer“. Die kosten Geld oder Einfallsreichtum. Wer letzteres bevorzugt und nicht fertig kaufen möchte (und löten kann), dem sei die Seite http://www.dl7awl.de/ee93.htm empfohlen. Der Autor beschreibt hier den Selbstbau eines Programmers, der mit minimalen Kosten (5 EUR) selbst gebastelt werden kann und am Parallelport eines (älteren) PC funktioniert. Die Software dazu gibt es auf der gleichen Seite.
Wer es etwas komfortabler mag, dem sei der Pony Programmer von Lancos (http://www.lancos.com/ bzw. http://www.lancos.com/e2p/ponyprog2000.html) empfohlen.
Peter H. Anderson hat auch noch eine Lösung beschrieben, die sich insbesondere für Leute eignet, die das Programm in C selbst verändern wollen. Auch er benutzt den Parallelport zur 93C56- Programmierung. Beschrieben ist das Ganze unter:
www.phanderson.com/printer/eeprom/eeprom.html .

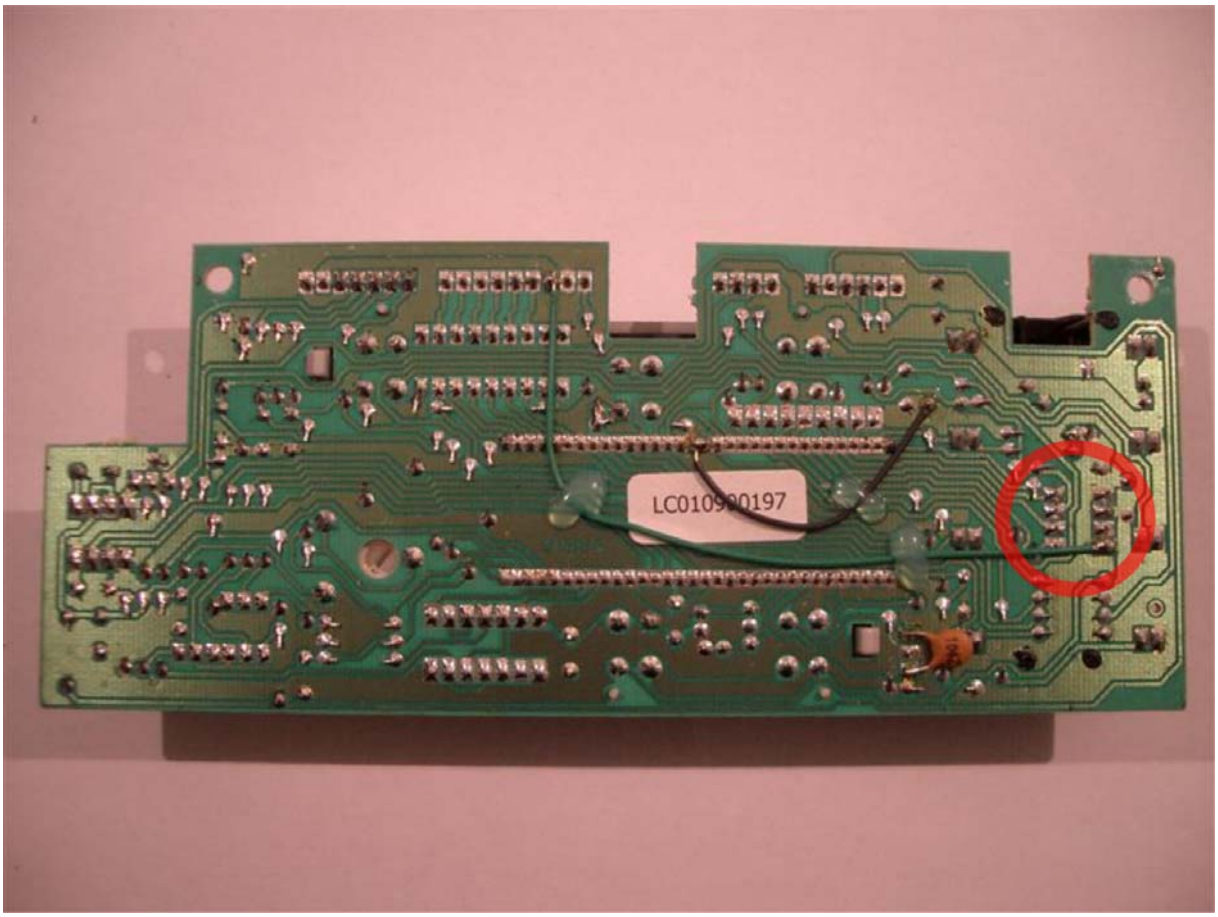
Natürlich gibt es noch –zig andere, aber ich habe mich hier mal auf „erschwingliche“ bzw. Selbstbau- Lösungen beschränkt.
Ich persönlich benutze den DW7AWL- Programmer. Damit begeben wir uns also an die Arbeit.

Das 93C56 sitzt bei älteren Impressa- Modellen frei zugänglich auf dem Logikprint.



{Bild xyz – EEPROM auf dem Logikprint einer Impressa 500}

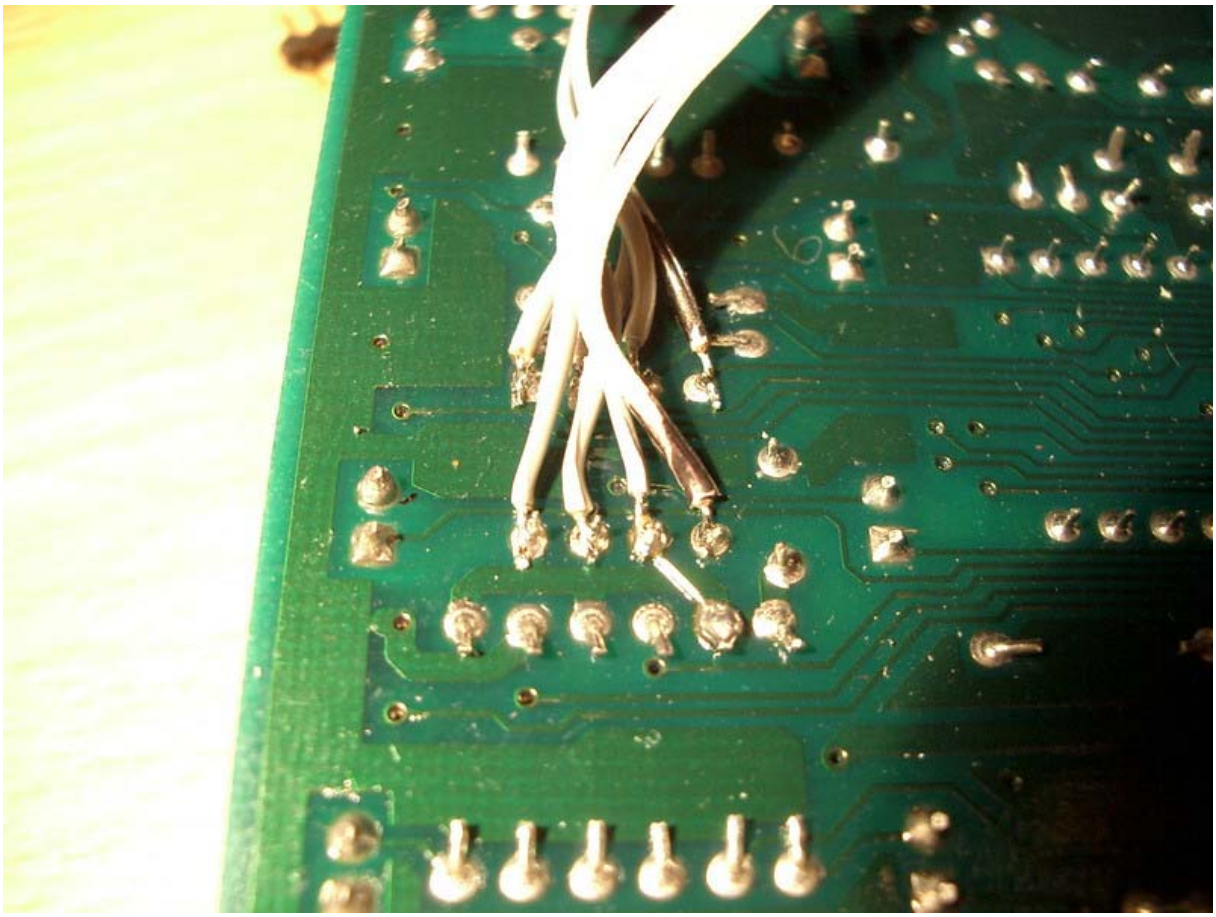
Bei den E-Serien- Modellen ist es etwas komplizierter. Da sitzt das EEPROM direkt unter den 8 LED's bzw. unter dem Display der Frontplatte.



{Bild xyz – Die Anschlusspins des EEPROMS unter dem LED Panel}

Ist man nur daran interessiert, die Daten aus dem EEPROM auszulesen, muss dieser nicht ausgebaut werden. Es bieten sich zwei Varianten, je nach Maschinentyp, an. Bei den älteren Impressa 500 liegt das EEPROM frei zugänglich. Bei diesen Maschinen kann man das EEPROM auch auslesen, indem man eine Chipklammer darauf setzt. Die zweite Variante, welche ohne Auslötarbeiten auskommt, ist das anlöten von 8 Leitungen an die Pins des EEPROMs von der Lötseite der Platine aus.

Hierfür eignet sich recht gut Flachbandkabel aus alten PC Beständen. Auf das andere Ende einfach eine weitere Präzisionsfassung auflöten – diese passt dann in den Sockel des Schreib-/Lesegerätes…



{Bild xyz – Lötaschluss an EEPROM Pins}

Will man jedoch auch Werte in das EEPROM zurückschreiben, muss der Chip ausgelötet werden, da er in eingebautem Zustand nicht beschrieben werden kann. Anschliessend wird ein 8-pin-Präzisionssockel eingelötet, damit wir das Ding später ohne weitere Lötarbeiten rein und raus bekommen. (Hinweis: Wenn bei den älteren Impressa Modellen der Sockel auch nach dem Zusammenbau der Maschine dort bleiben soll, muss der Tastaturrechen nachgearbeitet werden, da der Chip mit Sockel zu hoch ist).

Das ausgelötete EEPROM geht auf das selbst gebaute Interface des DW7AWL- Programmers, unbedingt die korrekte Pinlage beachten.
Software starten und jetzt wird das EEPROM zunächst mal ausgelesen.

Das Ergebnis sieht dann so aus:

```
0000  A722 4500 0000 0000 0000 0000 7000 C80D
0008  2200 1D00 1D00 0000 0000 0400 0000 3C00
0010  A64F 1F00 E000 0000 FFAD 3706 0000 0000
0018  0000 0000 0100 E203 8301 0200 0200 F503
0020  8C78 282D 282D 0000 3C96 5802 0064 3C00
0028  0003 0D03 0A10 1406 040A CC06 0A16 0407
0030  F003 0B00 5712 3012 0000 C003 060E 4C04
0038  A08C FA00 9710 6440 1410 9001 1414 7602
0040  0F00 E600 0A01 BE01 2202 A406 0807 8C0A
0048  C409 8A02 641A 6414 B004 DC05 E803 1C96
0050  2316 0A1B 004B 0290 050A 0210 FD00 B400
0058  0404 3C64 2003 3246 5014 3CC8 0E3C 8114
0060  9320 0099 7519 9300 6400 6C1B 9330 3491
0068  0500 C9A0 9212 5A15 0A08 0232 020C F203
0070  C201 D417 0606 7C01 1A10 5E0D 0005 0807
0078  4006 F401 003C 0005 FE4E 560C 0000 8400
```

Die Spalte mit der ersten Vierergruppe ist die Adresse des Speichers. Alle Zahlen sind in Hexadezimalzahlen angegeben, eben so wie sie aus dem DW7AWL – Programm herauskommen.
Umrechnen kann man die ganze Tabelle entweder „zu Fuß“ oder besser mit Excel. Man importiert die Textdatei (Datei; Öffnen – Texdateinamen eingeben) in eine Excel- Tabelle und schon stehen die Zahlen schön in der ersten Spalte einer Tabelle. Dann laden wir die Datei „Speicherdump Jura“ herunter und kopieren den Output ab Zelle A2 in die „Rohdaten“ - Tabelle.
Trennen wir also mal auf, ohne mehr von der Struktur und dem Inhalt der Speicherzellen zu wissen. Das macht die Tabelle „Excel-Daten(nur zw-Ergebnisse)“ mit ihren darin hinterlegten Funktionen. Weil die Programmierer immer so komisch denken, müssen jetzt noch die Zahlenpaare der Speicherzellen getauscht werden. Die letzten 2 Ziffern

werden die ersten zwei und umgekehrt. Das erledigt dann wieder die Tabelle „Daten2 (nur zw-Ergebnisse)“. In der letzten Tabelle „Ausgabe werden nun die Daten von Hexadezimalzahlen in „richtige“ Dezimalzahlen umgerechnet.
Die HEXINDEZ()- Funktion, die in diesem Tabellenblatt gebraucht wird, steht aber nur denjenigen in Excel zur Verfügung, die im Add-in Manager (Extras ->Add-in-Manager) auch die „Analysefunktionen“ mit installiert haben. Im „Standard- Excel“ ist diese Funktion nicht verfügbar.

So jetzt kann es losgehen. Auslesen- beziehen- auslesen. Und dann vergleichen.
So kommt man an die ersten Ergebnisse zu den Zählerständen. Gespeichert werden alle möglichen Details. Kleine Tassen, grosse Tassen (bei der I500), Wasserbezugsmenge und und und....
Alle bisher identifizierten Speicherzellen sind in der Excel Tabelle bereits dokumentiert und werden dort auch ständig nachgepflegt.
Jeder, der weitere Speicherzellinhalte herausfindet, möge sich bitte melden. Dann trage ich die Ergebnisse hier zusammen.

Tachomanipulationen (Bezugszähler ändern) sind eine triviale Anwendung der Speichermanipulation und von einigen begehrt. Ist einfach, hat aber keinen sittlichen Nährwert. Außer vielleicht, seine Mitmenschen beim Verkauf einer Gebrauchten zu betrügen. Igitt! Würde doch wohl niemand ernsthaft in Erwägung ziehen (glaube ich, um meine gute Meinung über die Menschheit zu erhalten).

Der zweite Weg

Jede Jura, AEG und Krups hat eine Serviceschnittstelle. Über dem Display der E- Serie (=AEG, Krups) ist ein kleiner Plastikdeckel, der abgenommen werden kann. Darunter ist ein 5- poliger Stecker: Die Serviceschnittstelle.

Bei den älteren Impressa 500 ist das übrigens mit Infrarot gelöst. Der IR-Sensor für Empfang (Receive, Abk: Rx) und Senden (Transmit, Abk: Tx) sitzt hinter der Klappe für die Programmiertasten. Wie das mit der IR-Programmierung geht, weiss ich noch nicht. Im Prinzip sollte die Verbindung zwischen jedem PC bzw. Laptop mit IR- Schnittstelle möglich sein. Hieran habe ich aber noch nicht gearbeitet.

An der (Stecker-) Serviceschnittstelle befinden sich 5 Pins (Löcher). Davon sind 4 belegt. Die Belegung ist wie folgt:
Tx – Transmit
Rx
SG Signal Ground / Masse
+5 Volt vom Logikprint.

Wer sich mit serieller Kommunikation zwischen PCs auskennt, dem dämmerts jetzt schon. 3-Leiter- Kommunikation ohne RTS/ CTS-Steuerung. So ziemlich das einfachste, was man mit der RS232 machen kann. Aber vorsicht! Nicht zu früh jubeln: Der Teufel steckt im Detail.

Die Pegel (Spannungen) des Mikroprozessors und der RS-232 sind nicht kompatibel. Einfach die Pins zwischen der Jura und dem PC verbinden geht so nicht. Um eine korrekte Wandlung eines TTL Signals (des Mikroprozessors der Kaffeemaschine) auf ein RS232 Signal (des PC'S) zu ermöglichen, muss eine Pegelumsetzung auf die höheren Pegel erfolgen und zusätzlich muss das Signal invertiert werden. Eine logische 1 des TTL- Levels entspricht 5V und bei RS232 -12V. Die Wandlung von RS232 auf TTL erfolgt analog dazu: Ein Pegel von -12V muss auf +5V umgesetzt werden. Solche Pegelumsetzer sind überall unter der Bezeichnung MAX232 erhältliche Chips. Der MAX232 verfügt über zwei TTL auf RS232 und zwei RS232 auf TTL- Stufen. Eine davon wird meist für RxD und TxD verwendet. Die zweite kann entweder für Steuerleitungen der RS232 oder

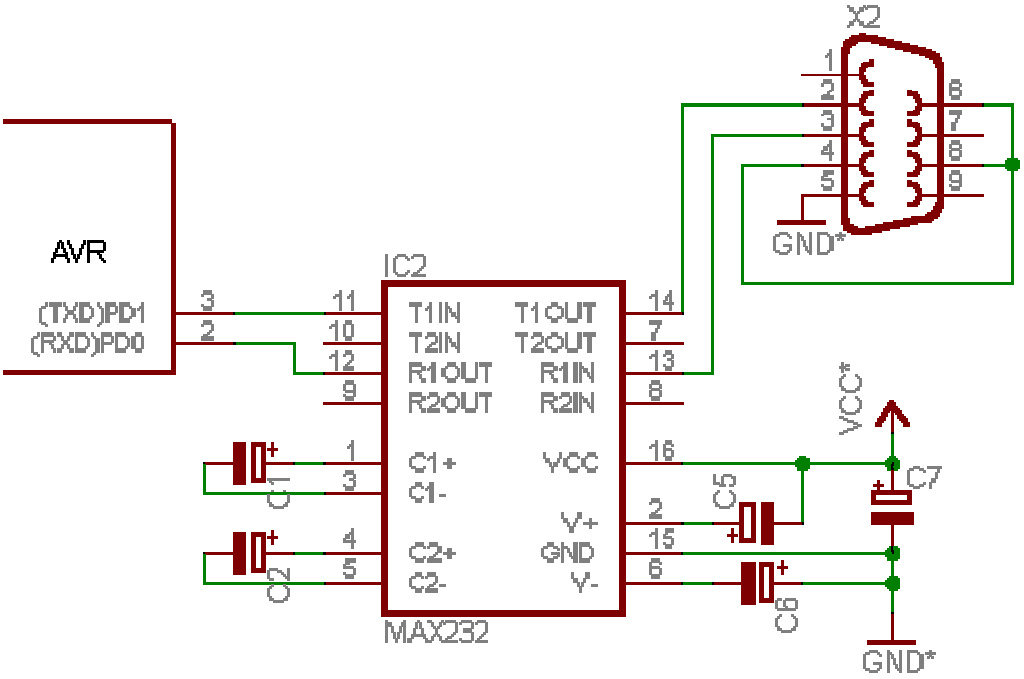
für einen zweiten Kanal verwendet werden. Diese zweite Stufe wird in unserem Fall nicht benötigt.

Die benötigen eine Versorgungsspannung von +5 Volt und als externe Beschaltung einige Kondensatoren. Die Beschaltung ist aber einfach zu bewerkstelligen. Die nötige Versorgungsspannung liefert die Maschine an der Serviceschnittstelle gleich mit.

Mehr über den MAX232 steht unter:

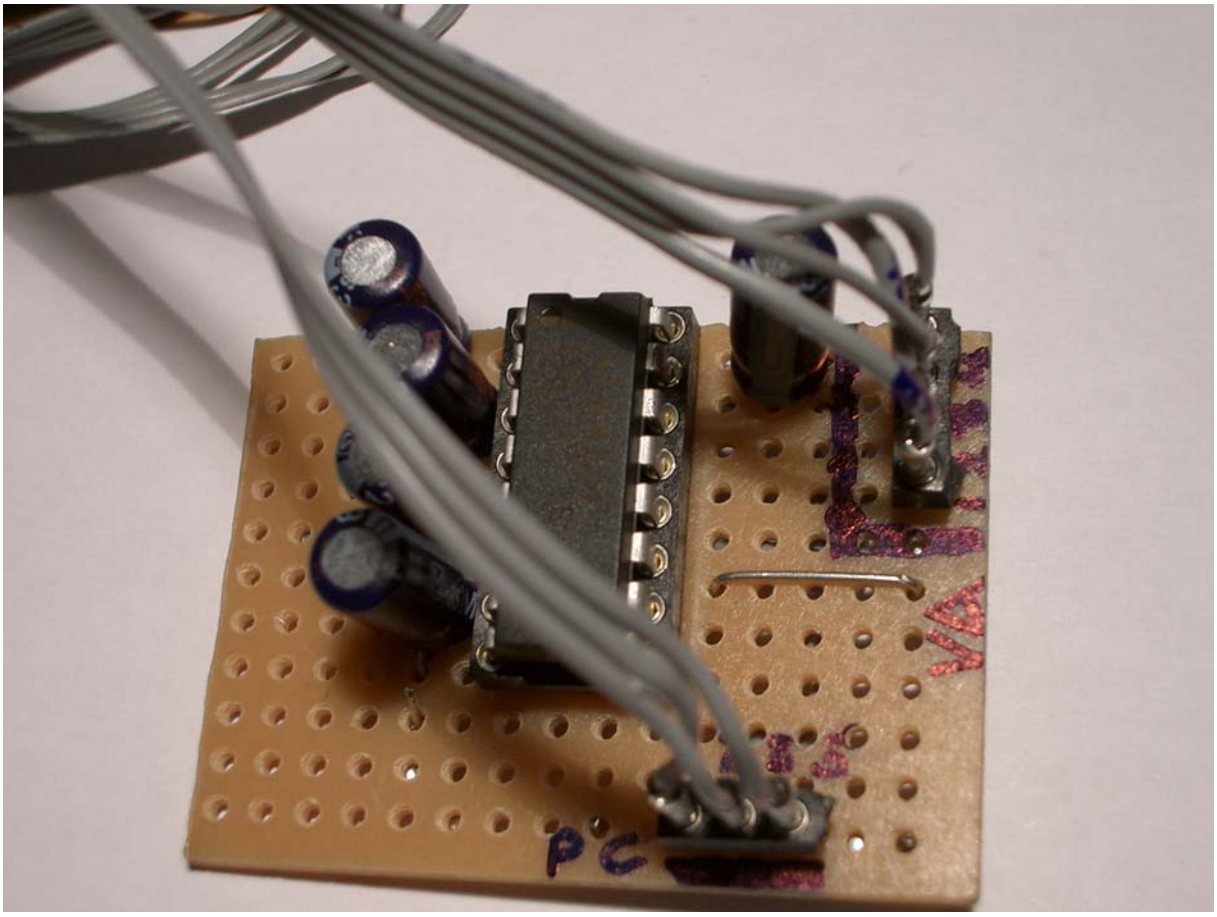
http://www.elektronik-kompendium.de/public/riederer/max232.htm .

Und so wird der MAX232 beschaltet:



Die 5 Kondensatoren sind 22µF-Elkos. Auf die richtige Polung achten! Der exakte Wert ist hier relativ unkritisch, in der Praxis sollte alles von ca. 10µF bis 50µF funktionieren. X2 ist ein 9-poliger Sub-D-Verbinder, female.

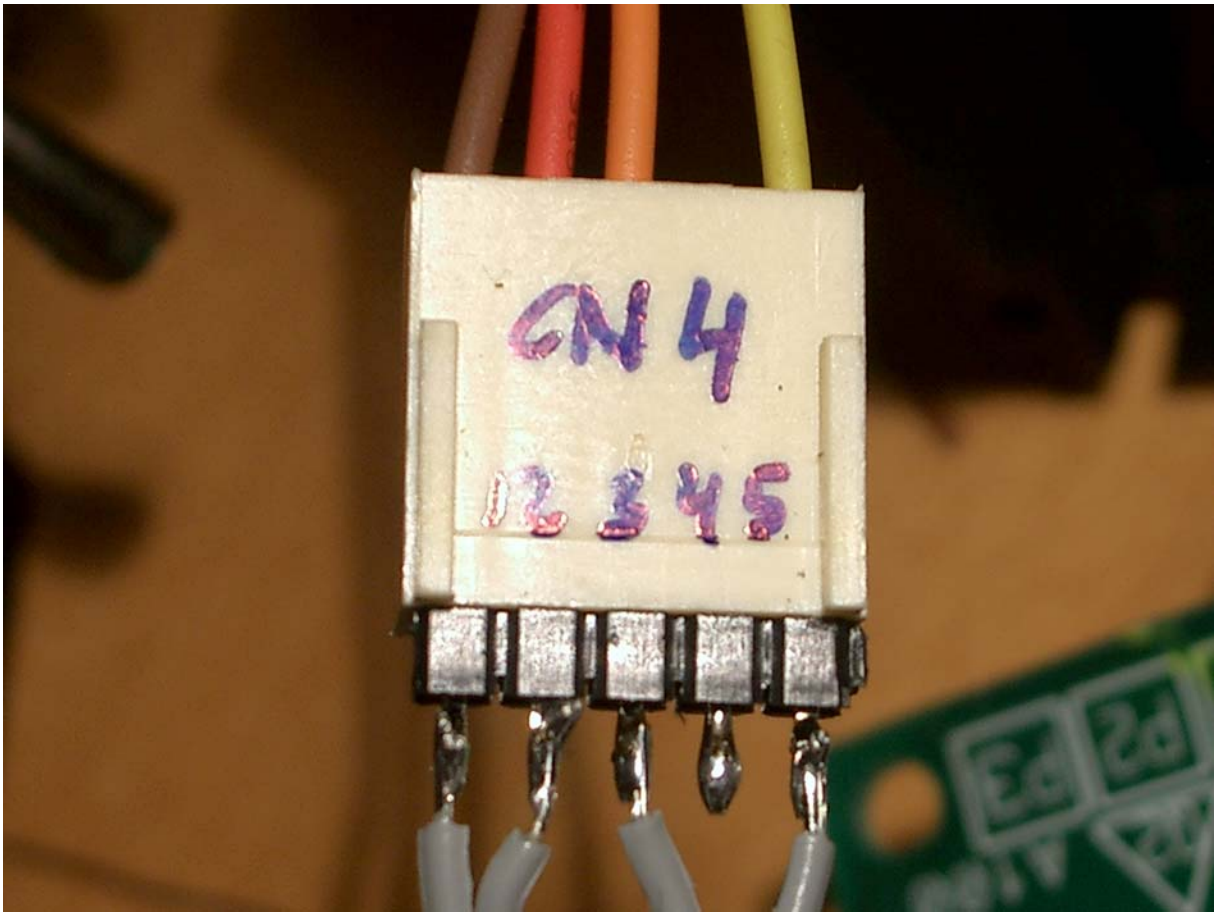
Die Schaltung lässt sich, eine ruhige Hand und etwas Löterfahrung auf einer Lochplatine in ein bis zwei Stunden aufbauen.



{Bild xyz – Pegelwandler auf Lochplatine}

Um flexibel den Versuchsaufbau zu verändern zu können, empfiehlt es sich, alle Zuleitungen steckbar zu gestalten. Wir haben hierfür einfach einen weiteren Präzisionssockel zerlegt und die Sockelstifte auf die Kabel aufgelötet.

Auch der Anschluss an den Serviceport wurde auf diese Art und Weise realisiert:



{Bild xyz – Anschluss an den Serviceport}

Auf der PC-Seite wird jetzt noch ein Programm benötigt, um die Parameter der RS232 – Schnittstelle einzustellen und um Signale an die Jura senden zu können. Hyperterminal (HYPERTERM.EXE) von Windows ist die einfachste Lösung und wird bei jeder Windows- Installation mit installiert.

Soweit, so gut. Aber wie geht es weiter? Die Antwort ist: Jetzt seid Ihr dran!.

Der Prozessor schweigt, wenn man mit ihm sprechen will. Wartet also auf das „Zauberwort“, um nicht unmotiviert los zu quatschen. Das ist bei derartigen Steuerungen wohl so üblich. Weitere Schwierigkeit: Wie sind die Kommunikationsparameter? Baudrate, Startbits, Stoppbits: Alles unbekannt.

Wie könnte es weiter gehen?

Man kann natürlich alle möglichen Bitkombinationen und alle möglichen Kommunikations-Parameter automatisiert durchprobieren. Das nennt man dann „Brute Force Attack“.
Langwierig, muss programmiert werden und das reine Bombardement ist solange unsinnig, wie nicht auch die Antwort ausgelesen wird. Also muss zunächst mal ein Programm her, das auch die Antwort vom Prozessor ausliest (soweit da wirklich was mitkommen sollte). Die Technik heisst „Man in the Middle“. Man suche unter „Serial Port Monitor“ mit Google und findet dann diverse Programme, die so was machen.
Zum Beispiel PORTMON.EXE (http://www.sysinternals.com).
Weitere Links zum Serial Port Monitoring

- http://www.kmint21.com/serial-port-monitor/ (Shareware)
- http://www.developer.com/net/cplus/article.php/633421 (Commspy)
- Alles zu serieller Kommunikation: http://www.lvr.com/serport.htm
- Serielle Kabelbelegung, u.a. T-Kabel http://www.airborn.com.au/rs232.html
- Google: Viele Links unter Suchbegriff „serial port monitor“

Weg 1 oder Weg2?

Das Auslesen des EEPROMS ist deutlich einfacher und führt schnell zu Ergebnissen. Der zweite Weg ist langwierig und vom Ausgang her ungewiss, hat jedoch einen gewissen Charme! Wer hat nicht schon immer davon geträumt, allemöglichen Parameter seiner Maschine auszulesen, mal die Heiztemperatur hochzudrehen oder die Mahlzeit zu verändern? Alles ist möglich!

Wir bieten allen, die daran interessiert sind sich an der Projektarbeit zu beteiligen, eine Plattform im Internet. Ihr findet das Projekt im Diskussionsforum von coffeemakers.de

Es wäre doch gelacht, wenn wird diese Herausforderung nicht gemeinsam schaffen würden. Jeder noch so komplizierte High-Tec BMW lässt sich Chip tunen, da sollte das doch bei einem simplen Kaffeeautomaten ein Kinderspiel sein…