Reengineering of a Historical Educational 4bit Microcomputer Trainer with Arduinos

"And now for something completely different"

Michael Wessel

2-2-2023

https://github.com/lambdamikel/Busch-2090

https://hackaday.io/lambdamikel

Why this talk? Some motivation

- **Disclaimer: this talk is nevertheless largely “AI off topic”**
  - EE / “Maker” focus
  - Reengineering of a historical educational 4bit microcomputer
  - Not entirely off topic: e.g., AI SBCs (Single Board Computers) like NVIDIA Jetson, …
  - We also had seminars about coffee and chocolate, so why not about Arduino tinkering?
  - **Also, Karen encouraged this presentation**
- **Why might you find it interesting, & what could you get out of this talk?**
  - Historical context / window to the past
    - Retro Computing: how was home computing in the late 70s / early 80s
    - To understand how we got where we are, to learn from the past
    - We are at a place where Computing History matters!
  - Personal road to computing
  - Motivation and pointers for your own Arduino tinkering ambitions
    - Pointers to different Arduinos & modules, EDA, KiCad, PCB services, …
    - **Maybe I can infect you with the Arduino / tinkering virus - Arduino is fun!**

Reengineering target:
My first computer - 1983
More soon

Box shows
a prototype board!

microtronic computer system 2090

Programmieren – ein faszinierendes Hobby!

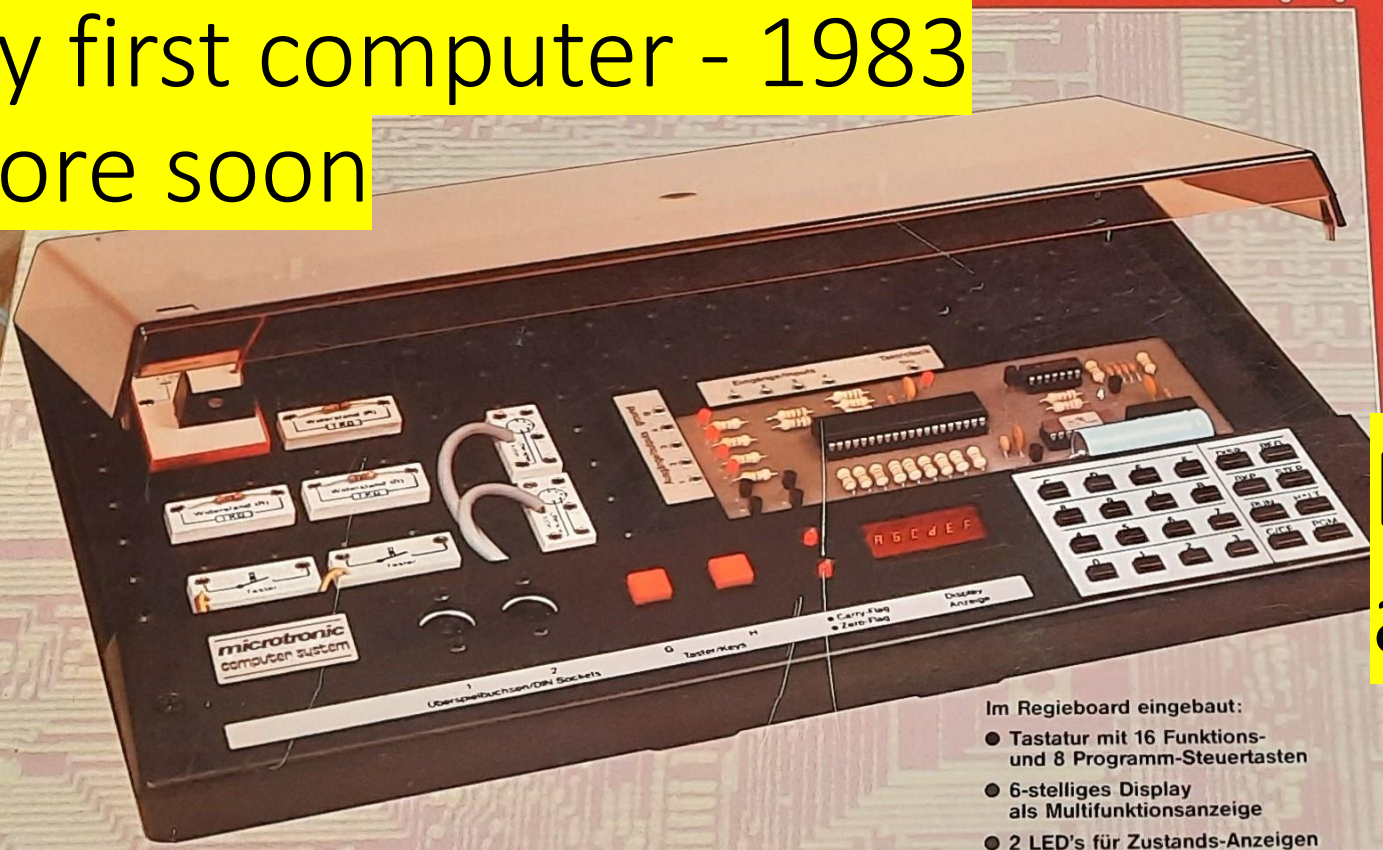
Das betriebsfertig installierte Micro-Computersystem mit den hunderttausend Möglichkeiten!

Herz des Computers: Der Microprozessor. Ein Schaltwerk mit ca. 35.000 Transistorfunktionen!

microtronic ist durch BUSCH-Electronic-Studios 2060, 2070 bzw. 2075 weiter ausbaufähig.

microtronic zeigt, wie ein Computer funktioniert. Wie man ihn programmiert. Wie er durch Befehlseingabe Daten anzeigt und speichert, zu Programmabläufen zusammenstellt und auf Abruf bestimmte Funktionen ausführt.





Im Regieboard eingebaut:

- Tastatur mit 16 Funktions- und 8 Programm-Steuertasten
- 6-stelliges Display als Multifunktionsanzeige
- 2 LED's für Zustands-Anzeigen
- 2 Sonderfunktionstasten
- 2 Normbuchsen für externe Geräte-Anschlüsse

und er ist Spiele-Partner. Er kann Reaktionszeiten, Temperaturen und Spannungen messen.

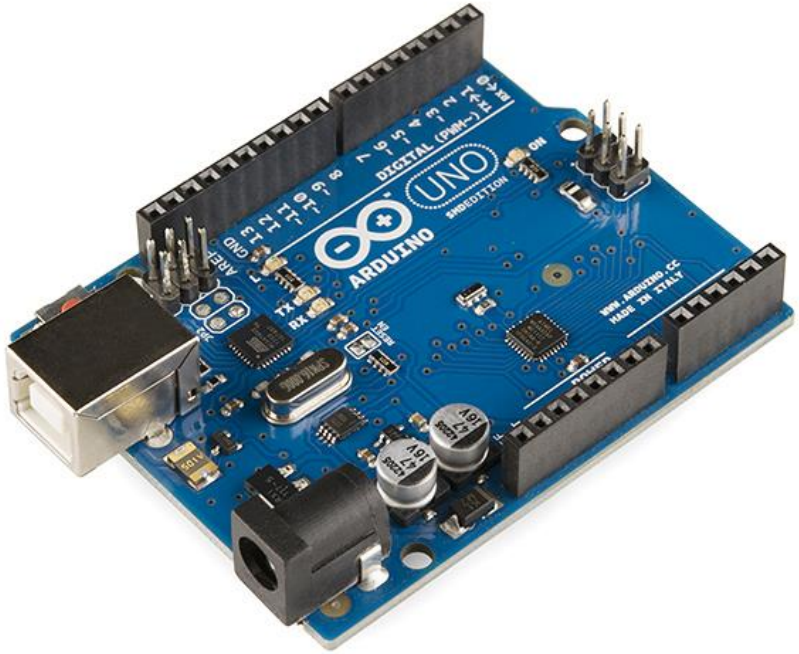
mier-Formulare, sowie versch. BUSCH-Electronic-Steckbausteine sind im Lieferumfang enthalten.

In Zusammenarbeit mit dem Elektronik-Magazin ELO

microtronic – Hobby der Zukunft, die bereits begonnen hat.

BUSCH

And this…



2005 | Timeline of Computer Hi

https://www.computerhistory.org/timeline/2005/

67%

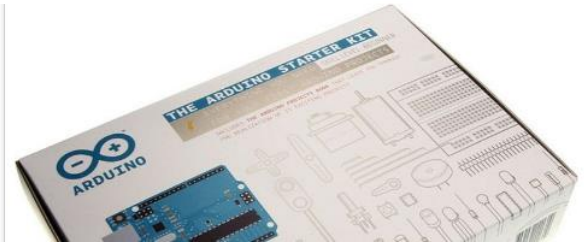
the bookmarks toolbar. Manage bookmarks...

EXPLORE CONNECT VISIT ABOUT

Timeline of Computer History

2000s

2000 2001 2002 2003 2004 2005 2006



Arduino

Computers

Harkening back to the hobbyist era o
project of the Interaction Design Insti
consisted of an inexpensive microcon
for use in any application connecting
Java-based integrated development environment and users could act

Blink | Arduino 1.8.5

Blink §

```
  This example code is in the public domain.

  http://www.arduino.cc/en/Tutorial/Blink
*/

// the setup function runs once when you press reset or power the board
void setup() {
  // initialize digital pin LED_BUILTIN as an output.
  pinMode(LED_BUILTIN, OUTPUT);
}

// the loop function runs over and over again forever
void loop() {$
  digitalWrite(LED_BUILTIN, HIGH);   // turn the LED on (HIGH is the voltage level)
  delay(1000);                       // wait for a second
  digitalWrite(LED_BUILTIN, LOW);    // turn the LED off by making the voltage LOW
  delay(1000);                       // wait for a second
}
```

32

Arduino/Genuino Uno on COM1



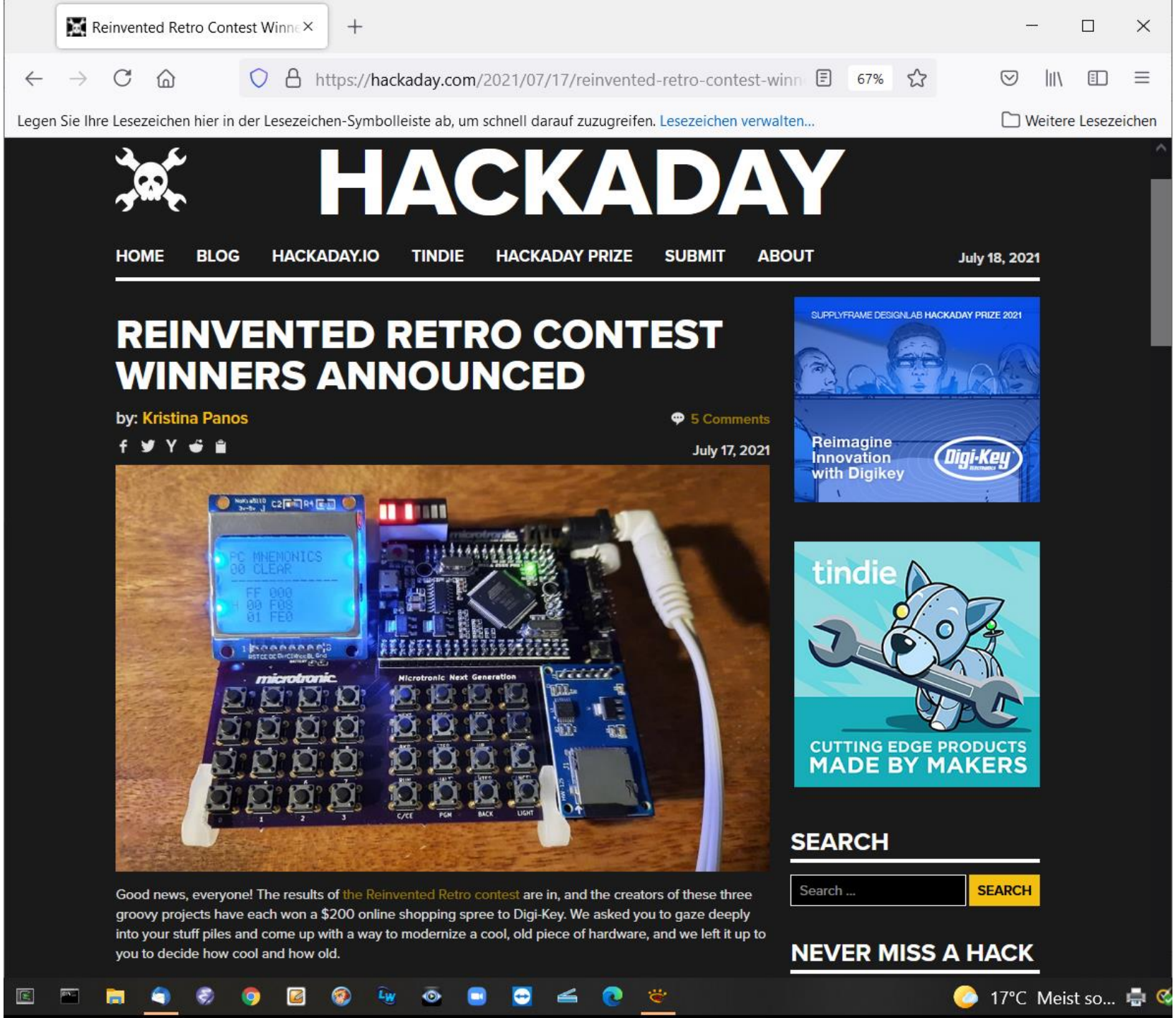
I started with an Arduino in 2016 like this… (BMOW = Big Mess of Wires)

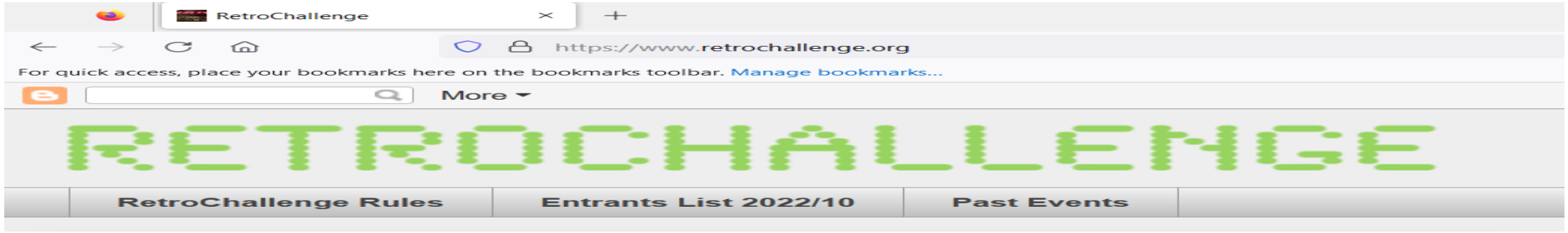
In 2019 I got “a team”!

This version was largely inspired by a team-designed version (on display). I’ll discuss the team-designed as well.

I submitted this to the Hackaday “Reinvented Retro Contest” in 2021 and won a prize.

I accepted the prize for the team and shared it with them.

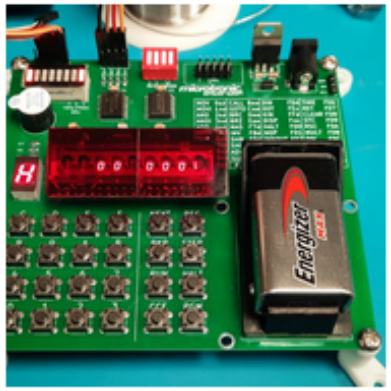




My final "solo" version – basically, back to the roots (LED 7segment display) but no more modules. SMD PCB.

A self-contained ATmega 328-based Single Board Computer.

Michael Wessel



Platform: Busch Microtronic

Project : A fully retro-authentic yet modern reimplemenation of the classic 1981 4bit educational single board computer. Hex in machinecode with a 7segment bubble display like its 1981!

Halftime Update: *MSW: one of the leading contenders this, the blog photos are beautiful and he's making real progress with some wonderful videos... definitely one to watch.*

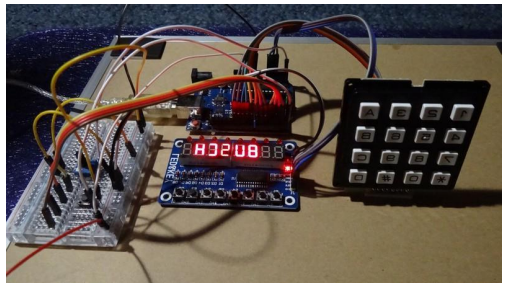
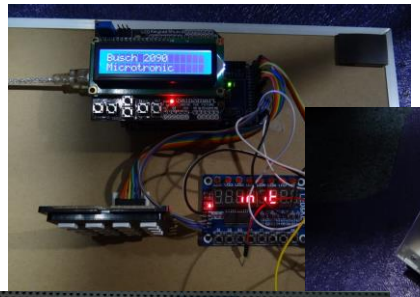
Final Comments: *4 bit microprocessor, MIDI syth, drum machine, bubble LED displays, great videos, and what really sold me: using his emulated Microtronic as a drum module for an original 1981 Busch 2090 Microtronic!! Awesome job!*
MSW: Can't beat that last video for wow factor, I need to include more drum modules in my projects!

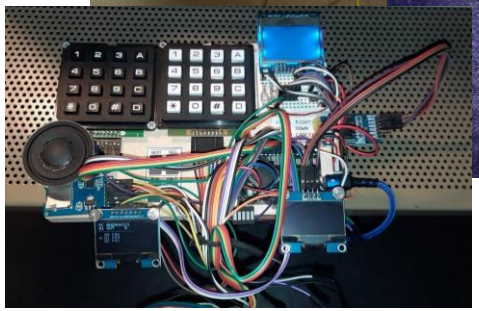
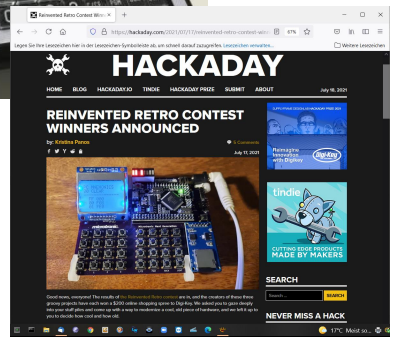
Final Status: *Prize "Winner"!!*

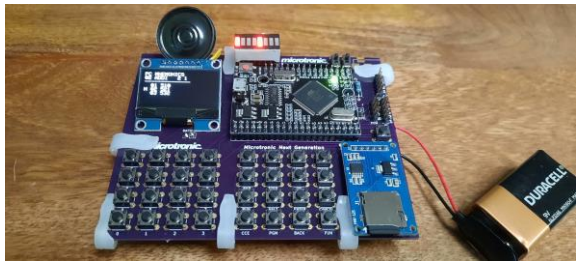
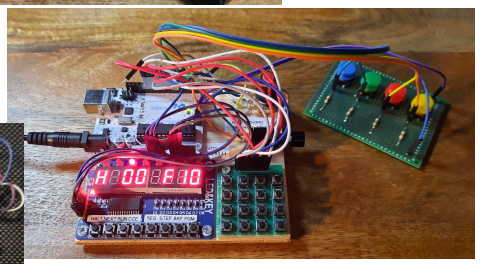
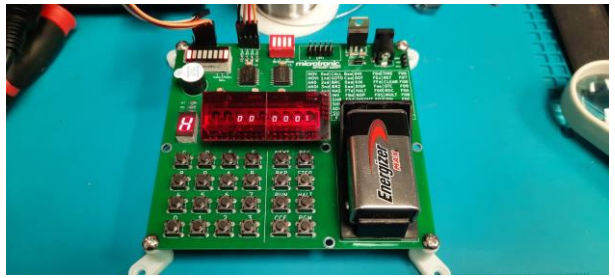
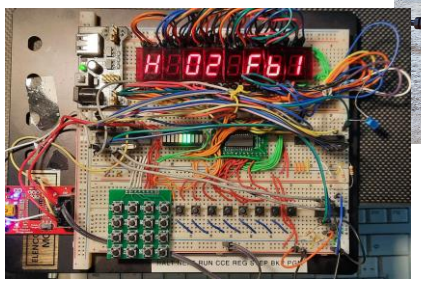
Outline

- **A “Show & Tell” Presentation**
  - MANY pictures ☺
- **The Whys – Motivation**
  - Why this talk
  - Why did I do this (actually, 10, not 7 times)
- **The Whats**
  - Some historical context
  - Microcontrollers
  - Homebrew computers
  - The Busch 2090 Microtronic
- **The Hows**
  - My “Arduino” journey
  - The 10 iterations in detail
  - From “Big Mess of Wires” (BMOW) to SMD (Surface Mount Device) PCBs (Printed Circuit Boards)

  -> A self-contained, module-less ATmega 328-based SBC (Single Board Computer)
- **Conclusion & Takeaways**

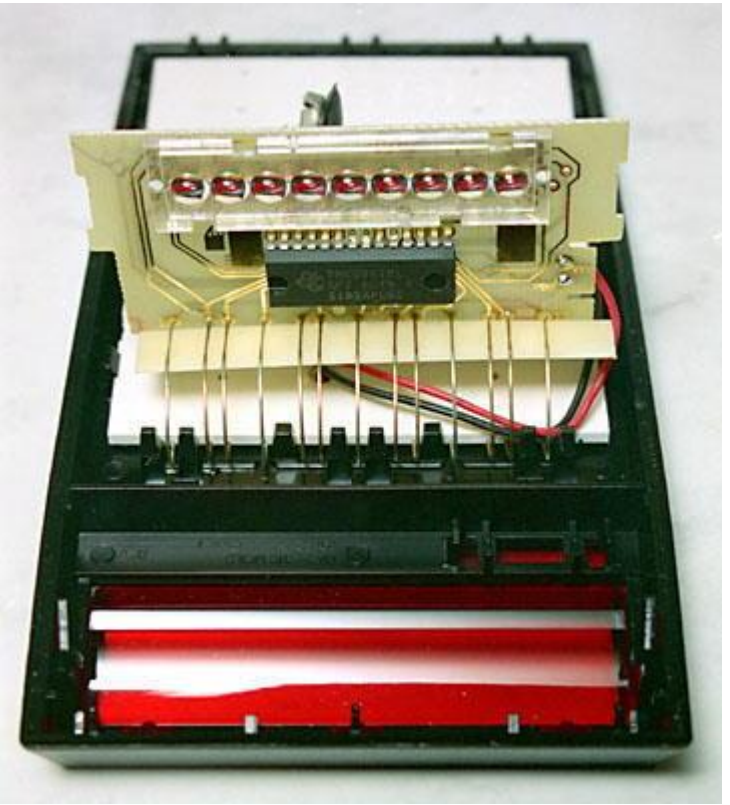





The Whys

Personal anecdote – How I got hooked by electronics

- Like Intel, TI got into the Microcontroller / CPU business over the calculator route
- As a kid, to the frustration of my parents, I used to take devices apart to learn what makes them tick
- Usually, I was able to learn something from that
- When I was 11, I opened the family **TI-30 (1976) calculator**
- I was **shocked** to find a **single chip** inside! How could a single chip be doing all that magic?
- **What did I learn? Nothing!**
- I felt **betrayed** - like opening a birthday present just to find **a locked box inside a box!**
- But it was also a **magical moment** – like finding an alien artifact that was beyond my comprehension
- **I knew that I needed to learn more about this kind of magic**





Why did I do this? 10 times…



Groundhog Day

Celebration

Groundhog Day is a popular North American tradition observed in the United States and Canada on February 2. Wikipedia

- **Nostalgia**
- **Educational aspects & self-improvement**
  - Doing Arduinos with the kids
  - Learning Physical Computing (Arduinos, MCUs, EE, EDA, FPGAs, …)
  - You need a non-trivial project as a driver
- **Ten versions**
  - **The Arduino Platform & Environment** offers a large design space
  - Different design iterations / versions that explore different aspects in the design space
  - Always one more idea to try, one more feature to add, one more thing to improve …
- **Fun**
  - Designing, tinkering, prototyping, building are very creative activities
  - Exploring different design options and modules is highly addictive - **like a kid in a candy store**
- **Exploring the “what ifs” - what if the machine had…**
  - … a real time clock, SDCard program storage, more memory, was faster
  - … speech or sound output, alphanumeric or graphical output
  - … could be used as a MIDI instrument

  -> How to realize these features in the “historical” architecture?

Why would your parents have bought this in the early 80s?

- **Affordable**
  - **DM 398** in **November 1981**, **DM 299** in **late 1983**
  - Average family income in Germany at that time: **32,198 DM**
    - **398 = ~ 1/80th**
    - **TRS 80 Model 1 Level 2 16 KB: 1995 DM = ~ 1/16th**
    - **VIC 20: ~ 798 DM**
    - **Atari 400: ~ 1298 DM**
  - Average family income in the US at that time: **$ 23,430**
    - **TRS 80 Model 1 Level 2 16 KB: $ 988 = ~ 1/27th**
  - Prices dropped significantly in 1983
    - Home computer price war / total market crash in December 1983



- **Educational & age appropriate**
  - Computers were the future!
  - Excellent age-appropriate manuals
  - **Focus on physical computing;** natural step-up from electronics (Science Fair) kits
  - No TV required, no video games (no “bad influences”)
  - **Build for physical computing & experimentation; less danger in doing damage to the ports etc.**

The TRS-80 System...



TRS-80 COMPUTERSYSTEME

Ab DM 100.– Auftragswert erfolgt die Lieferung porto- und verpackungsfrei. Lieferung per Nachnahme oder Vorauszahlung. **Alle Preise incl. Mwst.**



Zubehör für Modell III:
Disk Kit bestehend aus:
Double Density Controllerkarte, Diskettengerät, Netzplatine und Betriebssystem mit Handbuch **DM 2295.–** (Ohne Einbau)

Disk Kit 2 bestehend aus:
16K RAM Speicher und Diskettengerät **DM 1195,–** (ohne Einbau)

RS 232c Schnittstelle **DM 329,–**
Druckerkabel **DM 95.–**
Staubschutzhaube **DM 16.50**

TRS-80 Modell III mit:
❄ 32K RAM Speicher
❄ Level III BASIC
❄ Umlaute und Sonderzeichen nur **DM 3295.–**

TRS-80 Model III wie oben aber mit:
❄ 48K RAM Speicher
❄ Double Density Disklaufwerk (165K Byte Speicherplatz) nur **DM 5840.–**

TRS-80 Model III wie oben aber mit:
❄ 2 Double Density Disklaufwerken nur **DM 6995.–**

TRS-80 Modell III Computer sind bei uns mit bis zu 2.8 M Byte Diskettenspeicherplatz erhältlich (mit 5.25" Disketten!).

TANDY TRS-80 Mod. I Level II mit 16K RAM, Gross- Kleinschreibung, Monitor und 10er Tastatur **DM 1995,–**

TRS-80 Modell II:
- 80x24 Zeichen Video Display
- 8" Floppy Disk Laufwerk
- Zwei Z80A CPU's
- Multiuser faehig
- bis zu 320000 Bytes RAM Speicher
- zwei RS232c Schnittstellen

ab **DM 9995.–**

Fest-/Wechselplatte mit 10 M Byte fuer TRS-80 Mod. I + II, Apple II, S-100 u. a. incl. Controller **DM 16900.–** incl. Mwst.

Line Printer VI DM 2595,–
Line Printer V DM 3995,–

NEU!! LINE PRINTER VII
Erstmals koennen wir unseren Kunden einen Drucker mit diesem Preis-Leistungsverhaeltnis anbieten. Vergleichen Sie: Tractor und Einzelblattfuerung, 80 Zeichen/Zeile, 9½ Zoll Arbeitsbreite, durch Einzelpunktmatrix voll Grafikfaehig, Serielle und Parallele Schnittstelle nur **DM 995.–!**

Zubehör und Peripheriegeräte:
Level II Erweiterung **DM 325,–**
Expansion Interface mit 32K RAM Speicher **DM 1395,–**
Mini Disk 1: Einheit mit DOS, Manual und 4er Kabel **DM 1250.–**
Mini Disk 2..4: Einheit **DM 1150.–**
MINI DISK II mit 80 Tracks (204K-Byte) **DM 1450.–**
8 Zoll Disklaufwerk (Double Density) mit Gehäuse und Netzteil **DM 1995.–**
10er Pak Mini Disketten BASF Double Density **DM 95.–**
RS 232c Schnittstelle **DM 289.–**
Analog/Digital Wandler mit 8 Kanälen zur Messwerterfassung, Steuerung von Peripheriegeräten, Sprachsynthese usw. **DM 395,–**
Sprachprozessor zur Spracherzeugung und Spracherkennung **DM 395.–**
EPROM-Programmer für Ihren TRS-80! Liest und schreibt 2716-2732 EPROM's nur **DM 395.–**
Digital-Port Interface für Input-Output von 8 Bit Daten: 8 decodierte Ports. **DM 395.–**
Grafik Erweiterung zum Ansprechen 192x[illegible] **DM 445.–**
Data Sep[illegible]
Interface
Double D
Interface
Adapter z
Laufwerk
DAISY W
PLOTTE
mit A4 B

Software Hits für TRS-80 Computer:

	Tape	Disk
APL-80 Interpreter	DM 99,50	149,50
Level III BASIC	DM 149,–	149,–
BASIC-Compiler	DM 195,–	450,–
COBOL-Compiler	DM –	695,–
FORTRAN-Compiler	DM –	299,50
MACRO-Assembler	DM 129,–	299,50
Editor/Assembler	DM 99,50	109,50
muMATH/muSIMP	DM –	195,–
PASCAL Compiler	DM 99,50	395,–
SCRIPSIT Textverarbeitung	DM 174,50	245,–
Geschäftsadress-programm	DM 49,50	148,–
Inventory Control System (deutsch)	DM –	139,–
Wärmebedarfs-berechnung	DM –	74,50
Microchess	DM 49,50	49,50
SARGON II	DM 89,–	89,–

Modell II Software:
BASIC-Compiler incl. Macro-Assembler, Loader u.a. **DM 545,–**
FORTRAN-Compiler incl. Editor, Linking-Loader u.a. **DM 745,–**
MACRO-Assembler incl. Cross-Reference, Editor u.a. **DM 525,–**
COBOL-Compiler incl. Editor, Loader u.a **DM 795,–**

apple computer

Apple II u. III Die Profi-Systeme



Apple-II-80D-System
Der 80-Zeichen-Apple, Groß- u. Kleinschreibung, deutsche Zeichen, Grafik-Zeichen 3596.– DM

Profi-System 1
Apple II Europlus 48 KB, Disk mit DOS-3.3-Software, Daten-Monitor, 12" grün, 15 MHz
4796.– DM

Profi-System 2
Apple II Europlus 48 KB, Disk mit DOS-3.3-Software, Daten-Monitor, 12" grün, 15 MHz, MX-82-F/T-Grafik-Drucker
6996.– DM

VC-20 Commodore 789.– DM



In ihm steckt mehr als Sie vermuten

Jeder VC-20 + Demo-Kassette mit viel Software. Jetzt Erweiterungsmodul für 6 Steckplätze. Interface usw.

Kassettenrecorder	Supererweiterungsmodul
Floppy-Disk	Grafik und Musik
Matrix-Drucker	Handbücher 6502 (Hardware u. Software)
Speichererweiterung 3, 8, 16, 32 KB	Lerne Basic auf Kassette
Programmhilfemodul	Spiele: Autorallye. Super Copra
Maschinenspr.-Monitor	Luftkampf, Invasion v. Mars.
	17 + 4, Biorhythmus, Adreßdatei.

Fordern Sie die neueste VC-20-Preisliste an, es lohnt sich.

ATARI 400/800 neue Preise

Der Personal Computer mit der Super-PAL-Farbe.

Atari 400	1198.– DM
Atari 800	2396.– DM
Jetzt 32 K RAM Speichererweiterung (auf 48 K)	337.– DM

Programm Recorder	259.–	Software: Editor Assembler, Microsoft Basic,
Disketten-Station	1650.–	Musikcomposer, Graph IT, Space Invaders,
Grafik Tablet	975.–	Schach, Star Raiders, Zinsen u. Tilgung, Visi-
Drucker	998.–	calc, Calculator

Fordern Sie die neueste Atari-Preisliste an, es lohnt sich!

Neu! Der Sinclair ZX81 Personal-Computer.

Als Bausatz DM 298,-.
Die Fertigversion DM 398,-.



Erschließen Sie sich umfangreiches Computer-Verständnis. In wenigen, faszinierenden Stunden.

1980 – das Jahr eines einzigartigen Durchbruchs: Sinclair präsentiert mit dem ZX80 der Welt ersten Personal-Computer für unter Fünfhundert Mark. Mit bisher unübertroffenen Leistungsdaten.

Weltweit konnten über 50.000 Exemplare verkauft werden, und auch in der Fachwelt fand das Gerät höchst positive Resonanz.

Jetzt kann Sinclair diesen technologischen Vorsprung weiter ausbauen. Mit dem neuen ZX81.
Für nur DM 398,-.

Kleine Kosten-/ Nutzenrechnung.

ZX81 – das ist nach wie vor eine der einfachsten Möglichkeiten, Computer zu verstehen und mit ihnen zu arbeiten. Aber jetzt mit erweiterten Kapazitäten! Denn der Z 80-Mikroprozessor wurde mit dem neuen, noch leistungsstärkeren „8k Basic ROM" kombiniert. Zur „trainierten Intelligenz" des Computers.

Dieser Chip mit Dezimalzahlen, logarithmischen und trigonometrischen Funktionen

Als Bausatz oder die Fertigversion – wie Sie wünschen.

Die Abbildung veranschaulicht den unkomplizierten Zusammenbau des ZX81. Einfach die mit den 4 Chips und den anderen Bauelementen bestückte Platine anlöten – fertig. Das passende Netzteil (600 mA bei nom. 9 V) ist bei Bausatz und Fertigversion im Lieferumfang enthalten.

Das neue Basic-Handbuch. In deutscher Sprache.

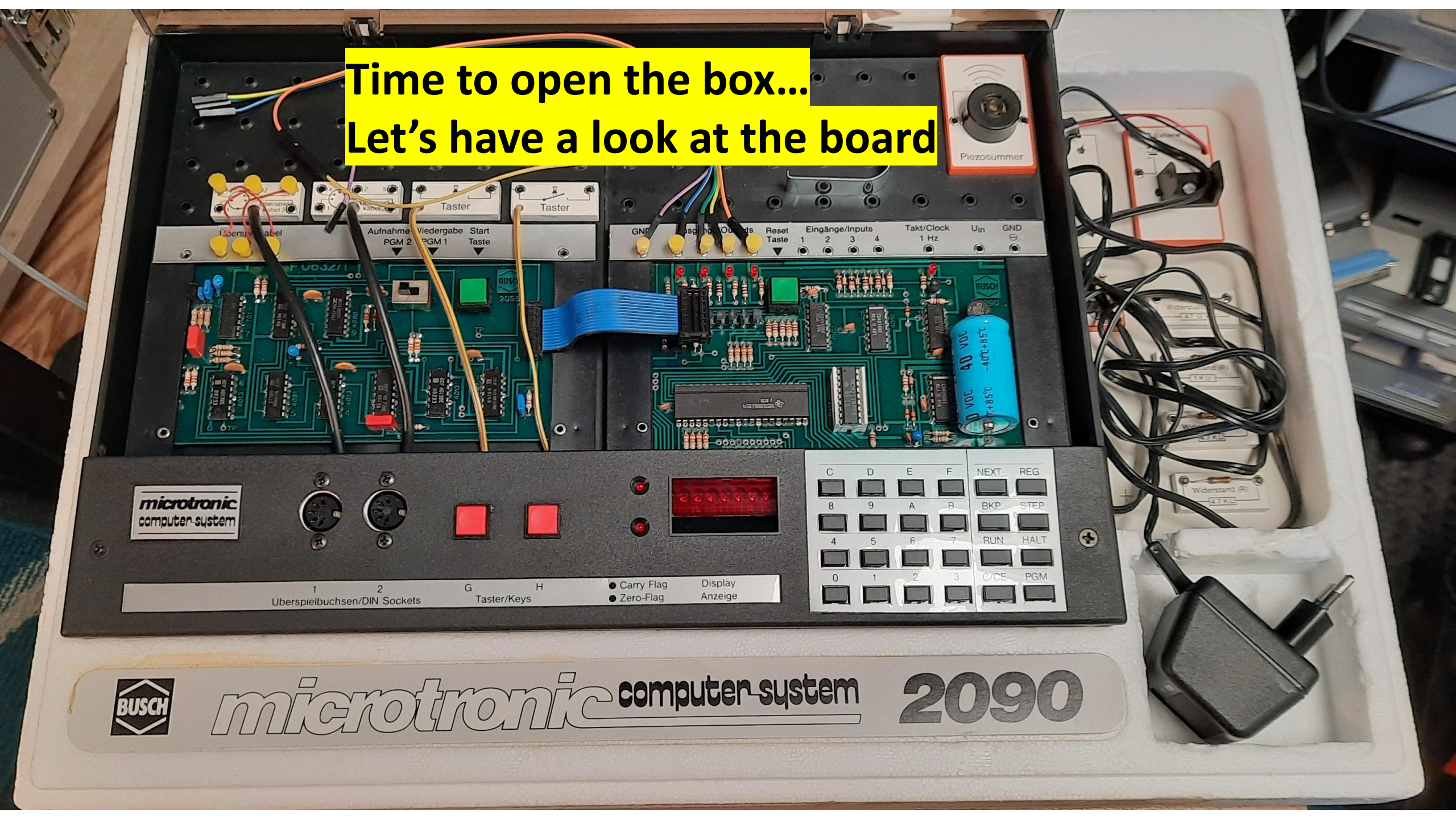


Fun fact:
RCA / SRI Princeton supplied the silver monitor (RCA B&W TV) and thereby determined the whole design for the 1977 TRS-80 Model 1!



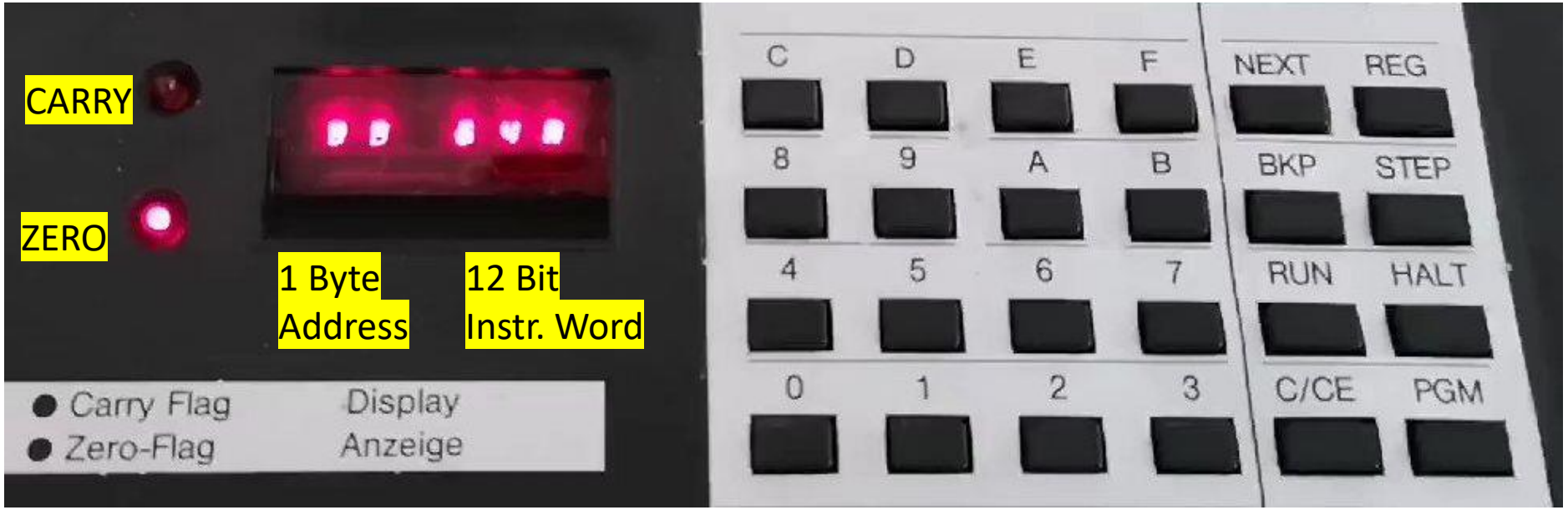
My TRS-80 Computers

The Whats


Time to open the box…
Let’s have a look at the board
Piezosummer
Taster
Taster
Aufnahme PGM 2
Wiedergabe PGM 1
Start Taste
Reset Taste
Eingänge/Inputs
1 2 3 4
Takt/Clock 1 Hz
Uin
GND
microtronic computer system
C D E F
NEXT REG
8 9 A B
BKP STEP
4 5 6 7
RUN HALT
0 1 2 3
C/CE PGM
1 2
Überspielbuchsen/DIN Sockets
G H
Taster/Keys
Carry Flag
Zero-Flag
Display
Anzeige
BUSCH
microtronic computer system 2090

Microtronic user experience & example program

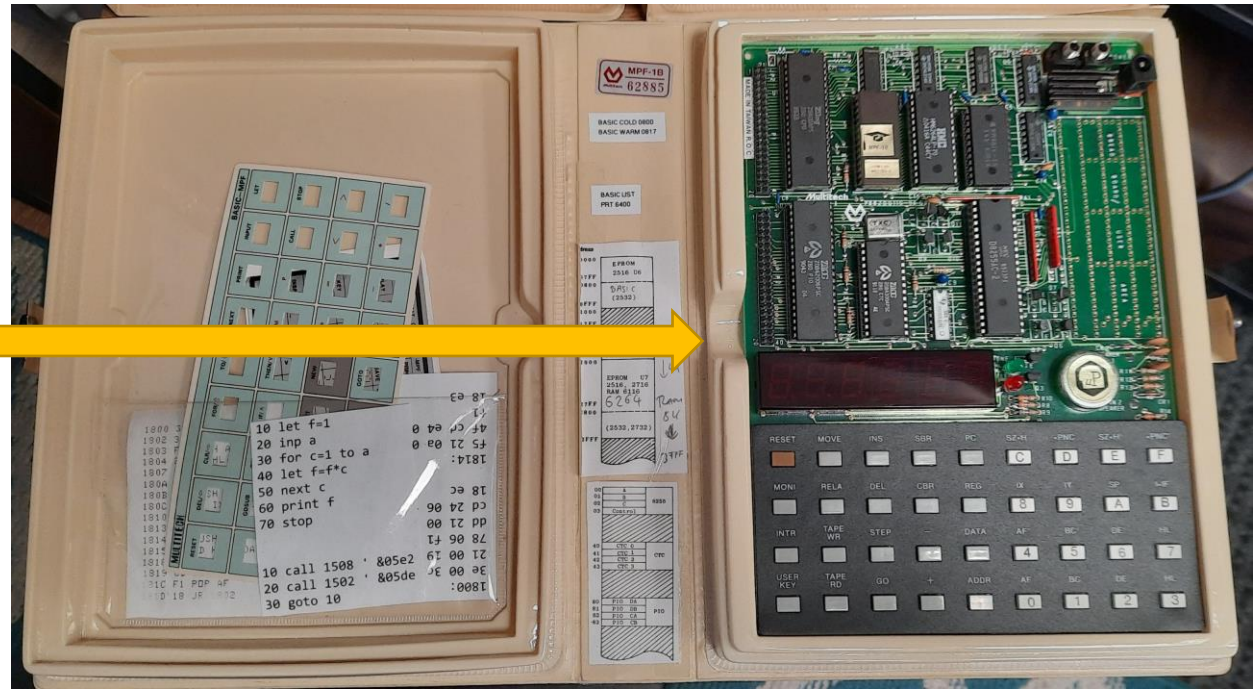
- **Monitor program allows to**
  - Enter / inspect / edit programs (**NEXT, C/CE**)
  - **RUN** and **HALT** programs
  - Inspect and modify registers (**REG**)
  - Set Breakpoints (**BKP**)
  - Single Stepping (**STEP**)
  - Some ROM programs (**PGM** – Load & Save, Nim Game, Clock, …)




Start
Taste
Reset
Taste
1
2
3
4
1 Hz
C
D
E
F
NEXT
REG
8
9
A
B
BKP
STEP
4
5
6
7
RUN
HALT
0
1
2
3
C/CE
PGM
G
H
Taster/Keys
Carry Flag
Zero-Flag
Display
Anzeige

Is it a toy?

- **Yes and no…**
- **Clever, not toy-like instruction set**
  - More details later
- **Look at the programs that are possible**
  - Almost everything that was achievable with a more powerful trainer (e.g., the Z80-based Microprofessor) is possible with the Microtronic as well
  - Largely due to a clever instruction set
  - **“Standard programs”** for SBCs of that time
    - NIM game
    - Biorhythms
    - Prime numbers (!)
    - Trigonometric calculations
    - Lunar Lander
    - Tic Tac Toe
    - Timers & clocks
    - **Electronics experiments (e.g., sound generator, …)**



Programm-Eingabe „Mondlandung"

Mit HALT – NEXT – 00 zum Programm-Anfang gehen und Befehlscodes lt. Tabelle eingeben. Nach jeder Befehlseingabe Taste NEXT nicht vergessen.

Adr. Nr.	Eingabe-Befehl			Befehlskürzel (Mnemonics)
00	F	0	2	DISOUT
01	F	0	8	CLEAR
02	F	E	0	DOT 0
03	1	4	2	MOVI 4,2
04	1	F	3	MOVI F,3
05	1	1	4	MOVI 1,4
06	1	2	5	MOVI 2,5
07	1	3	6	MOVI 3,6
08	1	8	7	MOVI 8,7
09	1	7	8	MOVI 7,8
0A	1	A	1	MOVI A,1
0B	0	2	D	MOV 2,D
0C	0	3	E	MOV 3,E
0D	0	4	F	MOV 4,F
0E	F	0	3	HXDZ
0F	F	5	D	DISP 5,D
10	F	F	B	KIN B
11	F	0	2	DISOUT
12	1	B	1	MOVI B,1
13	1	0	F	MOVI 0,F
14	0	5	D	MOV 5,D
15	0	6	E	MOV 6,E
16	F	0	3	HXDZ
17	F	5	D	DISP 5,D
18	F	F	B	KIN B
19	F	0	2	DISOUT
1A	1	C	1	MOVI C,1
1B	0	7	D	MOV 7,D
1C	0	8	E	MOV 8,E
1D	F	0	3	HXDZ
1E	F	5	D	DISP 5,D
1F	F	F	B	KIN B
20	1	0	D	MOVI 0,D
21	1	0	E	MOVI 0,E
22	F	2	D	DISP 2,D
23	F	F	B	KIN B
24	9	9	B	CMPI 9,B
25	D	2	9	BRC 29
26	0	D	E	MOV D,E
27	0	B	D	MOV B,D
28	C	2	2	GOTO 22
29	F	0	2	DISOUT
2A	1	0	F	MOVI 0,F
2B	F	0	4	DZHX
2C	6	D	7	SUB D,7
2D	F	C	8	SUBC 8
2E	D	6	9	BRC 69
2F	6	E	8	SUB E,8
30	D	6	9	BRC 69
31	7	5	D	SUBI 5,D
32	F	C	E	SUBC E
33	D	5	A	BRC 5A
34	4	D	2	ADD D,2
35	F	B	3	ADC 3
36	F	B	4	ADC 4

Adr. Nr.	Eingabe-Befehl			Befehlskürzel (Mnemonics)
37	4	E	3	ADD E,3
38	F	B	4	ADC 4
39	6	5	2	SUB 5,2
3A	F	C	3	SUBC 3
3B	F	C	4	SUBC 4
3C	D	7	B	BRC 7B
3D	6	6	3	SUB 6,3
3E	F	C	4	SUBC 4
3F	D	7	B	BRC 7B
40	6	D	5	SUB D,5
41	F	C	6	SUBC 6
42	D	8	0	BRC 80
43	6	E	6	SUB E,6
44	D	8	0	BRC 80
45	9	0	4	CMPI 0,4
46	D	0	A	BRC 0A
47	9	0	3	CMPI 0,3
48	D	0	A	BRC 0A
49	9	5	2	CMPI 5,2
4A	D	0	A	BRC 0A
4B	9	0	6	CMPI 0,6
4C	D	0	A	BRC 0A
4D	9	5	5	CMPI 5,5
4E	D	0	A	BRC 0A
4F	1	E	0	MOVI E,0
50	1	E	1	MOVI E,1
51	1	E	2	MOVI E,2
52	1	E	3	MOVI E,3
53	1	E	4	MOVI E,4
54	1	E	5	MOVI E,5
55	1	F	6	MOVI F,6
56	F	E	6	DOT 6
57	F	6	0	DISP 6,0
58	F	F	0	KIN 0
59	C	0	0	GOTO 00
5A	6	D	F	SUB D,F
5B	6	F	2	SUB F,2
5C	F	C	3	SUBC 3
5D	F	C	4	SUBC 4
5E	D	7	B	BRC 7B
5F	6	5	2	SUB 5,2
60	F	C	3	SUBC 3
61	F	C	4	SUBC 4
62	D	7	B	BRC 7B
63	6	6	3	SUB 6,3
64	F	C	4	SUBC 4
65	D	7	B	BRC 7B
66	4	F	5	ADD F,5
67	F	B	6	ADC 6
68	C	4	5	GOTO 45
69	1	E	0	MOVI E,0
6A	1	A	1	MOVI A,1
6B	1	E	2	MOVI E,2
6C	1	A	3	MOVI A,3
6D	1	F	F	MOVI F,F

Adr. Nr.	Eingabe-Befehl			Befehlskürzel (Mnemonics)
6E	1	F	E	MOVI F,E
6F	F	E	F	DOT F
70	7	1	E	SUBI 1,E
71	D	7	3	BRC 73
72	C	7	0	GOTO 70
73	F	4	0	DISP 4,0
74	F	E	F	DOT F
75	1	0	D	MOVI 0,D
76	F	E	D	DOT D
77	7	1	E	SUBI 1,E
78	D	5	8	BRC 58
79	F	0	2	DISOUT

Adr. Nr.	Eingabe-Befehl			Befehlskürzel (Mnemonics)
7A	C	7	3	GOTO 73
7B	1	A	0	MOVI A,0
7C	1	A	1	MOVI A,1
7D	1	A	2	MOVI A,2
7E	1	A	3	MOVI A,3
7F	C	6	D	GOTO 6D
80	1	F	0	MOVI F,0
81	1	A	1	MOVI A,1
82	1	F	2	MOVI F,2
83	1	A	3	MOVI A,3
84	C	6	D	GOTO 6D

Damit das Spielende auch akustisch wahrgenommen werden kann, muß noch der Piezo-Summer (siehe auch Seite 7) angeschlossen werden. Wenn wir nach der letzten Befehlseingabe nochmals die Taste NEXT betätigt haben, gehen wir wie üblich mit HALT – NEXT – 00 zum Programm-Anfang und starten mit RUN. Das Display muß jetzt die Meldung bringen: A0 500. Sollte der Computer diese Meldung nicht bringen (oder auf dem Display z. B. überhaupt nichts anzeigen) hat sich bei der Programm-Eingabe ein Fehler eingeschlichen. Wir erinnern uns, daß wir die Programm-Eingabe überprüfen können, und wir gehen nochmals mit HALT – NEXT – 00 zum Programm-Anfang und wir sehen unter Adresse 00 den ersten eingegebenen Befehl F02. Immer wenn wir die NEXT-Taste betätigen, wird uns der nächste Befehl angezeigt. Entdecken wir einen Fehler, betätigen wir zweimal die Taste C/CE, wodurch der falsche Befehl gelöscht und der richtige Befehl eingegeben werden kann. Mit HALT – NEXT – 00 und RUN wird das Programm neu gestartet. Sollte immer noch nicht die richtige Meldung kommen, muß das ganze Programm wie beschrieben kontrolliert und richtiggestellt werden.



Wichtige Information bezüglich Netzstromanschluß!

Wir haben erstmals mit einem gewissen Zeitaufwand ein längeres Programm in unserem Computer gespeichert. Wenn wir unsere „Mondlandung" einem Bekannten vorführen möchten, darf **die Zuleitung zur Netzsteckdose nicht unterbrochen werden.** Der Computer speichert das eingegebene Programm nur solange, bis die Stromzufuhr unterbrochen wird. Ohne Strom kann der Speicher des Computers die Informationen nicht behalten.

Wir können unseren Computer jedoch unbesorgt im Dauerbetrieb an der Netzsteckdose angeschlossen halten, ohne mit einer Beschädigung oder Abnutzung rechnen zu müssen. Der Stromverbrauch ist minimal (ca. 4 Watt) d. h., daß der Computer in 24 Stunden weniger Strom verbraucht, als eine 100 Watt

Haushalts-Glühlampe pro Stunde. Wir können den Computer in den Leerlauf-Zeiten, z. B. als Digital-Leuchtuhr weiterlaufen lassen, in dem wir uns mit den Tasten HALT – PGM – 4 die Uhrzeit anzeigen lassen. Wir erinnern uns, daß für die Uhrzeit der „Eingang 4" mit „Takt/Clock" auf der Computer-Platine verbunden sein muß. Eine Verbindungsleitung zum Piezosummer klemmen wir ab, weil sonst im Sekundentakt ein Pfeifton entsteht.

Übrigens, wenn wir das Gerät bereits einige Stunden im Betrieb haben, ohne daß die Uhrzeit eingestellt wurde (jedoch unter der Voraussetzung, daß die Verbindungsleitung zwischen „Eingang 4" und Takt/Clock vorhanden war) zeigt uns der Computer mit PGM 4 an, wie lange das Gerät seit der letzten Netzunterbrechung in Betrieb gewesen ist. Wir haben einen automatischen Betriebsstundenzähler!

Wir erinnern uns auch, daß wir mit HALT – PGM – 3 die Uhrzeit neu einstellen können (und diese mit HALT – PGM – 4 anzeigen). Mit HALT – NEXT – 00 – RUN können wir jederzeit (wenn die Netzleitung inzwischen nicht unterbrochen wurde) wieder unsere Mondlandung aufrufen.

Lunar Lander Program

Probieren und experimentieren?!

Wir haben inzwischen so viel gelernt, daß wir ohne Angst (etwas kaputt zu machen) experimentieren können. Wir sollten ruhig versuchen, mit den bis jetzt bekannten Befehlen, die ersten eigenen Programm-Ideen zu verwirklichen. Wir können auch eines der im zweiten Teil des Anleitungsbuches enthaltenen größeren Programme ausprobieren.

Wenn wir ein neues Kapitel durchgearbeitet haben, sollten wir unbedingt Variationen der vorgegebenen Programm-Vorschläge ausprobieren. Auf diese Weise werden wir schnell mit unserem Computer vertraut. Wir wissen, daß wir mit HALT – NEXT – 00 jeweils zu der zuerst eingegebenen Programm-Adresse kommen, und daß wir das Programm mit RUN starten. Mit HALT kann das Programm jederzeit abgebrochen werden. Wenn wir z. B. unser zuvor eingegebenes Programm „Mondlandung" im Computer-Speicher weiter halten möchten (keine

Kid-friendly illustrations

Beim Programm „Mondlandung" haben wir den letzten Befehl unter Adresse 84 mit C6D eingegeben. Wenn wir die Tabelle „Gegenüberstellung dezimales/hexadezimales Zahlen-System" auf Seite 17 betrachten, erkennen wir, daß mit dem Programm „Mondlandung", welches mit der hexadezimalen Ziffer 84 endet (= dezimal Nr. 132) erst etwa die Hälfte der Speicherkapazität unseres Computers belegt wurde. In unserem Speicher ist also noch genügend Platz, um weitere Programme einzugeben. Es ist zweckmäßig, einige Adressen-Nummern freizulassen und das folgende Programm mit der Adressen-Nummer 90 zu beginnen. In diesem Fall sagen wir dem Computer nicht wie bisher HALT – NEXT – 00, sondern HALT – NEXT – 90. Das nächste Programm beginnt also bei Adresse 90 und wenn wir dieses Programm aufrufen möchten, lautet logischerweise der Befehl HALT – NEXT – 90. Die Inbetriebnahme wie üblich mit RUN.

Für das im nächsten Kapitel beschriebene Experiment benötigen wir 14 Programm-Schritte, d. h., wenn wir mit der hexadezimalen Adresse 90 beginnen, wird die letzte belegte Adresse 9D sein. Wenn wir wieder unser „Mondlandespiel" aufrufen möchten, erreichen wir dieses mit HALT – NEXT – 00 – RUN, während wir das neu eingegebene Programm mit HALT – NEXT – 90 aufrufen.

Zwei wichtige Informationen sollten wir uns noch merken:

Gemäß Tabelle „dezimales/hexadezimales Zahlen-System" können wir bis zu 255 Programm-Schritte (Adressen) speichern, d. h. bis zur hexadezimalen Adresse FF. Würden wir ein sehr langes Programm eingeben, wodurch die zur Verfügung stehenden 255 dezimalen Programm-Schritte nicht ausreichen, springt unser Computer bei der automatischen Adressen-Vergabe nach der hexadezimalen Ziffer FF automatisch zum Programm-Anfang 00 zurück. Ein dort bereits eingegebenes Progamm wird gelöscht.

Wir können die Speicherkapazität des Computers bis zur Adresse FF ausnutzen. Wenn wir uns merken, bei welcher Adresse die verschiedenen Programme beginnen, können wir alle Adressen zwischen 00 und FF belegen. Außerdem steht uns jederzeit mit HALT – PGM – 4 die Uhrzeit zur Verfügung. Rufen wir jedoch mit PGM – 7 das Nimm-Spiel auf, so wird der Programm-Ablauf für dieses Spiel automatisch in unserem Programm-Speicher beginnend mit Adresse 00 eingespielt. Mit HALT – NEXT – 00 – RUN, können wir jetzt wieder das Nimm-Spiel durchführen. Da wir jedoch für das Nimm-Spiel die Speicherkapazität bis zur hexadezimalen Adresse 44 bele-
ndlandung" ge-
n verbliebenen
len, wird unser
ch weiterrech-
rch den fehlen-
d.

Abschließend soll noch erwähnt werden, daß zur Übersichtlichkeit bei allen Programmen grundsätzlich die Start-Adresse 00 genannt wird. Es bleibt uns vorbehalten, evtl. mit einer anderen Adressen-Nummer (z. B. 90) zu starten, so daß wir dieses neue Programm mit HALT – NEXT – 90 – RUN aufrufen können.

Vergleichen – ein wichtiger Befehl!

Bei den bisherigen Demonstrations-Programmen hat es uns vermutlich immer wieder gestört, daß der Computer mit dem ungewohnten hexadezimalen Zahlen-System arbeitet.

Mit dem folgenden automatischen Zähler möchten wir erreichen, daß dieser mit Dezimalzahlen arbeitet, d. h., daß nach der Ziffer 9 nicht wie bisher üblich mit A, B, C, D. . ., sondern mit

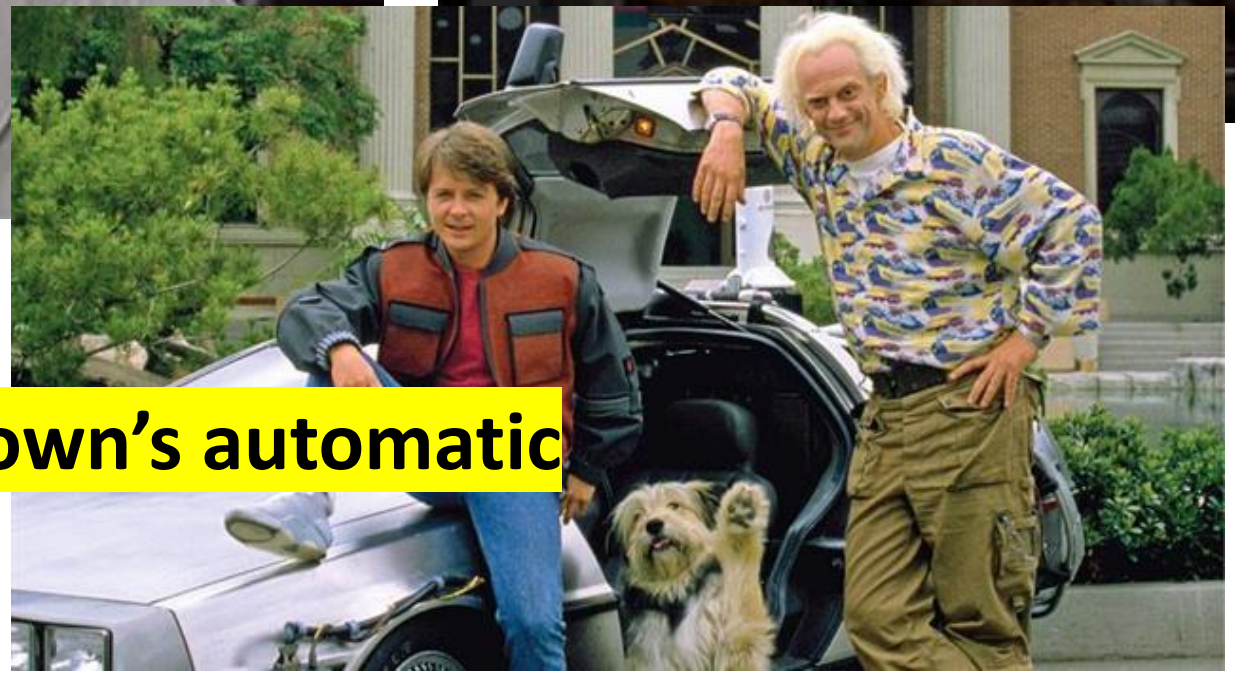
ezimal arbei-
azu bringen,
Mit HALT –

Adresse	Eingabe-Befehl	Mnemonik
00	**F08**	CLEAR
01	**F21**	DISP 2,1
02	**510**	ADDI 1,0
03	**FB1**	ADC 1
04	**991**	CMPI 9,1
05	**D07**	BRC 07
06	**C02**	GOTO 02
07	**561**	ADDI 6,1
08	**512**	ADDI 1,2
09	**992**	CMPI 9,2
0A	**DOC**	BRC 0C
0B	**C02**	GOTO 02
0C	**562**	ADDI 6,2
0D	**C02**	GOTO 02

Programm-Start: HALT – NEXT – 00 (oder 90) – RUN
Wir sehen, unser Computer zählt nun treu und brav von 0-99, um dann wieder von vorne zu beginnen. Die hexadezimalen Werte A-B-C-D-E-F sind nicht mehr vorhanden.

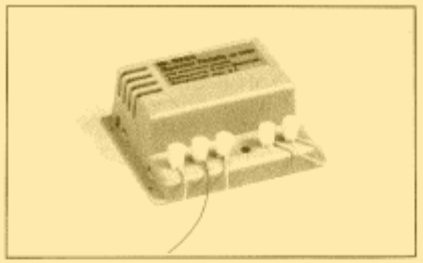






Recreate Doc Brown’s automatic dog feeder?

BUSCH-Schwachstrom-Schaltrelais Nr. 5964

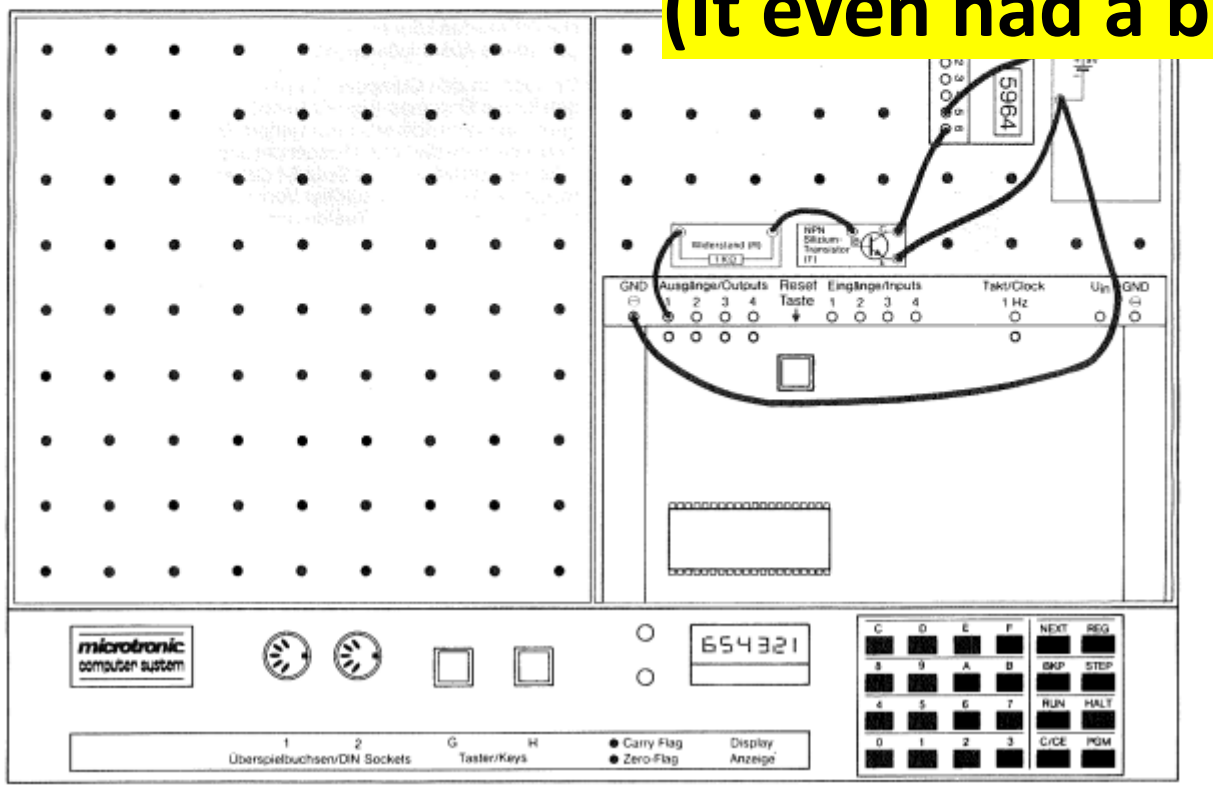


Dieses Relais wird u. a. auch im BUSCH-Modellbahn-Zubehör-Programm angeboten. Es ist für vielseitige Schalt- und Steuerfunktionen verwendbar.

Dieser Zustand bleibt solange erhalten, bis am Ausgang des Computers „keine Spannung" mehr vorhanden ist. Das Relais springt zurück: Schaltkreis 1 wird eingeschaltet – Schaltkreis 2 ausgeschaltet. An den Kontaktstellen des Relais können die entsprechenden Stromkreise (Schaltkreise) angeschlossen werden. Dem Relais liegt eine genaue Beschreibung bei.

Bei Experimenten z. B. in Verbindung mit einer Modellbahn, ist unbedingt darauf zu achten, daß in keinem Fall stromführende Teile mit einem Computer-Anschluß in Verbindung kommen. Die Modellbahn-Trafospannung 16 V Wechselstrom führt unweigerlich zur Zerstörung einzelner Komponenten.

Anschluß des Relais 5964 an einen Computer-Ausgang



Da jeder Computer-Ausgang durch entsprechende Programmierung einzeln angesteuert werden kann, ist der Anschluß von 4 Relais möglich. Zu jedem Relais ist jeweils ein Widerstand 1 KΩ und ein Transistor erforderlich.

BUSCH-Netzstrom-Schaltgerät 2087



Neben vielen anderen Einsatzmöglichkeiten kann der Computer in Verbindung mit BUSCH-Elektronik-Studios und Schaltgerät 2087 z. B. während eines Ferien-Aufenthaltes die Anwesenheit der Bewohner eines Hauses vortäuschen. Ein Fotowi- ine Beleuch- altet. Eine ge- übergehende räuschpegel rte Computer sschalt-Vorgänge stellbar sind. Eine bald die Tür geöff-

Die folgenden Programm-Beispiele sollen lediglich Anregungen geben. Es würde den Rahmen dieses Anleitungsbuches übersteigen, wenn alle Anwendungsmöglichkeiten aufgezählt und erläutert werden sollten. Den Besitzern zusätzlicher BUSCH-Electronic-Studios eröffnen sich unzählige Möglichkeiten.

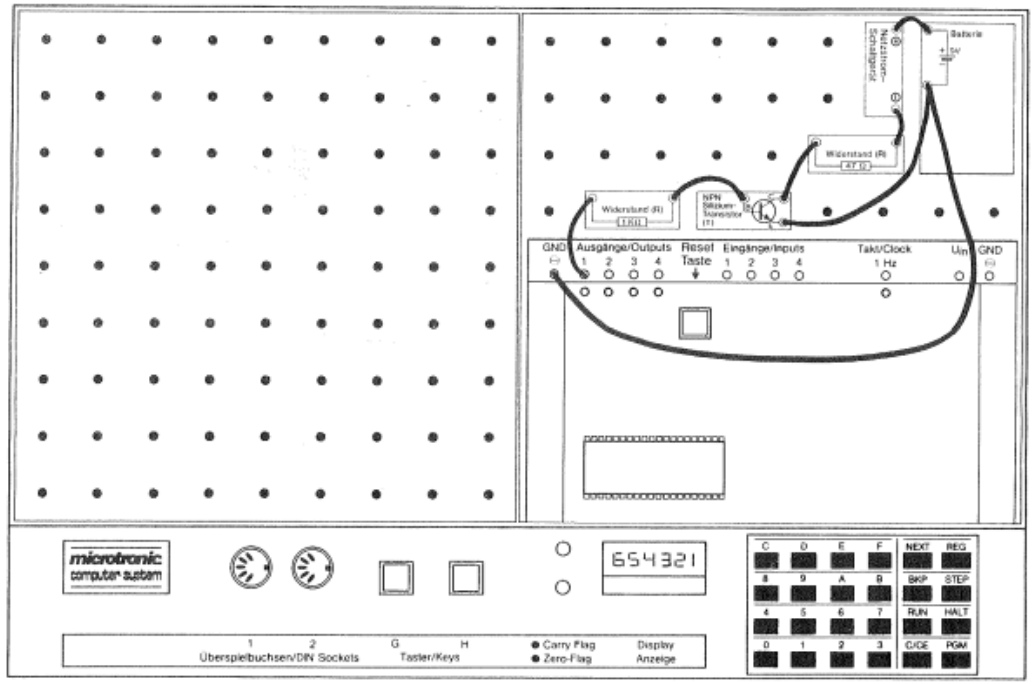
Auch dieses Schaltgerät kann mit Schwachstrom (6-9 Volt) angesteuert werden. Es löst über ein Relais Schaltvorgänge im Lichtstromnetz (220 V, max. 1000 Watt) aus. Mit zwei Widerständen und einem Transistor (siehe Abbildung) wird das Schaltgerät an einem Computer-Ausgang angeschlossen.

Das Schaltgerät 2087 entspricht den VDE Vorschriften. Es ist absolut gefahrlos einzusetzen, weil der 220 V Netzstromteil von der Schwachstromseite völlig getrennt ist.

Das Schaltgerät 2087 ist ein „Ein-/Ausschalter". Sobald eine geringe Steuerspannung vorhanden ist, schließt das Relais einen 220 V Schaltkreis. Dieser Stromkreis wird wieder abgeschaltet, wenn die Steuerspannung unterbrochen wird. Das Relais ist für 20 Millionen Schaltvorgänge ausgelegt und kann im Dauereinsatz betrieben werden. Es dient zum Ein- und Ausschalten von Radios, Tonbandgeräten, Warnanlagen, Heizlüftern und allen elektrischen Geräten mit einem Anschlußwert bis zu 1000 Watt. Werden Geräte mit höherem Anschlußwert geschaltet, muß mit einer erheblichen Verringerung der Schaltleistung gerechnet werden, es ist deshalb z. B. beim Anschluß von Heizlüftern usw. unbedingt darauf zu achten, daß die 1000 Watt Grenze nicht überschritten wird.

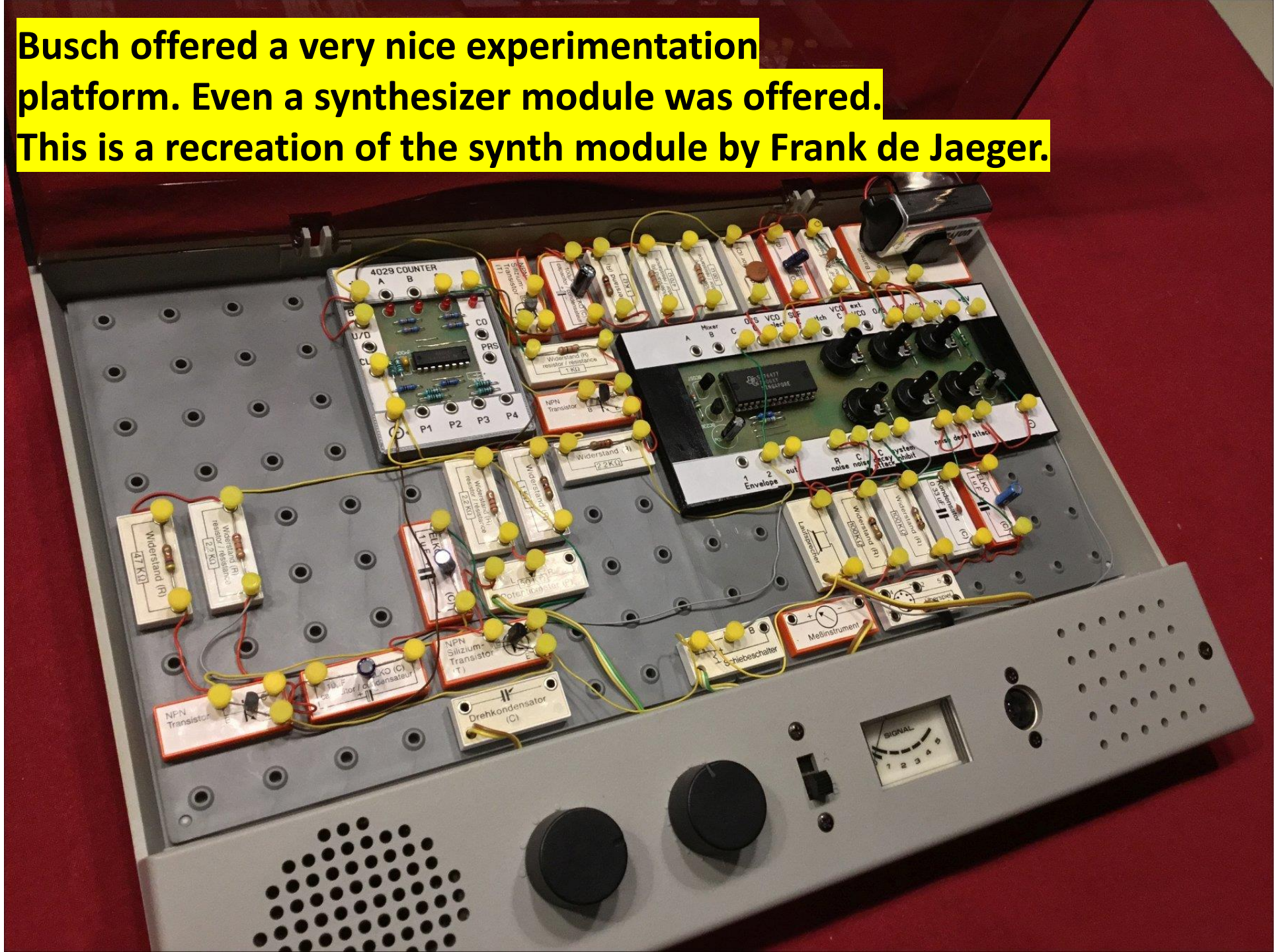
Bis zu 4 Netzstrom-Schaltgeräte 2087 können an den Computer-Ausgängen angeschlossen werden. Für jedes Schaltgerät sind zwei Widerstände und ein Transistor erforderlich.

Anschluß des Netzstrom-Schaltgeräts 2087 an einen Computer-Ausgang



Microtronic got you covered!
(It even had a built-in clock)

Busch offered a very nice experimentation platform. Even a synthesizer module was offered. This is a recreation of the synth module by Frank de Jaeger.



2016 - Exploring Arduinos

- **Arduino is the Microtronic of today**
  - Mostly programmed in C / C++ instead of Machine Language
  - Other languages available as well
- **Started with the kids as a STEM activity, got hooked**
- **Created my first Microtronic hardware extensions:**
  - Speech Synthesizer using the Emic 2:

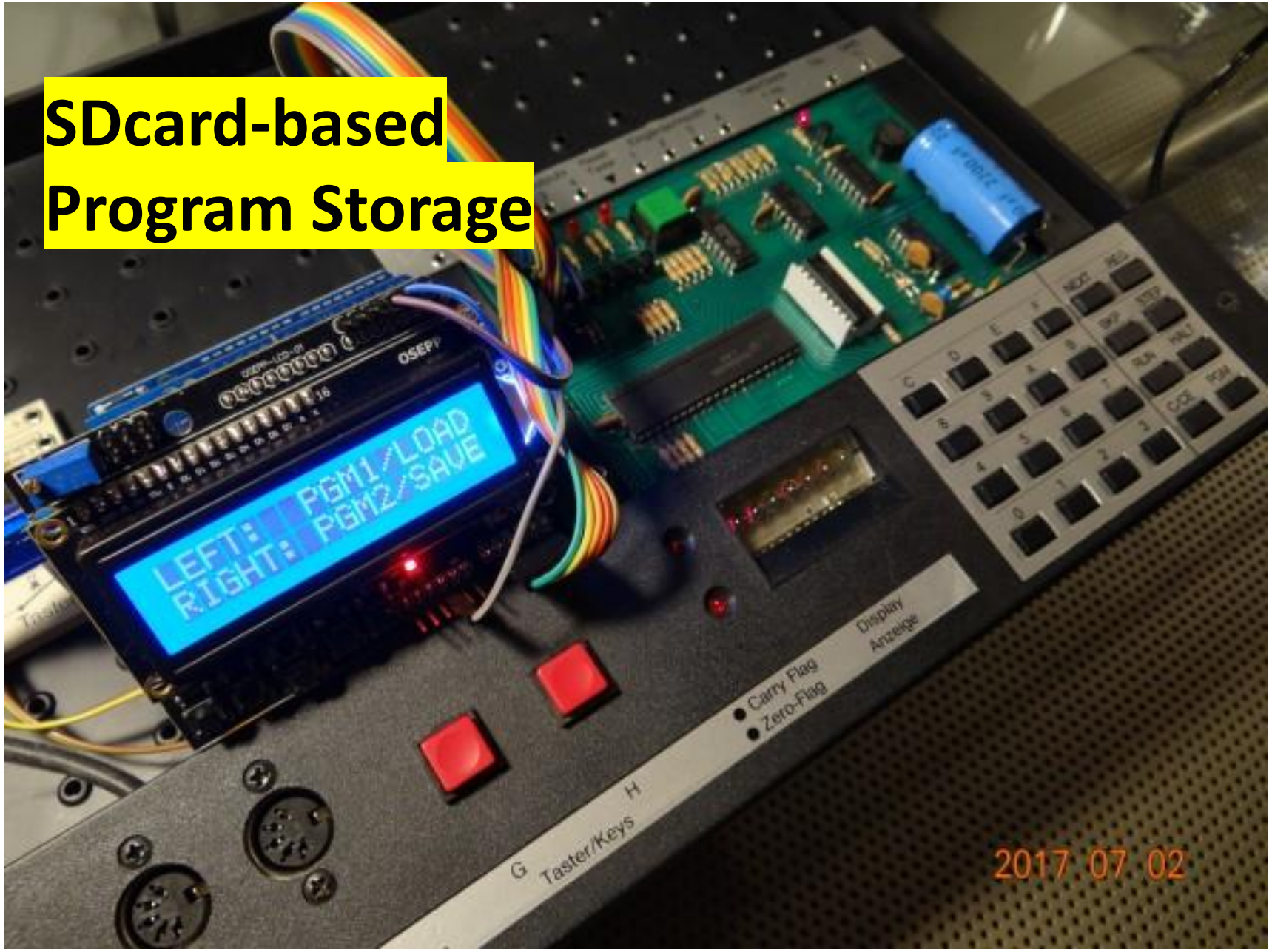
    A speech synthesizer was announced for the Microtronic in 1981.
    But it was never released ☹

    With Arduinos I could built my own!
  - SDcard-based “mass storage” for programs instead of a tape recorder:

    Possible to use a text editor on your PC to write and transfer programs.

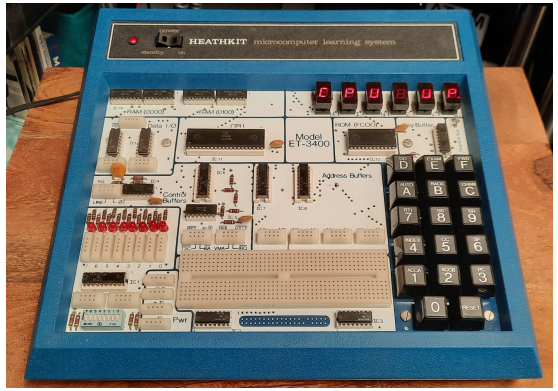


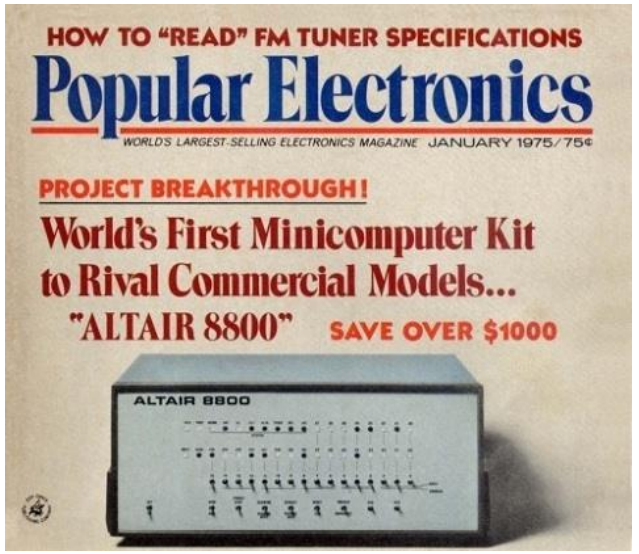
Emic 2 & Arduino Uno R3-based Speech Synthesizer


SDcard-based
Program Storage
LEFT: PGM1/LOAD
RIGHT: PGM2/SAVE
Carry Flag
Zero-Flag
Display
Anzeige
G
Taster/Keys
H
2017 07 02

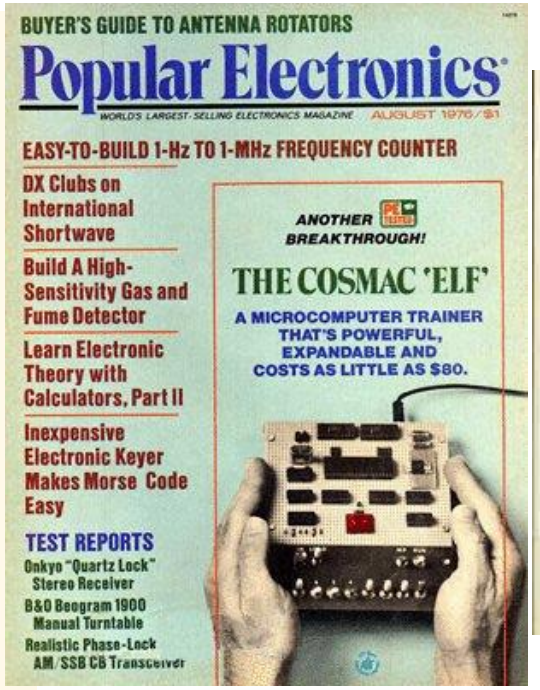
Early homebrew computers

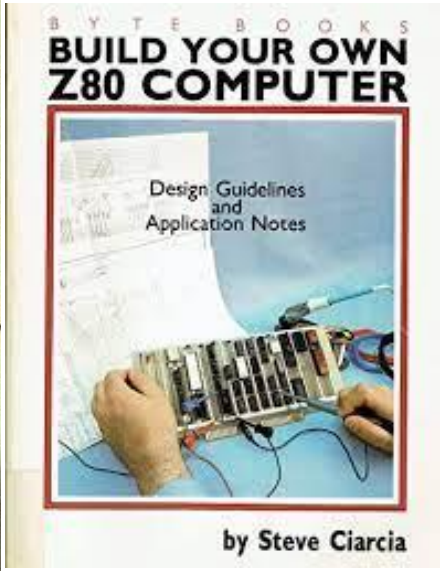
- **Common hobby in the late 70s/early 80s**
  - **And still is a popular hobby nowadays (Hackaday)**
  - It was never very difficult to build your own computer
  - BUT it was STILL expensive and not everybody had the tools and skills
    - Some CPUs were only 10 $ (6502)
    - RAM was very expensive
    - There were some more affordable kits
- **SLAC Homebrew Computer Club**
  - Apple 1, Apple][
  - TRS-80 Model 1
- **Home Computer Revolution “Trinity” of 1977**
  - Apple II
  - TRS-80 Model 1
  - Commodore PET











I built my own computer – but what does that mean?

- **Nowadays, hobbyists have many more options, and things got MUCH more affordable**
- **Question: on what level did you do that?**
  - "I built my own PC from off-the-shelf parts vs. I built my own computer with a custom CPU"
  - Hmm…
- **Retro Computers**
  - **Classic Standard CPU**
    - Z80, 6502, … like back in the day – but use modern components as well (e.g., for video, RAM, storage, …)
    - Building your own retro SBC running CP/M, BASIC, … is a very popular hobby!
    - Contemporary versions of these CPUs are still made
  - **Custom CPU**
    - Maybe even from discrete TTL chips or transistors (!)
  - **Software / Hardware Emulators**
    - Pure software that runs on a PC / Mac (easy, but to achieve high fidelity might be tricky)
    - Embedded software – runs on a custom piece of hardware using a Microcontroller or FPGA
    - High-fidelity hardware emulation on the chip level (FPGAs, CPLDs, …) is trickier (i.e., SRI's Microcircuit Emulation Center)

The Hows

How to implement an educational computer in 1981?

- **Design objectives**
  - Cost
  - Keep it simple - “kid friendly” yet powerful machine language
- **Answer: Microcontrollers – back in the day and today**
  - MUCH reduced circuit complexity and cost
  - No extra RAM, ROM, IO circuitry required
- **The Busch 2090 Microtronic**
  - TMS1600 @ 500 kHz (bigger version in the TMS1000 family)
  - **Firmware: Emulated Virtual CPU & Monitor Program**
  - Harvard architecture
    - 256 x 12bit words in program RAM
    - two sets of 16 4bit registers are the only program writable memory
    - only immediate and register direct addressing
  - **Concise and powerful custom Machine Language (ML)**
    - MULT, DIV, CLOCK, RND, INPUT & OUTPUT -> concise programs
    - Single invocation subroutines (no Stack!)
    - ML knowledge not transferable (unlike Z80, …)
    - But very good exposure to the basics of ML

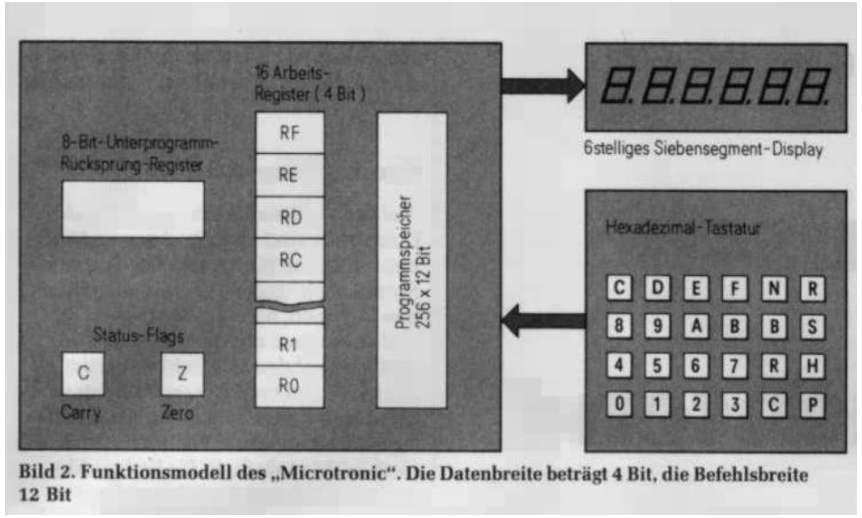
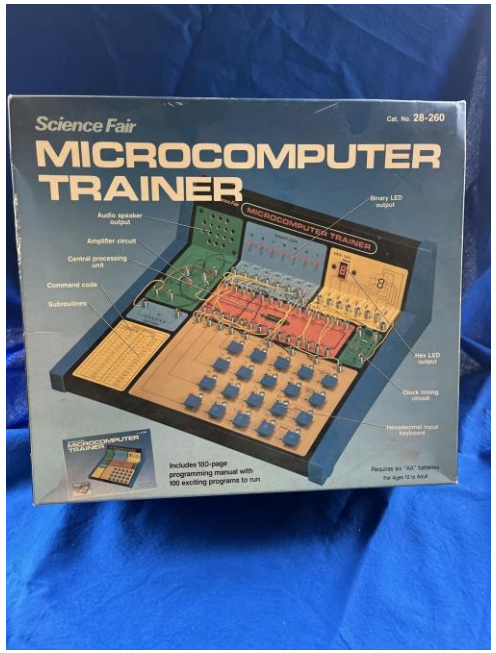


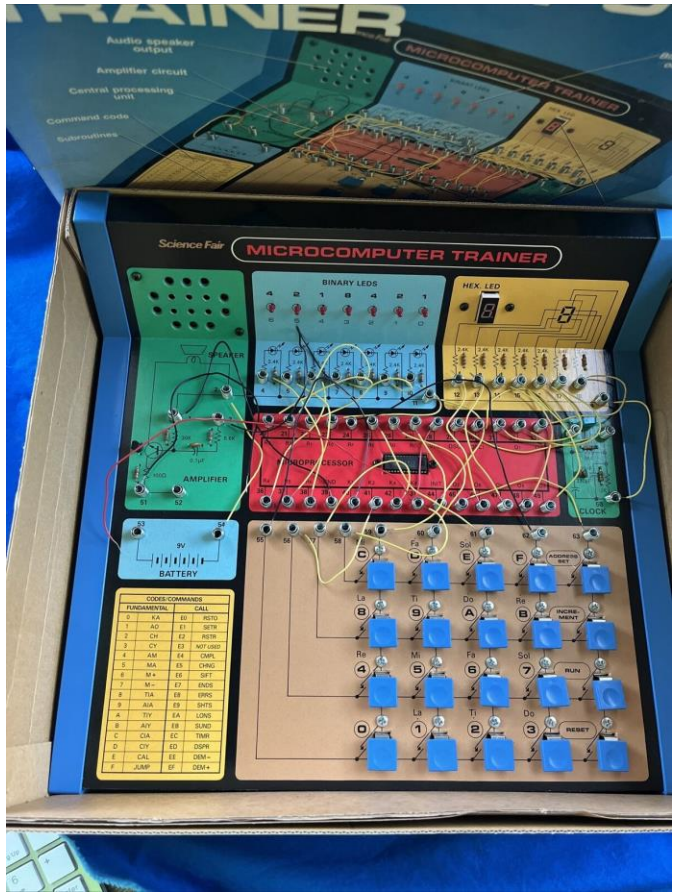
Bild 2. Funktionsmodell des „Microtronic“. Die Datenbreite beträgt 4 Bit, die Befehlsbreite 12 Bit

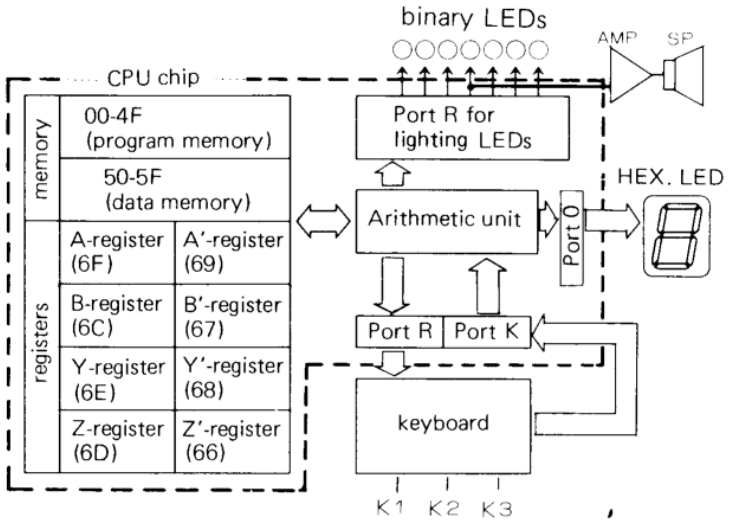


How to implement an educational computer in 1976?

- **A related design from the U.S.**
  - TMS-1100 @ 400 kHz
  - Tandy Radio Shack
  - **1976** (!) – 2 years after the TMS1000 became available
  - A single chip design
- **Emulated Virtual CPU & Monitor**
  - **Completely different from the Microtronic**
  - 80 addresses for program
  - 16 addresses for data (!)
  - 4 to 12bit words
  - 4bit registers:
    A, A', X, Y, X', Y'
  - No sub-routines
- **Both the Microtronic and the Science Fair have NOTHING in common with the native TMS instruction set**







The Legendary TMS1000 Microcontroller

- **TI invented the Microcontroller – patent filed 12/74**
  1977 - Gary W. Boone, Michael J. Cochran
  https://patents.google.com/patent/US4074351
- **Largely forgotten nowadays… hugely enabling back then**
- https://spectrum.ieee.org/texas-instruments-calculator
- https://hackaday.com/2020/02/18/the-tms1000-the-first-commercially-available-microcontroller/







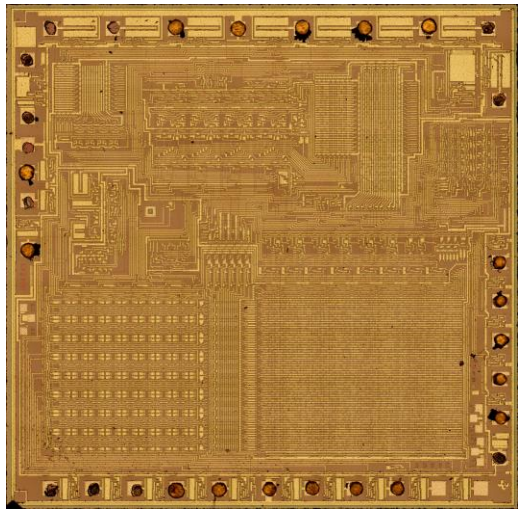


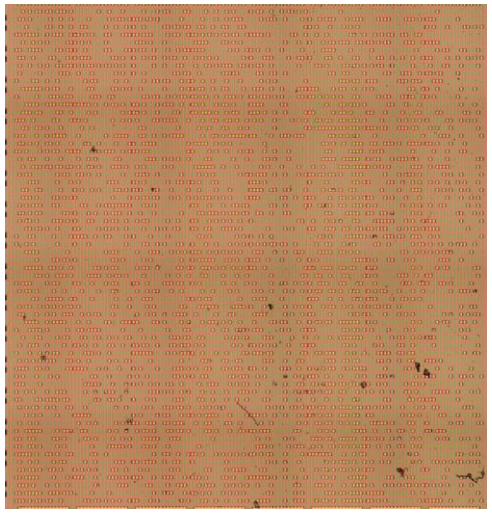


Original firmware? Hmm…



- **The TMS1000 uses “mask-programmed” ROMs**
- **These cannot be read out, and the masks are lost to history (I asked TI….)**
  - https://seanriddle.com/simon.html
  - Decap the chip and decode the Mask ROM matrix -> destructive
  - Firmware of many devices on his site (Merlin, Speak & Spell, Little Professor)



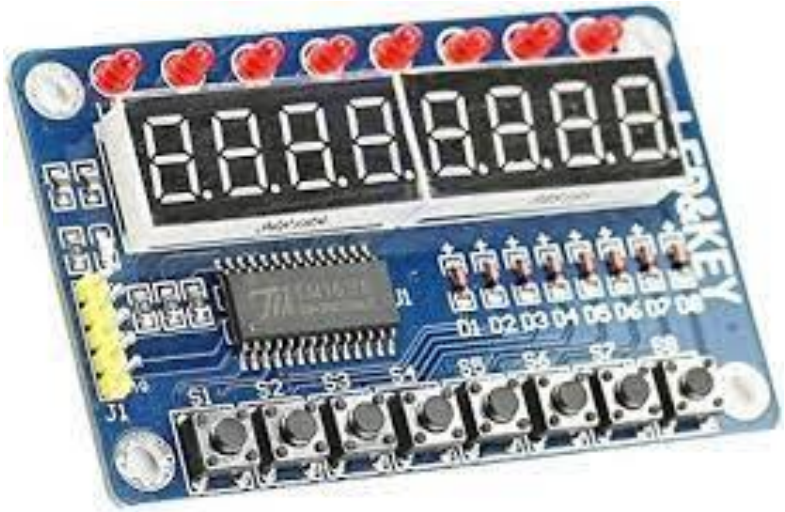


1000110101
1101000011
1001001111
1110010100

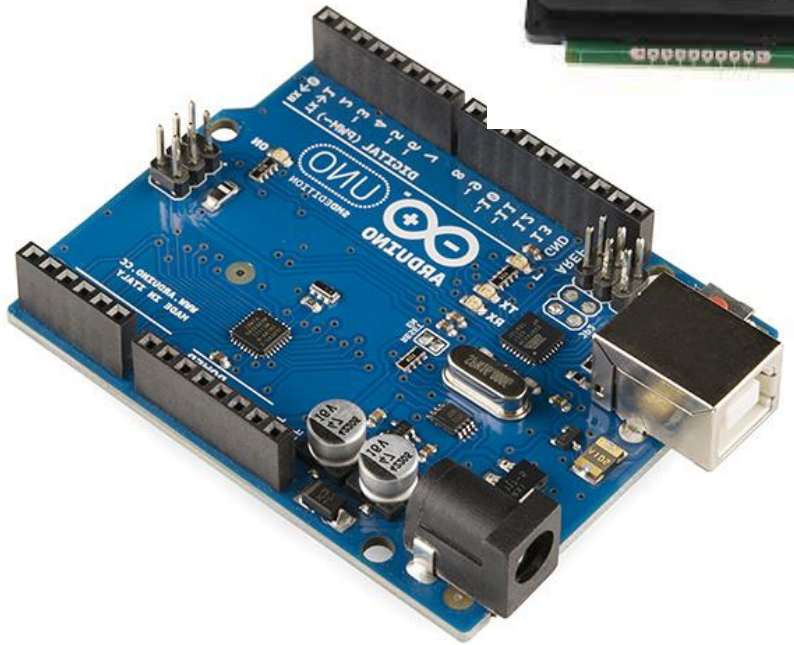
TMS1000
Emulator

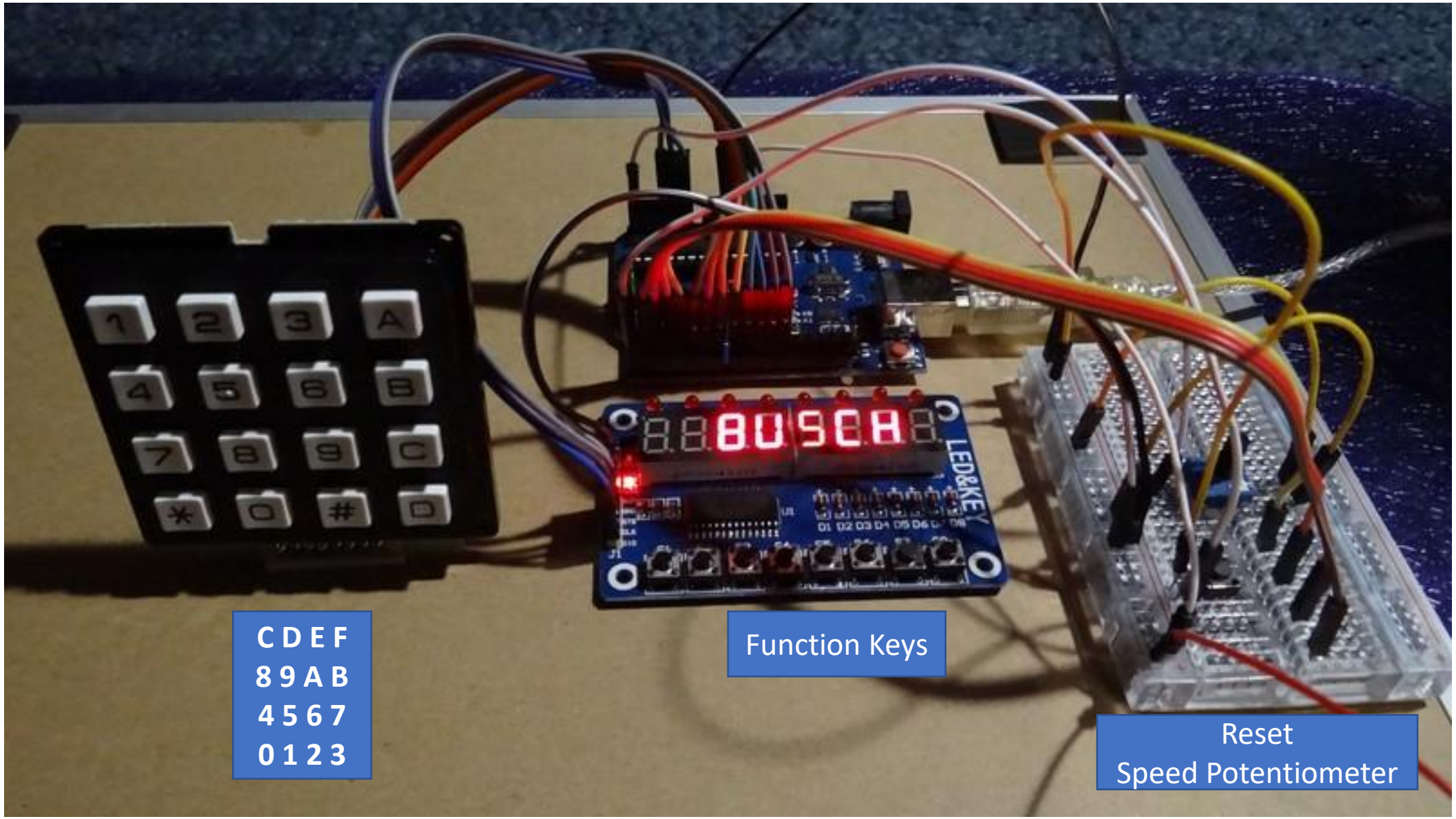
Iteration 1 - 2016

- **Build a Microtronic emulator from modules**
  - CPU emulator
  - Monitor program
  - Hardware drivers for modules
- **Original firmware? No…**
  - The TMS1600 ROM cannot be read out!
  - Firmware is not available at all
  - Reengineer from scratch (Arduino C program)
- **Good enough to run original Microtronic programs**
  - Writing a software-based CPU emulator is not difficult
  - Doesn’t have to be timing accurate
  - Don’t need to emulate the chip and its signals
- **Off-the-shelf modules and DuPont wires**
  - Arduino R3
  - “LED&Key” TM1638 module
  - Matrix-encoded keypad
  - Breadboard for some extra buttons and speed potentiometer








C D E F
8 9 A B
4 5 6 7
0 1 2 3
Function Keys
Reset
Speed Potentiometer

CPU emulation is simple

```
223   boolean clock = false;
224   boolean carry = false;
225   boolean zero = false;
226   boolean error = false;
```

CPU Flags (Carry, Zero)

```
250   //
251   // RAM program memory
252   //
```

RAM: simple byte arrays

```
254   byte op[256];
255   byte arg1[256];
256   byte arg2[256];
```

```
258   boolean jump = false;
259   byte pc = 0;
260   byte breakAt = 0; // != 0 -> breakpoint set
```

Program Counter (PC)

```
272   byte stack[STACK_DEPTH];
273   byte sp = 0;
```

Stack
Stack Pointer (SP)

```
279   byte reg[16];
280   byte regEx[16];
```

Registers
Extra Registers

Firmware LOC:
20% CPU
80% Monitor
~ 2000 LOC

```
1505    byte op1 = op[pc];
1506    byte hi = arg1[pc];
1507    byte lo = arg2[pc];
1508
1509    byte s = hi;
1510    byte d = lo;
1511    byte n = hi;
```

Fetch from RAM using program counter

```
1522    switch (op1)
1523    {
1524    case OP_MOV:
1525
1526      reg[d] = reg[s];
1527      zero = reg[d] == 0;
1528
1529      break;
1530
1531    case OP_MOVI:
1532
1533      reg[d] = n;
1534      zero = reg[d] == 0;
1535
1536      break;
1537
1538    case OP_AND:
```

"Decode" Instruction

Implement Instruction

`MOV s -> d`

`0<source><dest>`

`012 = MOV R1 -> R2`

`MOVI n -> d`

`1<const><dest>`

`112 = MOVI 1 -> R2`

Some clever instructions leading to concise programs.

CALL/RET

Digital Input & Output

Dec -> Hex
Hex -> Dec
Random
Clock

Multiplication
Division

Der Microtronic-Befehlssatz (siehe auch Anleitungsbuch Seite 74 bis 76)

Befehls-Kürzel	Befehls-Code	Erklärungen		Ausführlich mit Beispiel erklärt auf Seite
MOV	0 s d	**s** nach **d**	Inhalt von Reg. **s** in Reg. **d** bringen	22
MOVI	1 n d	**n** nach **d**	Konstanten Wert **n** in Reg. **d** bringen	33
AND	2 s d	**s** und **d** = **d**	Logische „Und"-Operation	59-60
ANDI	3 n d	**n** und **d** = **d**	Logische „Und"-Operation mit einer Konstanten	59-60
ADD	4 s d	**s** ⊕ **d** = **d**	Addition Reg. Inhalte **s** ⊕ **d** = Ergebnis in **d**	18-21
ADDI	5 n d	**n** ⊕ **d** = **d**	Addition konst. Wert **n** ⊕ Reg. **d** = Ergebnis in **d**	19-21
SUB	6 s d	**d** ⊖ **s** = **d**	Subtraktion Reg. Inhalte **d** ⊖ **s** = Ergebnis in **d**	35
SUBI	7 n d	**d** ⊖ **n** = **d**	Subtraktion Reg. **d** ⊖ Konst. Wert **n** = Ergebnis in **d**	35
CMP	8 s d	Vergleiche **s, d**	Registerinhalt **s** mit Reg. Inhalt **d** vergleichen	74
CMPI	9 n d	Vergleiche **n, d**	Konst. Wert **n** mit Registerinhalt **d** vergleichen	26
OR	A s d	**s** oder **d** = **d**	Logische „Oder"-Operation	59-60
CALL	B a a	Sprung **a a**	Sprung zu Unterprogramm beginnend bei Adresse **a a**	63-64
GOTO	C a a	Sprung **a a**	Sprung zu einer beliebigen Adresse **a a**	14-15
BRC	D a a	Carry-Sprung **a a**	Sprung zur Adresse **a a**, wenn Carry-Flag gesetzt	26
BRZ	E a a	Zero-Sprung **a a**	Sprung zur Adresse **a a**, wenn Zero-Flag gesetzt	33
MAS	F 7 d	A.-Reg. → S.-Reg.	Arbeitsregister **d** in Speicherregister **d** bringen	61
INV	F 8 d	Invertiere **d**	Inhalt von Reg. **d** soll invertiert werden	50
SHR	F 9 d	Schiebe **d** rechts	Registerinhalt **d** dual nach rechts schieben	49
SHL	F A d	Schiebe **d** links	Registerinhalt **d** dual nach links schieben	48
ADC	F B d	**d** ⊕ C = **d**	Gesetztes Carry-Flag addiert „1" in Register **d**	19-21
SUBC	F C d	**d** ⊖ C = **d**	Gesetztes Carry-Flag subtrahiert „1" in Reg. **d**	35
DIN	F D d	Daten Ein	Daten von Eingänge in Register **d** speichern	37
DOT	F E s	Daten Aus	Daten von Register **s** an Ausgänge bringen	36
KIN	F F d	Tasten-Wert	Warten auf Eingabe. Eingabewert in Reg. **d** speichern	14-15
DISP	F n s	Display Ein	Anzeigen: **n** = wieviele (Reg.)Stellen. **s** = ab Reg. Nr.	14-15
HALT	F 0 0	Programm-Stop	Gestopptes Programm mit Taste STEP dann RUN fortsetzen	63
NOP	F 0 1	Keine Operation	Es kann nachträglich ein Befehlscode eingegeben werden	63
DISOUT	F 0 2	Display Aus	Anzeige abschalten (für schnellere Operationen)	14-15
HXDZ	F 0 3	HEX in DEZ	Umwandlung hexadezimal in dezimal in Reg. D–E–F	43
DZHX	F 0 4	DEZ in HEX	Umwandlung dezimal in hexadezimal in Reg. D–E–F	43
RND	F 0 5	Zufallsgenerator	Zufallszahlen in Reg. D–E–F ermitteln	41
TIME	F 0 6	Uhrzeit	Uhrzeit in Register A–B–C–D–E–F übernehmen	44-45
RET	F 0 7	Rück-Sprung	Aus einem Unterprogramm ins Hauptprogramm zurückspringen	63-64
CLEAR	F 0 8	Register löschen	Alle Registerwerte auf „0" setzen	21
STC	F 0 9	Carry setzen	Carry-Flag soll ausgelöst (gesetzt) werden (Wert: 1)	52
RSC	F 0 A	Carry rücksetzen	Ein gesetztes Carry-Flag soll zurückgesetzt werden (0)	52
MULT	F 0 B	Multiplikation	In den Registern 0 bis 5 wird Multiplikation ausgeführt	61-62
DIV	F 0 C	Division	In Reg. 0 bis 3 wird Division ausgeführt. (Reg. 4 und 5 Wert „0")	62-63
EXRL	F 0 D	Tausche A/S 0–7	Inhalte Arbeitsregister 0 bis 7 mit Speicherregister 0 bis 7 tauschen	61
EXRM	F 0 E	Tausche A/S 8–F	Inhalte Arbeitsregister 8 bis F mit Speicherregister 8 bis F tauschen	61
EXRA	F 0 F	Tausche 0–7/8–F	Inhalte Arbeitsregister 0 bis 7 mit Arbeitsregister 8 bis F tauschen	33

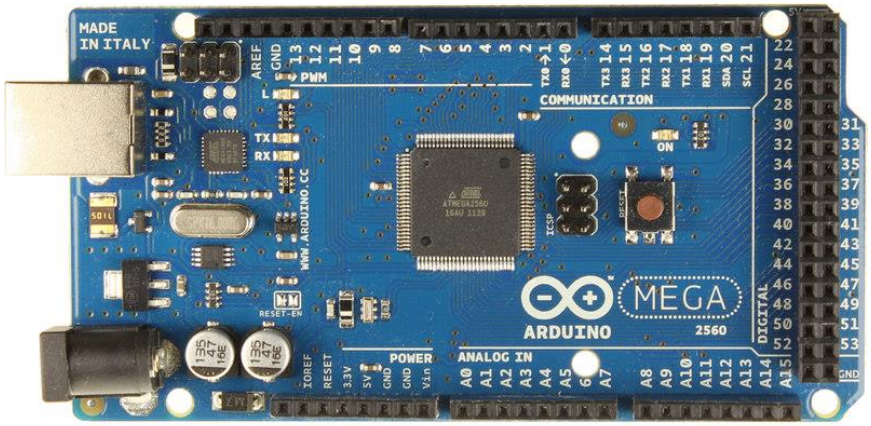
Z80 ML would have been MUCH less concise – Limited RAM…

Iteration 2 – 2016

- **LCD Display & SDCard for storage would be nice**
- **LCD + Keys Shield module -** plugs onto the Mega
  - Mnemonics display
  - All register at once display

  … but keep the “LED display & hex user experience”
- **SDCard & Ethernet Shield module**
  - Save & Load
  - Exchange / edit programs on your PC
  - I used OCR on the listings -> ~40 programs
- **Arduino Uno R3 no longer sufficient**
  - FAT library
  - Many strings for the LCD-based “user interface”
  - **Software from ~ 2000 LOC to ~ 4000 LOC**
  - Hint: Use the “f” macro!!

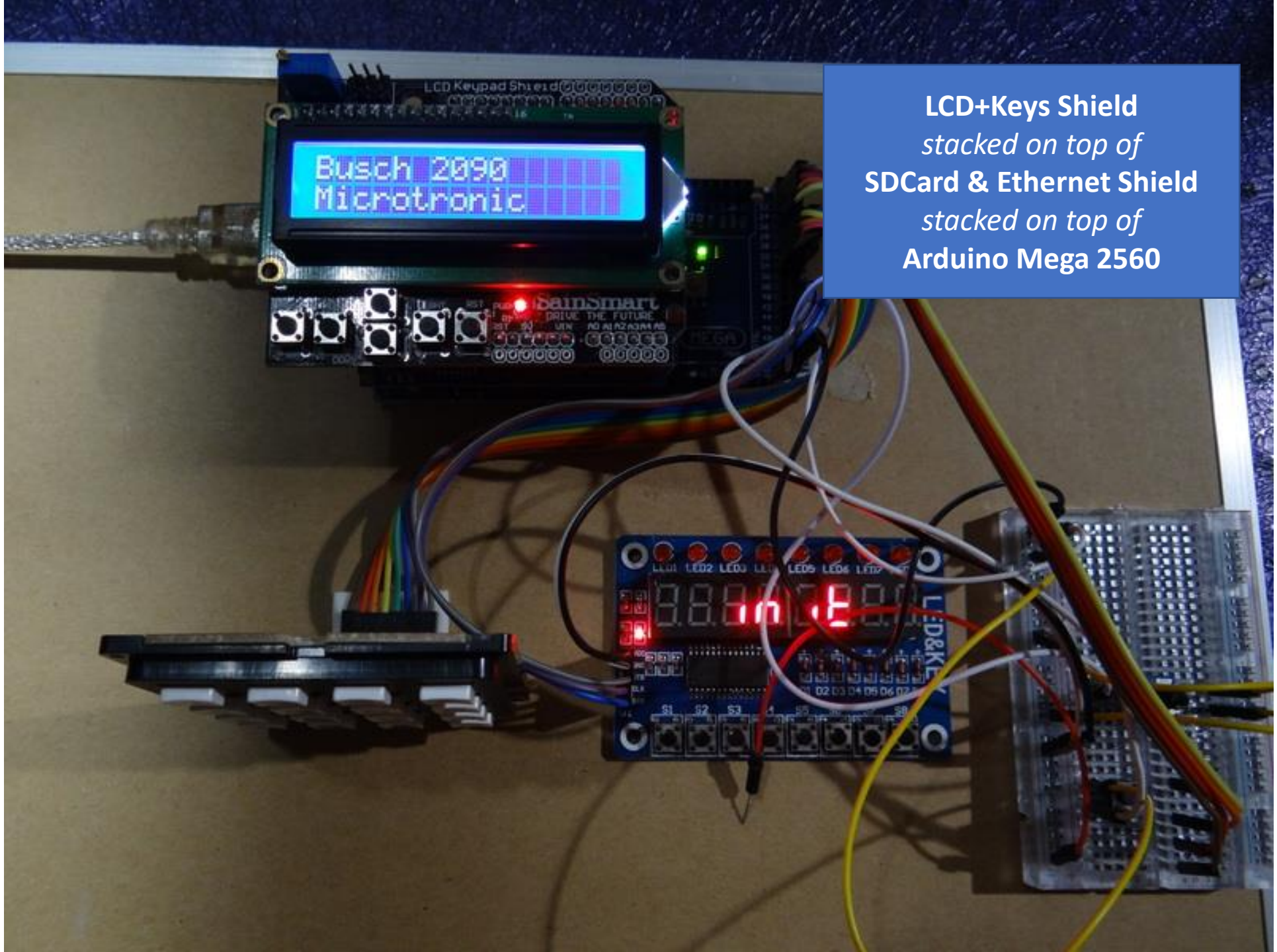
  -> switched to Arduino Mega 2560



“Arduino Mega 2560 R3 Front” from arduino.cc
Reproduced under CC BY-SA 3.0 License







LCD+Keys Shield
stacked on top of
SDCard & Ethernet Shield
stacked on top of
Arduino Mega 2560

Iteration 3 – 2016



- **Something more presentable and durable**
  - **Custom laser-cut front plate**
  - Retired the LED&KEY & LCD+Keys shields; I2C LED displays & LCD module instead
  - One more telephone keypad for function keys



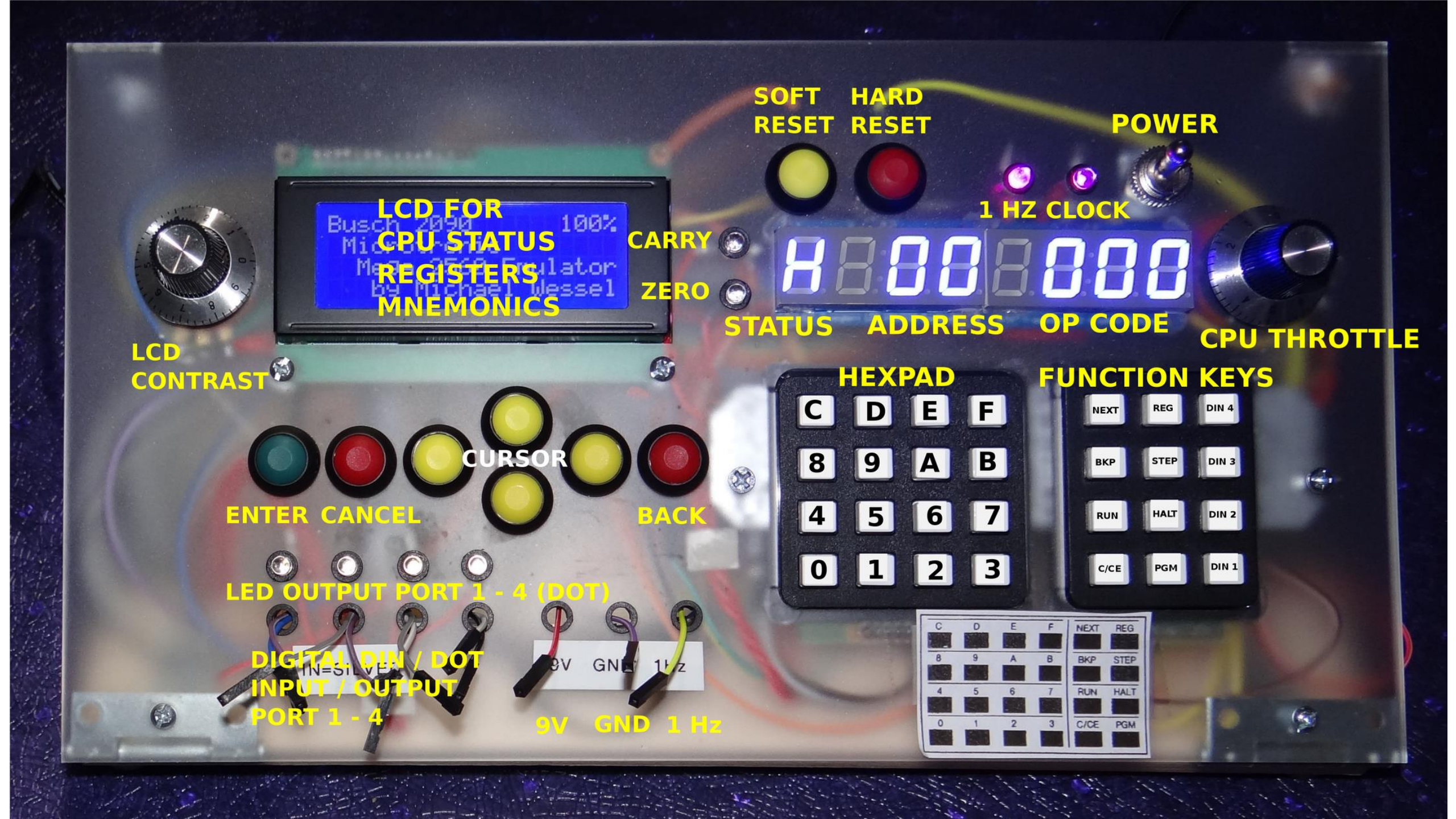
- **Some fixes**
  - OUTPUT ports had worked before
  - But digital INPUT ports (for electronics experiments) were problematic
    - **Unconnected** Microtronic input pin (and **GND**) **-> LOW**, **5V -> HIGH**
    - **Unconnected** Arduino input pin -> **FLOATING** (bad)
    - Arduino's **internal pull-up resistors invert** logic levels, hence **unconnected -> 5V = HIGH** and **GND -> LOW**
    - C-code negation such that **unconnected -> LOW** doesn't help: then **5V = HIGH from external electronics -> LOW** (experiments won't work)
    - Hence**, external pull-down resistors** were needed

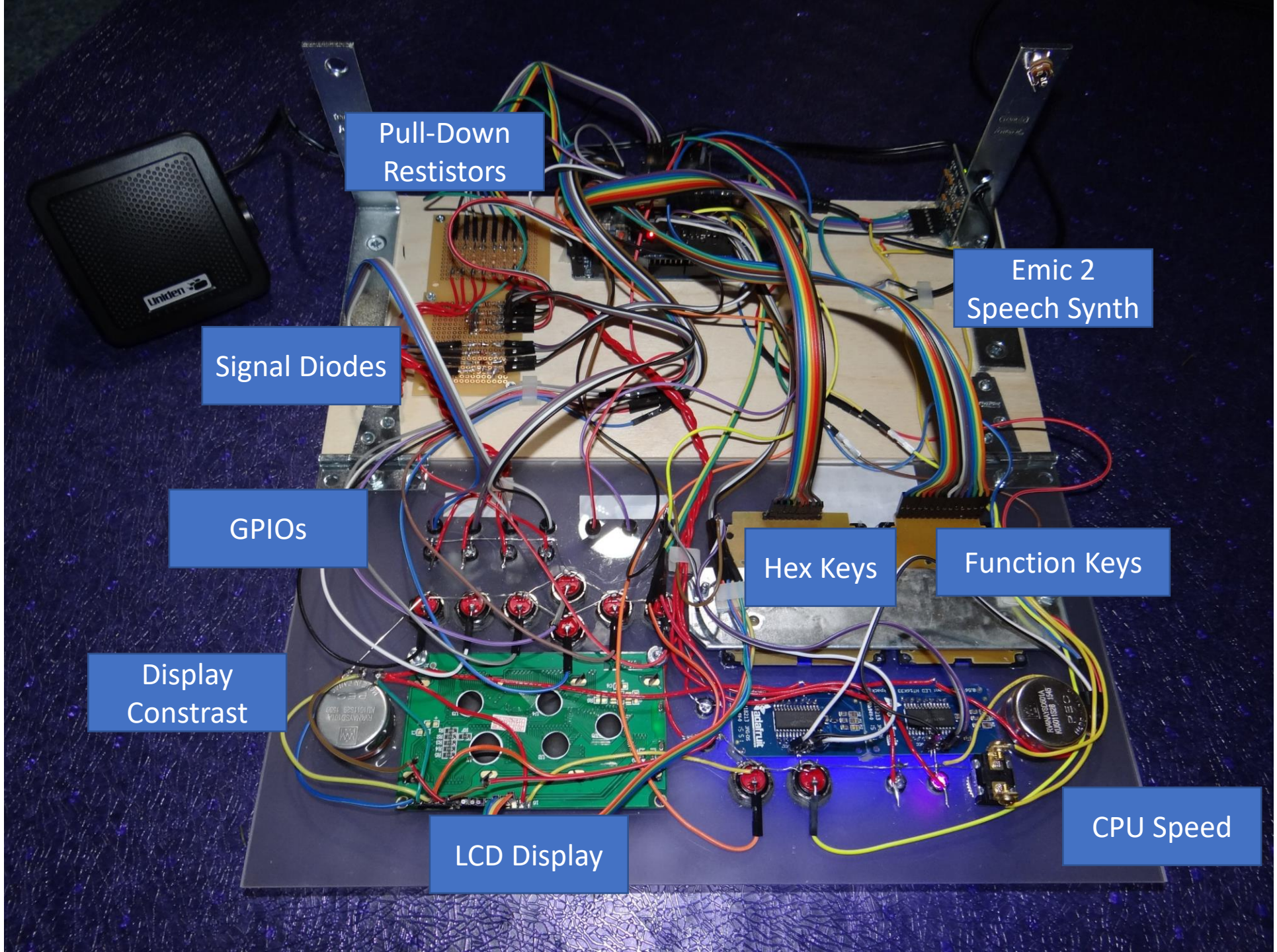


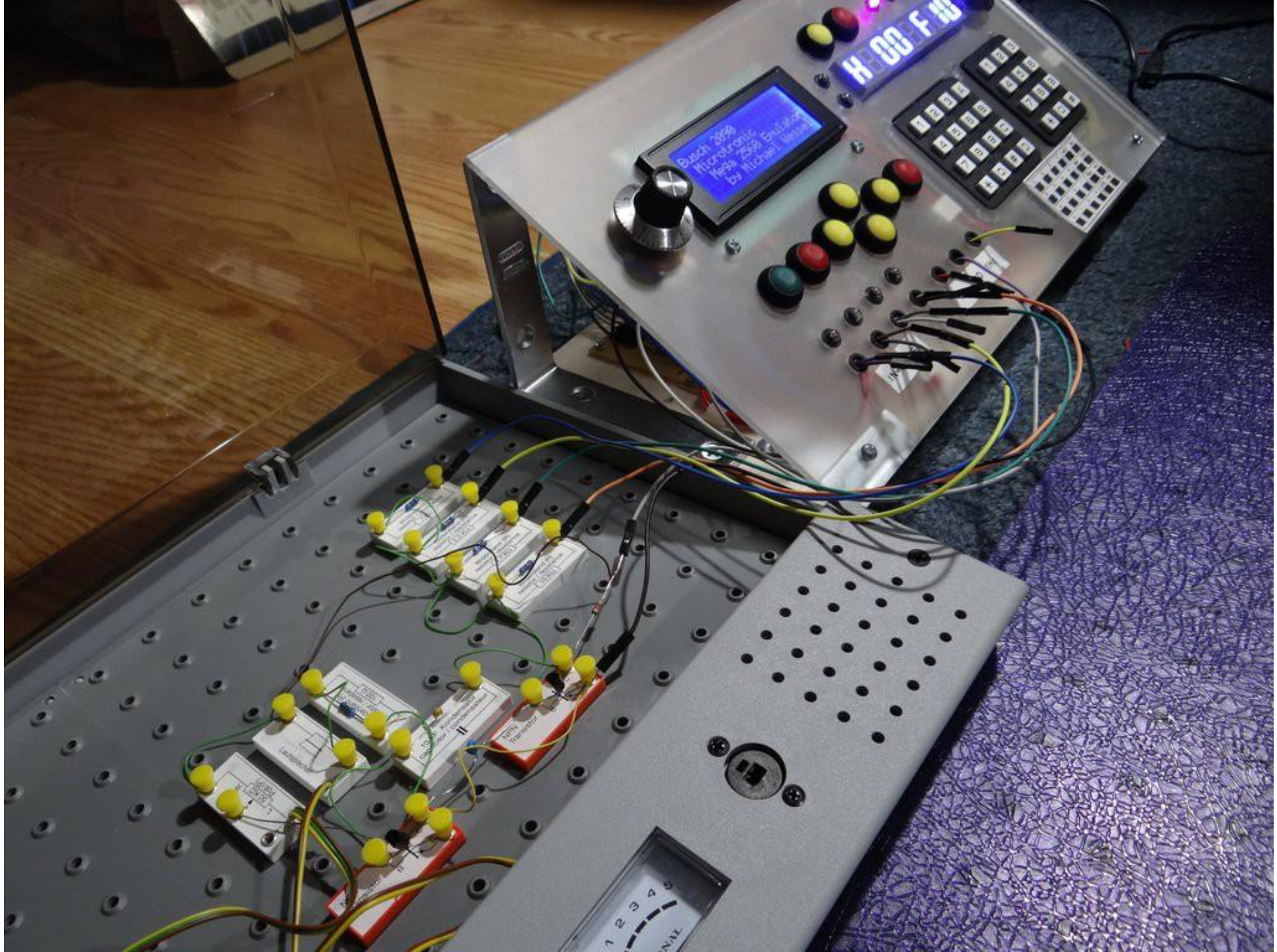
- **Added the Emic 2 Speech Synthesizer**
  - How to drive / program it? No unused op-codes in the Microtronic!
  - But there are semantically "meaningless" op-codes, e.g., **MOV x -> x**
  - **These "vacuous" op-codes are not used in real programs**

  -> Extended the CPU emulator with **extra-side effects for these** (e.g., for speech synth)


PC
0: 00: OP MNEM 100%
WR 929: 5GG ADDI
SR 0000000000000000
6d4b28
1 2 3 A
4 5 6 B
7 8 9 C
* 0 # D
1 2 3
4 5 6
7 8 9
* 0 #

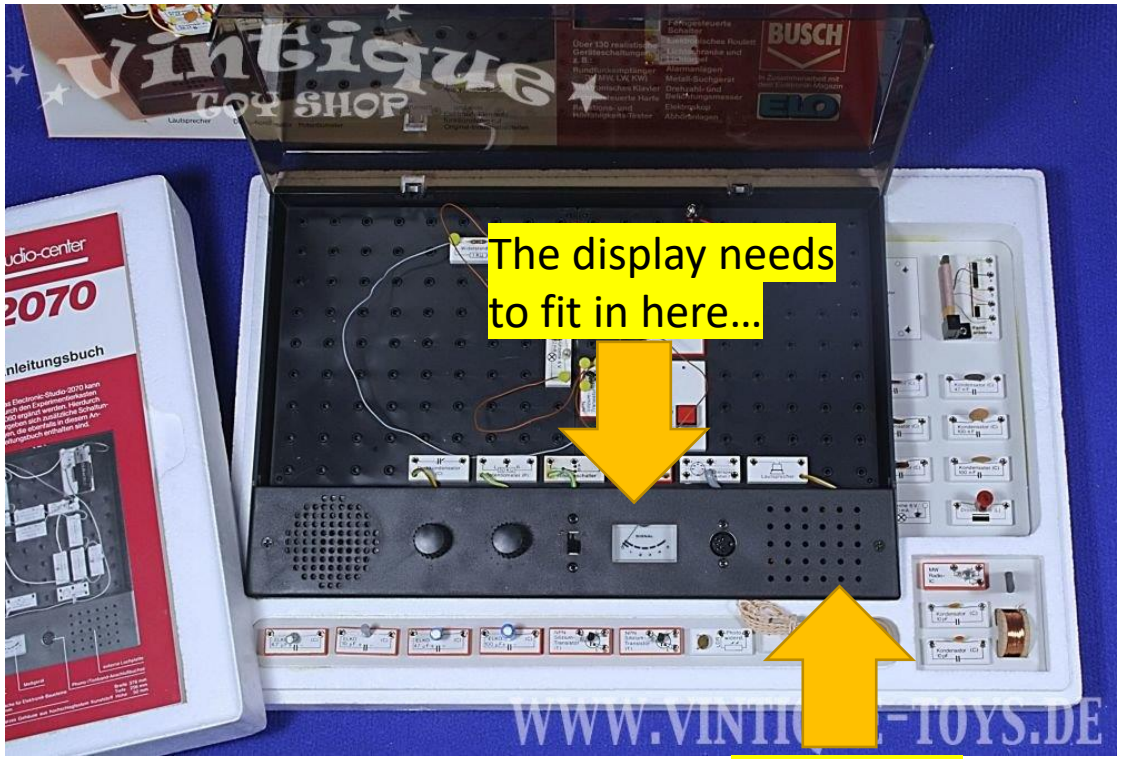

SOFT RESET
HARD RESET
POWER
1 HZ CLOCK
LCD FOR CPU STATUS REGISTERS MNEMONICS
CARRY
ZERO
STATUS
ADDRESS
OP CODE
CPU THROTTLE
LCD CONTRAST
HEXPAD
FUNCTION KEYS
CURSOR
ENTER
CANCEL
BACK
LED OUTPUT PORT 1 - 4 (DOT)
DIGITAL DIN / DOT INPUT / OUTPUT PORT 1 - 4
9V
GND
1 Hz


Pull-Down Restistors
Emic 2 Speech Synth
Signal Diodes
GPIOs
Hex Keys
Function Keys
Display Constrast
CPU Speed
LCD Display

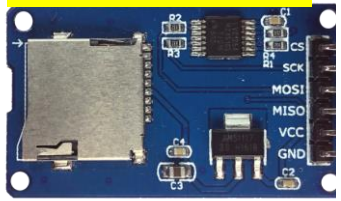

Busch 2668
Microtronic
Mega 2668 Emulator
by Michael Wessel

Fast-forward to 2020 - iterations 4 & 5

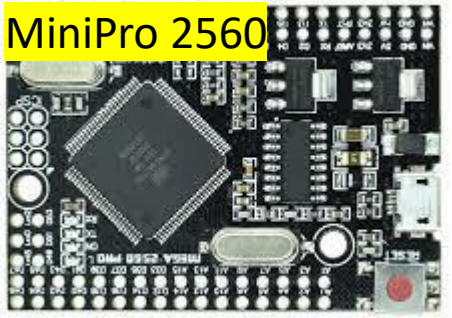
- **My “team” found me ☺**
  - Frank de Jaeger (Belgium) & Manfred Henf (Germany)
  - Via GitHub and Hackaday pages
  - Their proposal: integrate into a Busch console, a more authentic **Microtronic 2nd Generation**
  - They asked me to adjust the and extent the firmware
- **Mostly identical to my Arduino Mega version, but:**
  - Smaller Arduino board **MiniPro 2560**
  - No shields for this form factor -> **SDcard SPI module instead of SDcard shield**
  - **A display that would fit into the console** Nokia 5110 LCD
  - Major rewrite of the firmware
- **Frank designed a PCB for off-the-shelf modules**
- **3D printed parts from Manfred**
- **Even better GPIO ports**
  - Frank replicated the original port drivers for maximal electrical compatibility



SDCard Module

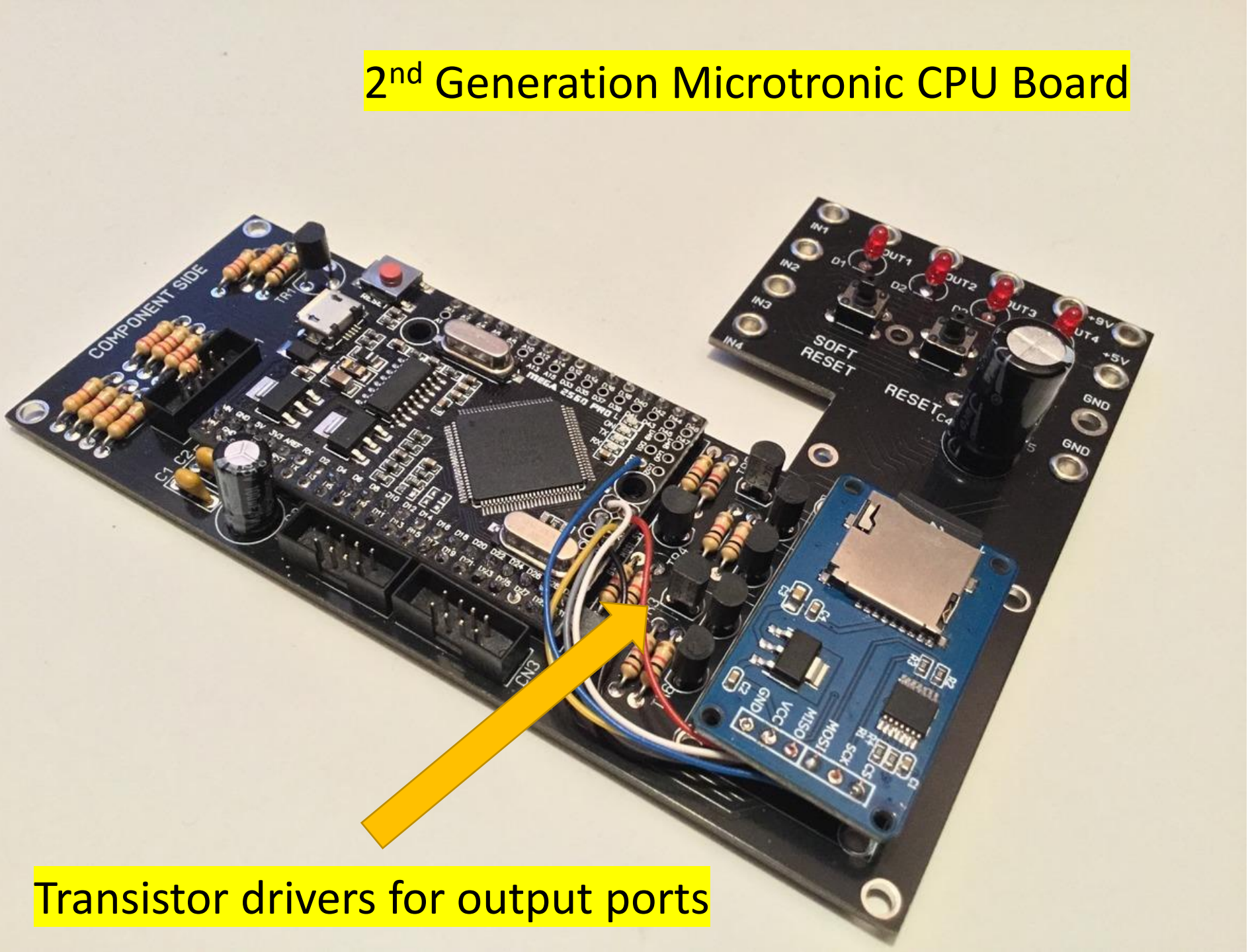


MiniPro 2560

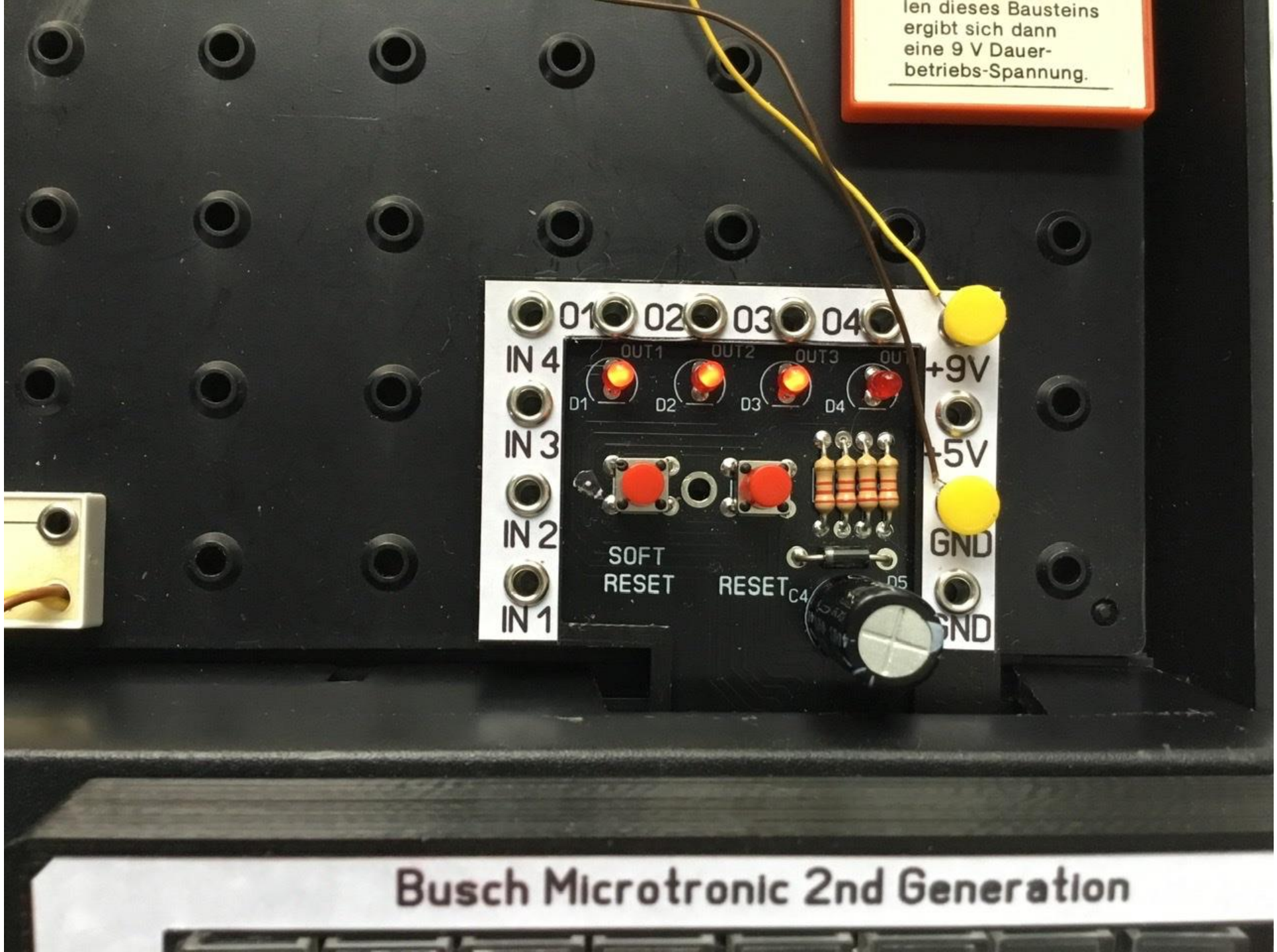


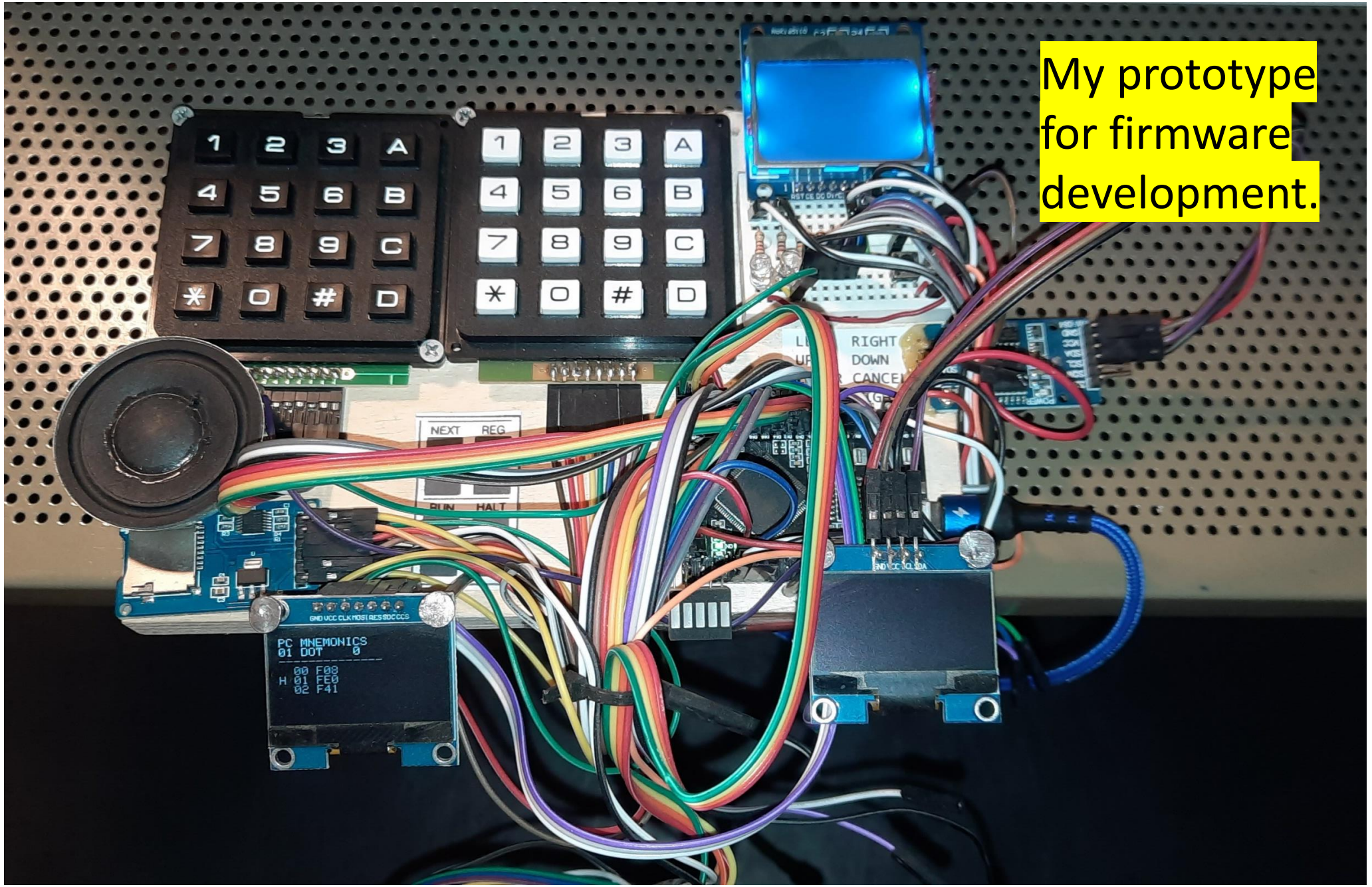

2nd Generation Microtronic in an original Busch console.
With 3D-printed keyboard and Nokia 5110 display.
Designed by
Frank & Manfred.
Firmware by myself.
Busch Microtronic 2nd Generation
C D E F NEXT REG LEFT RGHT
8 9 A B BKP STEP UP DOWN
4 5 6 7 RUN HALT ENTR CNCL
0 1 2 3 C/CE PGM BCK LGHT
CF
ZF
1Hz
HEX Keypad
Function keys

2nd Generation Microtronic CPU Board



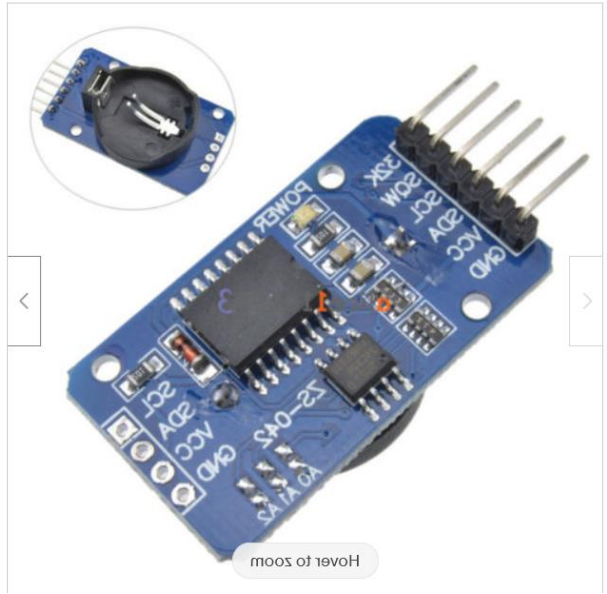
Transistor drivers for output ports

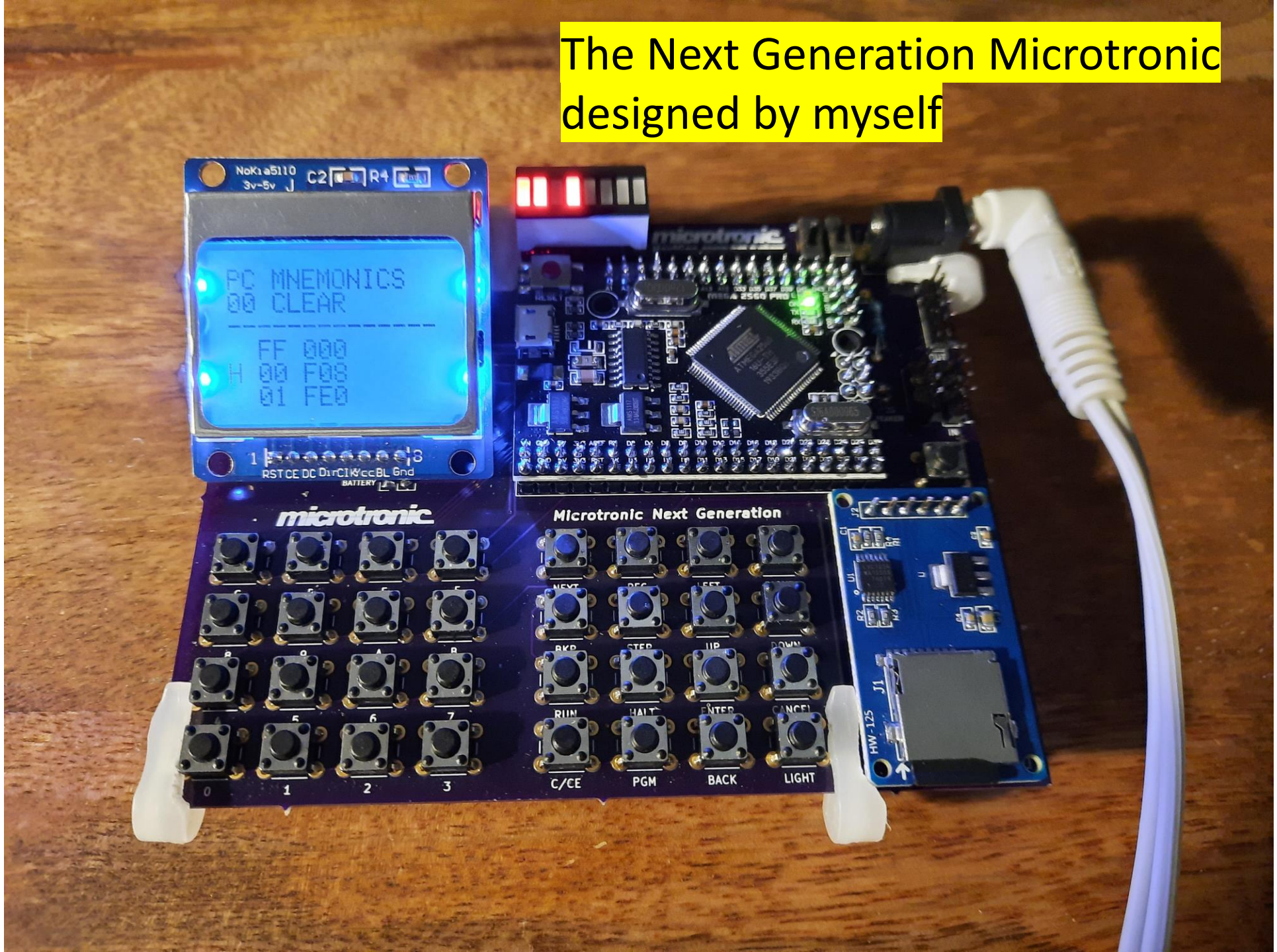

len dieses Bausteins
ergibt sich dann
eine 9 V Dauer-
betriebs-Spannung.
01
02
03
04
IN 4
IN 3
IN 2
IN 1
OUT1
OUT2
OUT3
D1
D2
D3
D4
+9V
+5V
GND
GND
SOFT
RESET
RESET
C4
D5
Busch Microtronic 2nd Generation


My prototype for firmware development.
1 2 3 A
4 5 6 B
7 8 9 C
* 0 # D
NEXT REG
RUN HALT
RIGHT
DOWN
CANCEL
PC MNEMONICS
01 DOT 0
00 F08
H 01 FE0
02 F41

Iterations 6 & 7 – 2021 - A Pocket Microtronic

- The **2nd Gen Microtronic** was a good success
  - ~ 10 or so other Busch enthusiasts build their own
  - Frank and Manfred supplied DIY kits to the community
- **However, I wanted a much simpler, more compact, and portable version**
  - I had also learned how to design my own PCBs with KiCAD: **The Next Generation Microtronic**
  - Same hardware initially, but switched to a **SPI OLED display** later
  - Removed transistor drivers and overkill diode-keyboard matrix
- **I added**
  - A battery backed-up Real Time Clock module (I2C device)
  - Loudspeaker for sound
    - Sound op-codes
    - Again, using “vacuous” op-codes for sound
  - Capability to plug into a real Microtronic for SDcard storage
- Winner of the **Reinvented Retro Contest / 2021 Hackaday**
  - **I accepted the prize for the “team” as it was based on the 2nd Generation Microtronic and shared it with the team**




The Next Generation Microtronic
designed by myself
PC MNEMONICS
00 CLEAR
FF 000
H 00 F08
01 FE0
microtronic
Microtronic Next Generation
C/CE
PGM
BACK
LIGHT

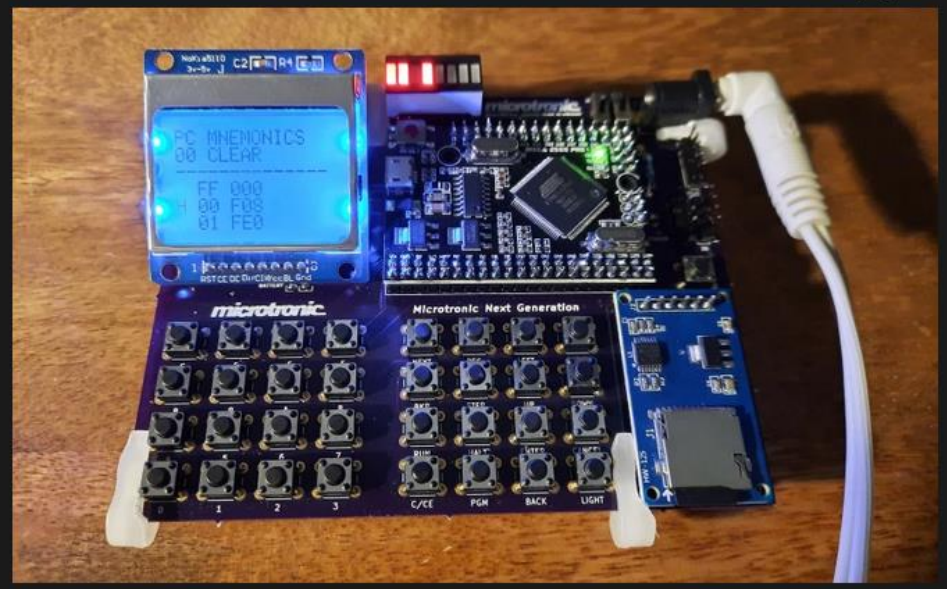
Reinvented Retro Contest Winn
https://hackaday.com/2021/07/17/reinvented-retro-contest-winn
67%
Legen Sie Ihre Lesezeichen hier in der Lesezeichen-Symbolleiste ab, um schnell darauf zuzugreifen. Lesezeichen verwalten...
Weitere Lesezeichen

REINVENTED RETRO CONTEST WINNERS ANNOUNCED

by: Kristina Panos

5 Comments

July 17, 2021



Good news, everyone! The results of the Reinvented Retro contest are in, and the creators of these three groovy projects have each won a $200 online shopping spree to Digi-Key. We asked you to gaze deeply into your stuff piles and come up with a way to modernize a cool, old piece of hardware, and we left it up to you to decide how cool and how old.



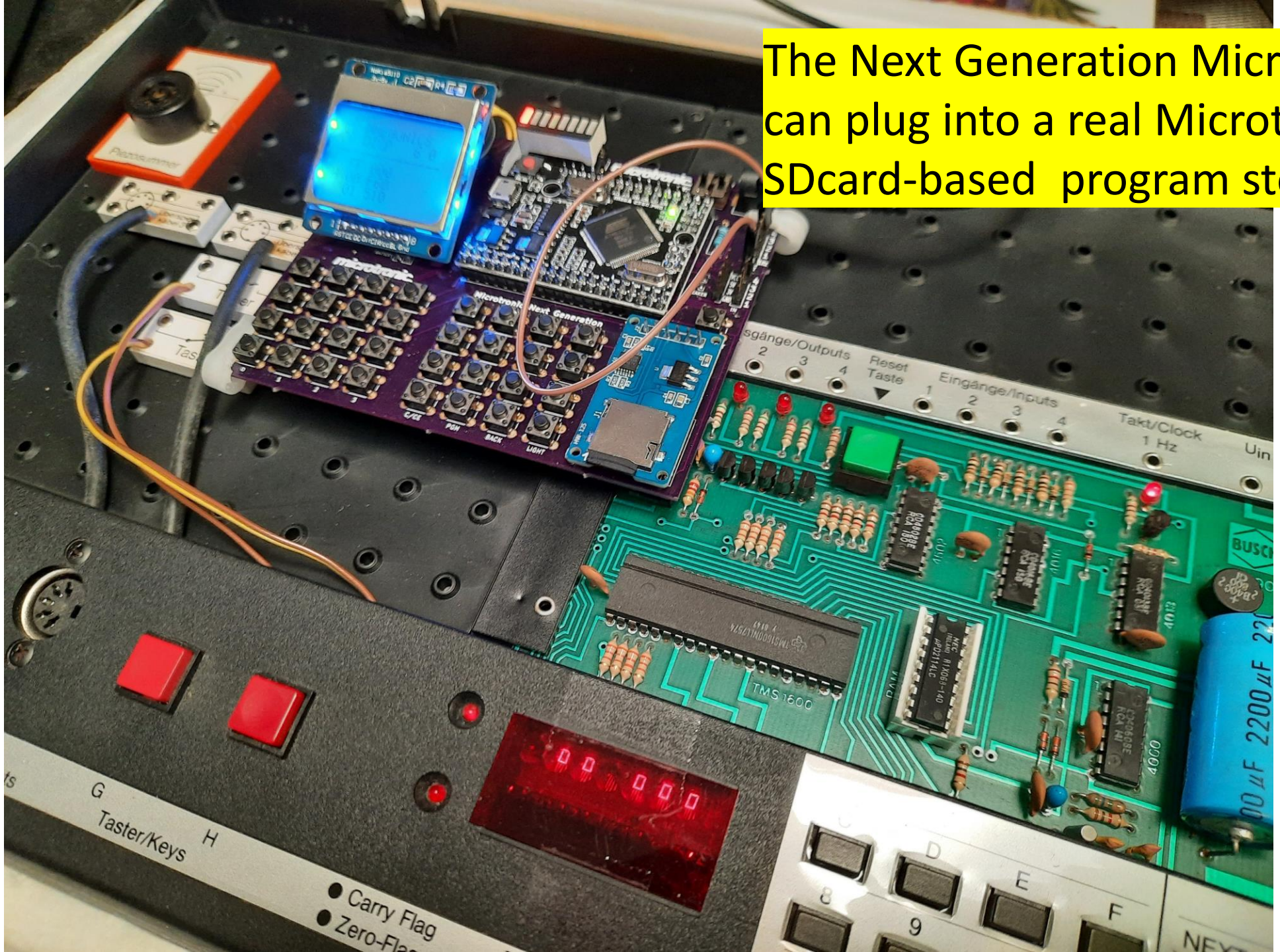


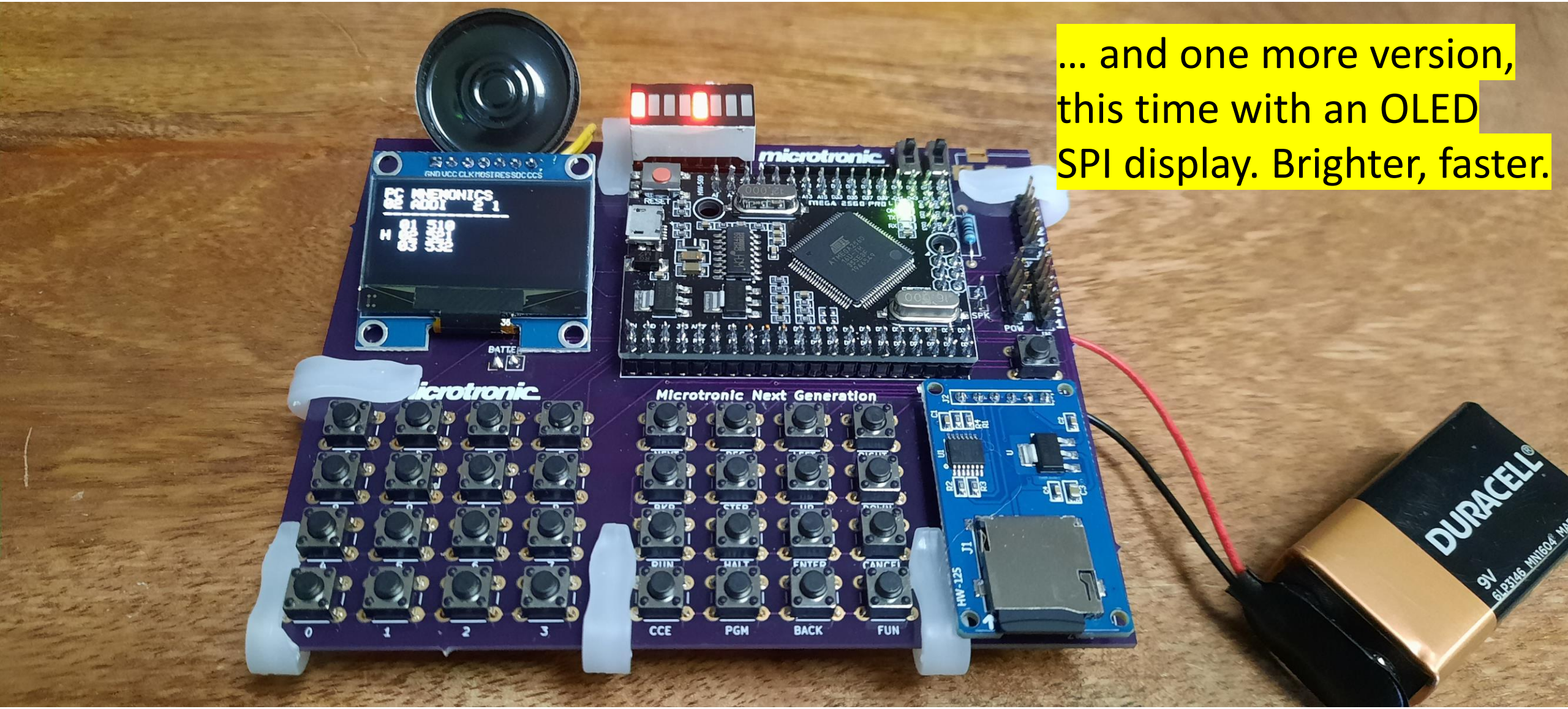
SEARCH

Search ... SEARCH

NEVER MISS A HACK

17°C Meist so...


The Next Generation Microtronic can plug into a real Microtronic for SDcard-based program storage.
Microtronic
Next Generation
Taster/Keys
Carry Flag
Zero-Flag
TMS 1600
Takt/Clock
1 Hz
Reset
Taste
Eingänge/Inputs
Ausgänge/Outputs

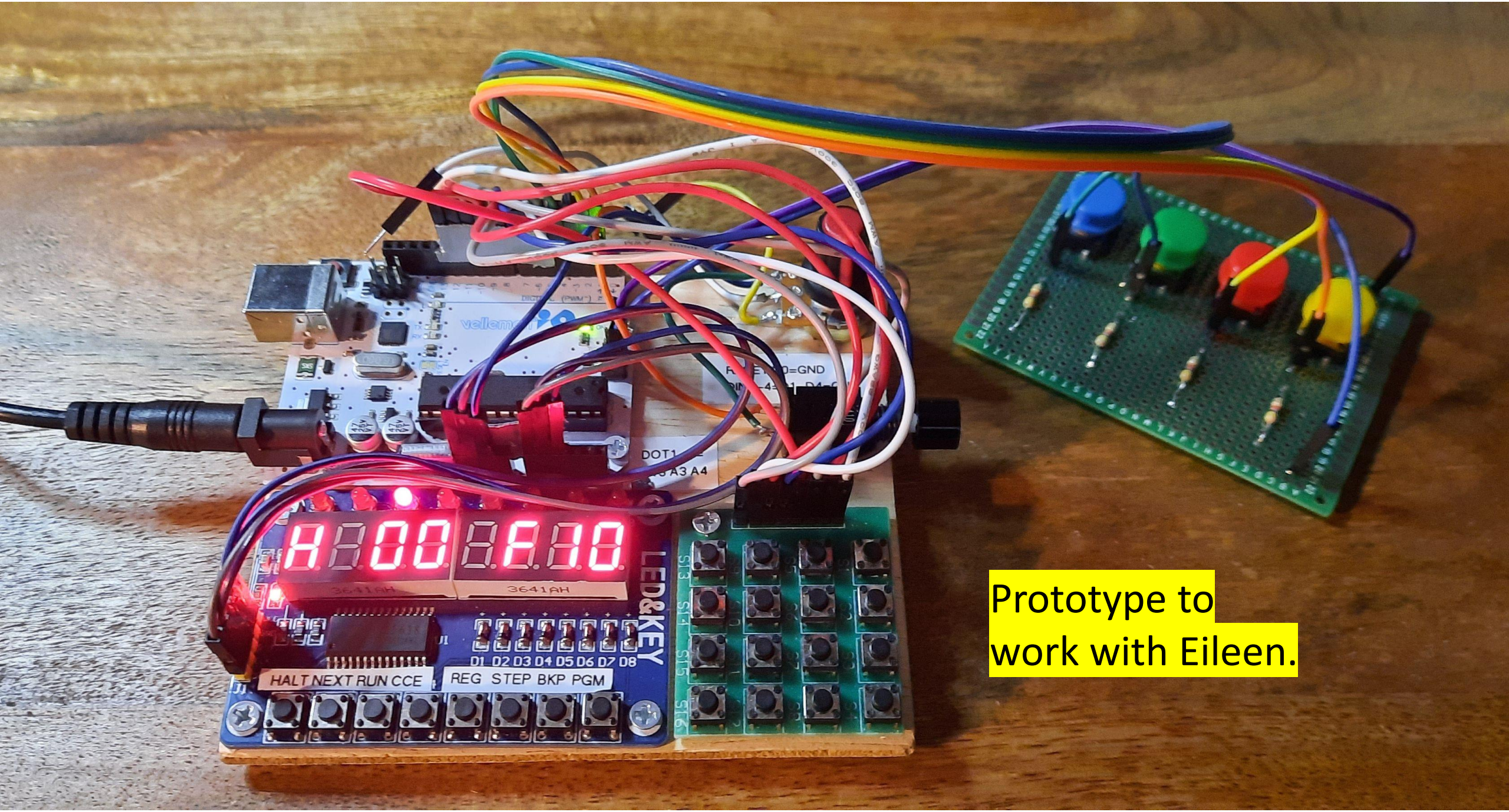

… and one more version, this time with an OLED SPI display. Brighter, faster.
microtronic
Microtronic Next Generation
RESET
SPK
POW
BATTE
GND VCC CLK MOSI RES DC CS
PC MNEMONICS
0E ADDI 2 1
H
0 1 2 3
CCE
PGM
BACK
FUN
HALT
ENTER
CANCEL
J1
HW-125
DURACELL
9V

Iterations 8 & 9 – 2021

- In January 2021, enthusiast Eileen (“Lilly”) picked up the old Uno-based version from 2016
- I wasn’t sure if the old code would still compile… but it did!
- Built her own Uno Microtronic
  - found some bugs
  - reintegrated some features back from the newer versions (Single Stepping, Breakpoints)
  - wanted to add some new features

    -> The 2016 Uno R3-based firmware got another round of refactoring
- I had to rebuild my 2016 version to work with her on the firmware

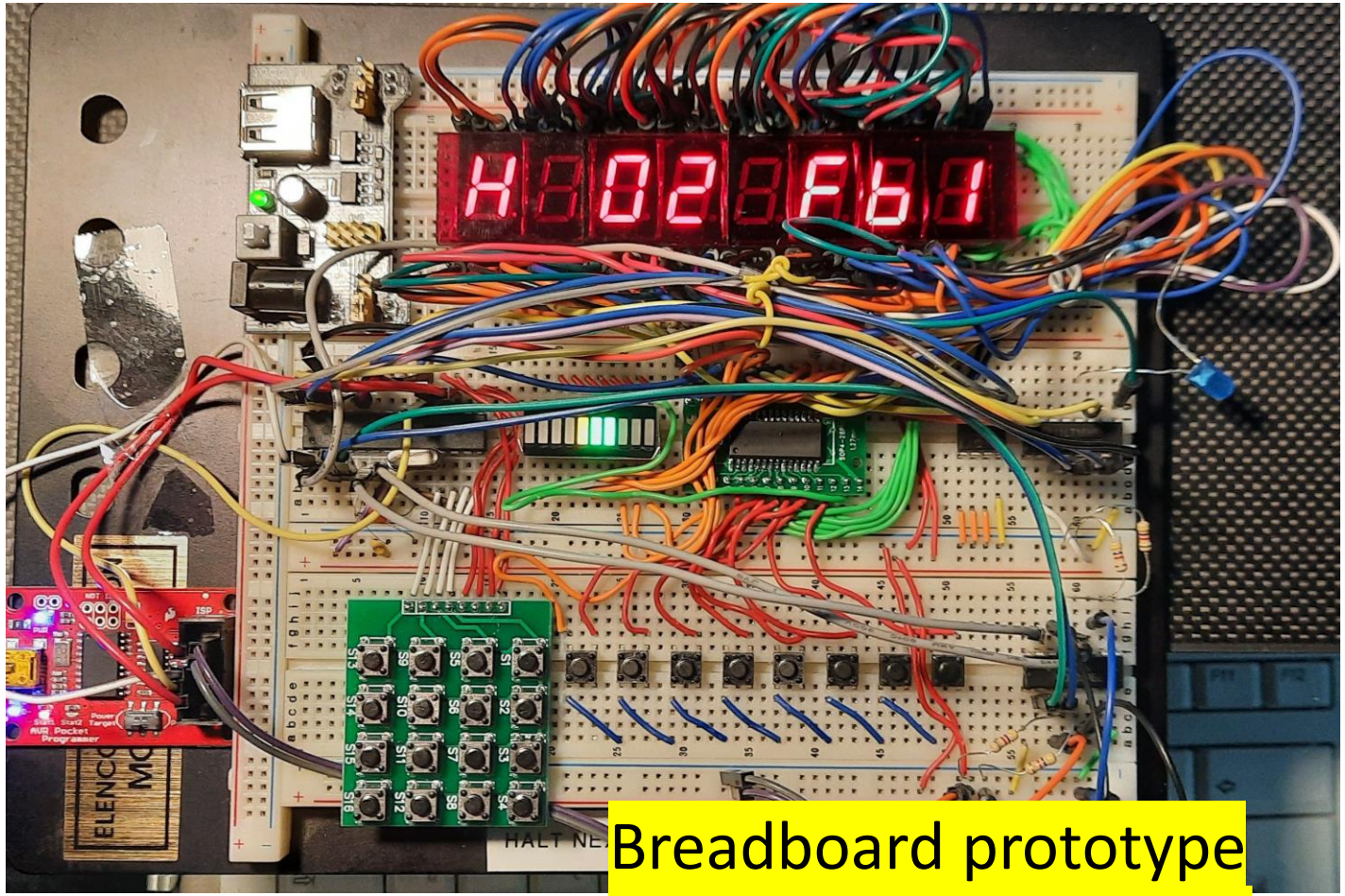




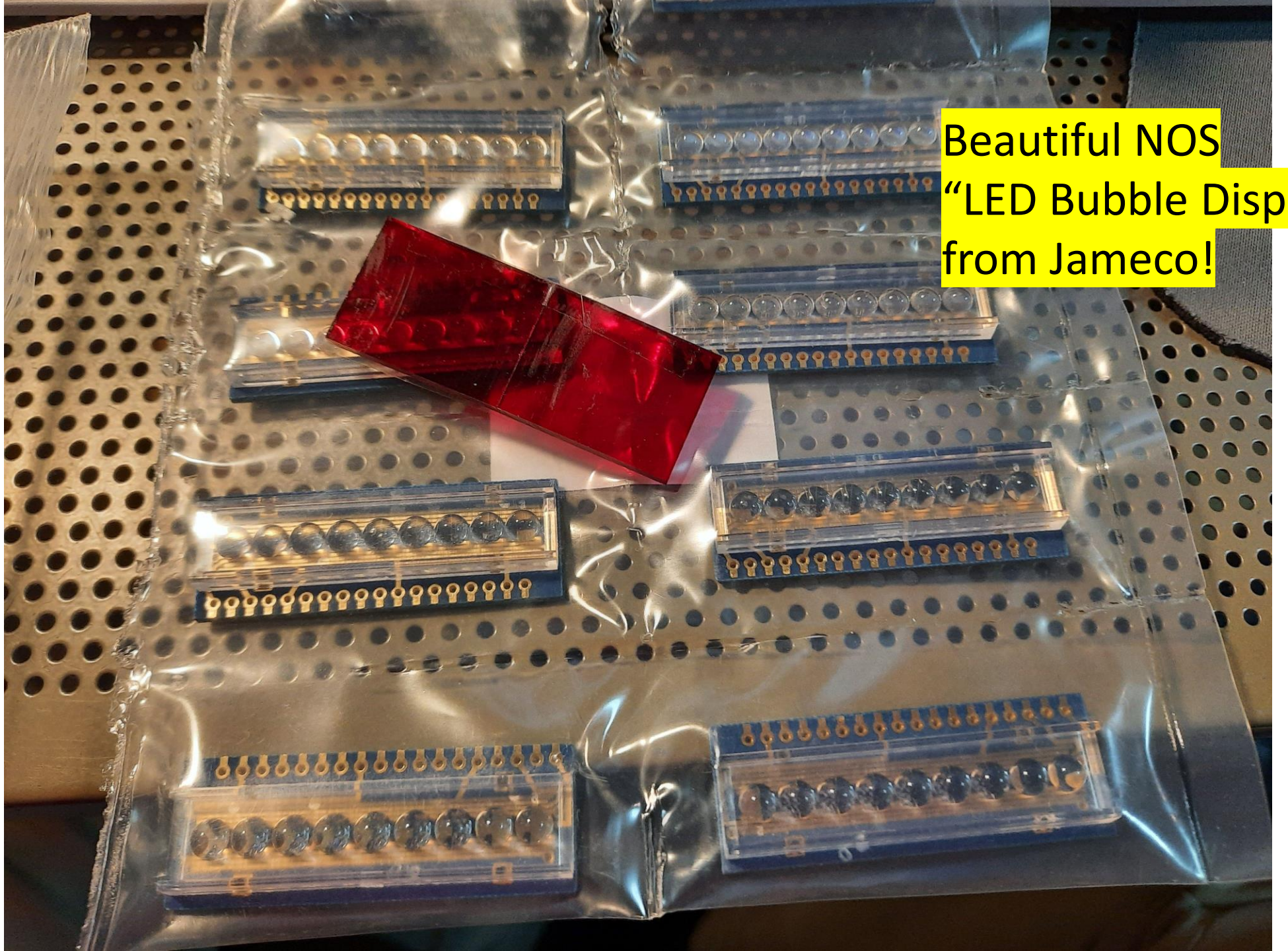
Prototype to
work with Eileen.

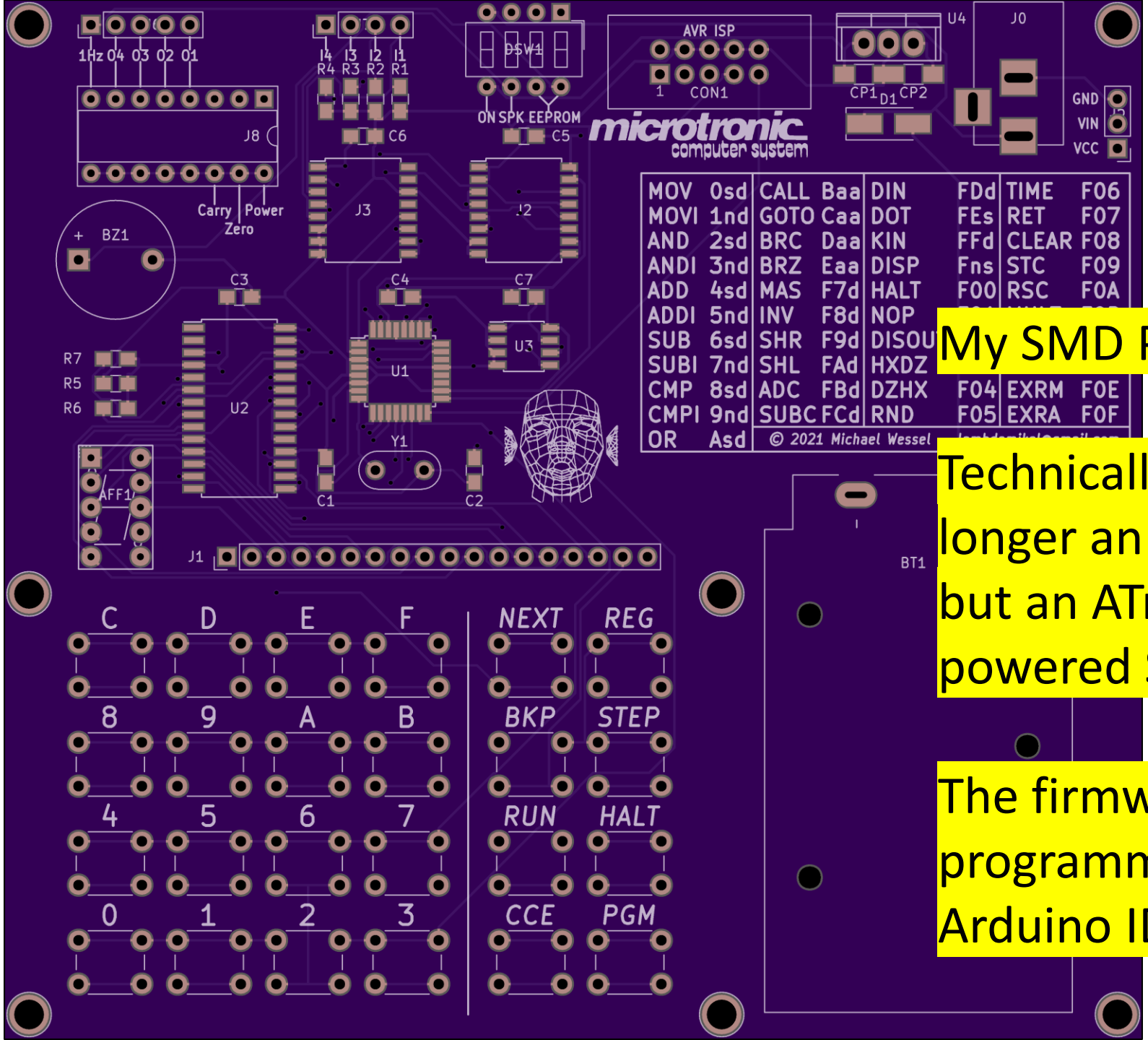
Iteration 10 – towards an SBC without modules

- **Using modules feels a bit like cheating…**
  - You want to build your own device
- **One more round**
  - Maximally authentic user experience
  - KiCAD custom SMD PCB
- **I re-engineered the LED&KEY module**
  - Schematics found on the Web
  - TM1638 driver chip only available as SMD
- **“Shrimped” the Arduino (ATmega 328)**
  - **Still programmable via Arduino IDE**
  - **Needs crystal, power supply, no more USB**
- **A 256 Kbit SPI EEPROM instead of the SDcard**
  - Good for storing ~ 42 full memory dumps!
- **LED display**
  - First individual 7segments (LOTs of wires…)
  - Then Bubble LED displays like in the original!
- **Removed the Real Time Clock**
- **Buzzer instead of sound / loudspeaker**



Breadboard prototype towards a module-less SBC PCB.

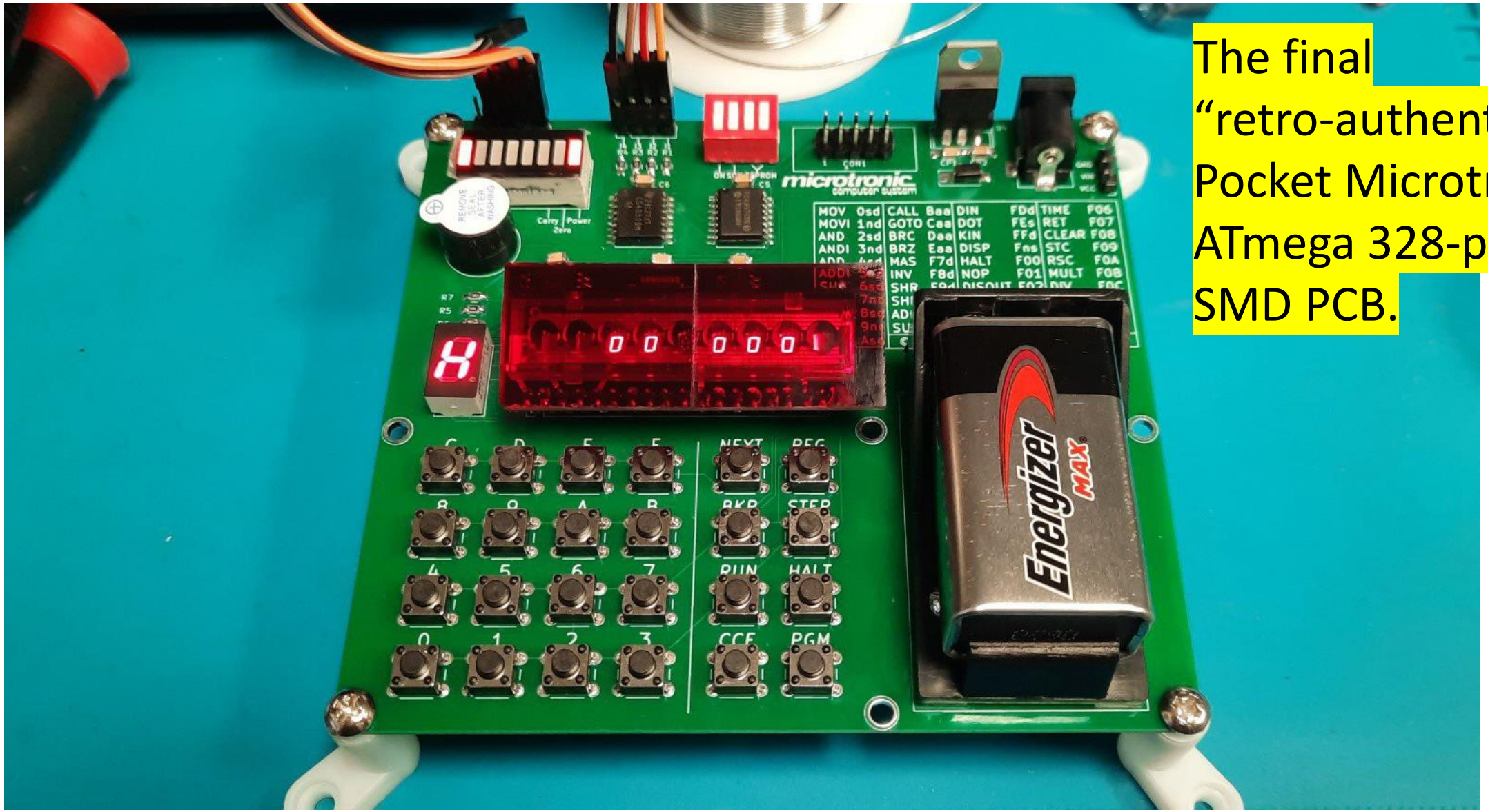

Beautiful NOS
“LED Bubble Displays”
from Jameco!



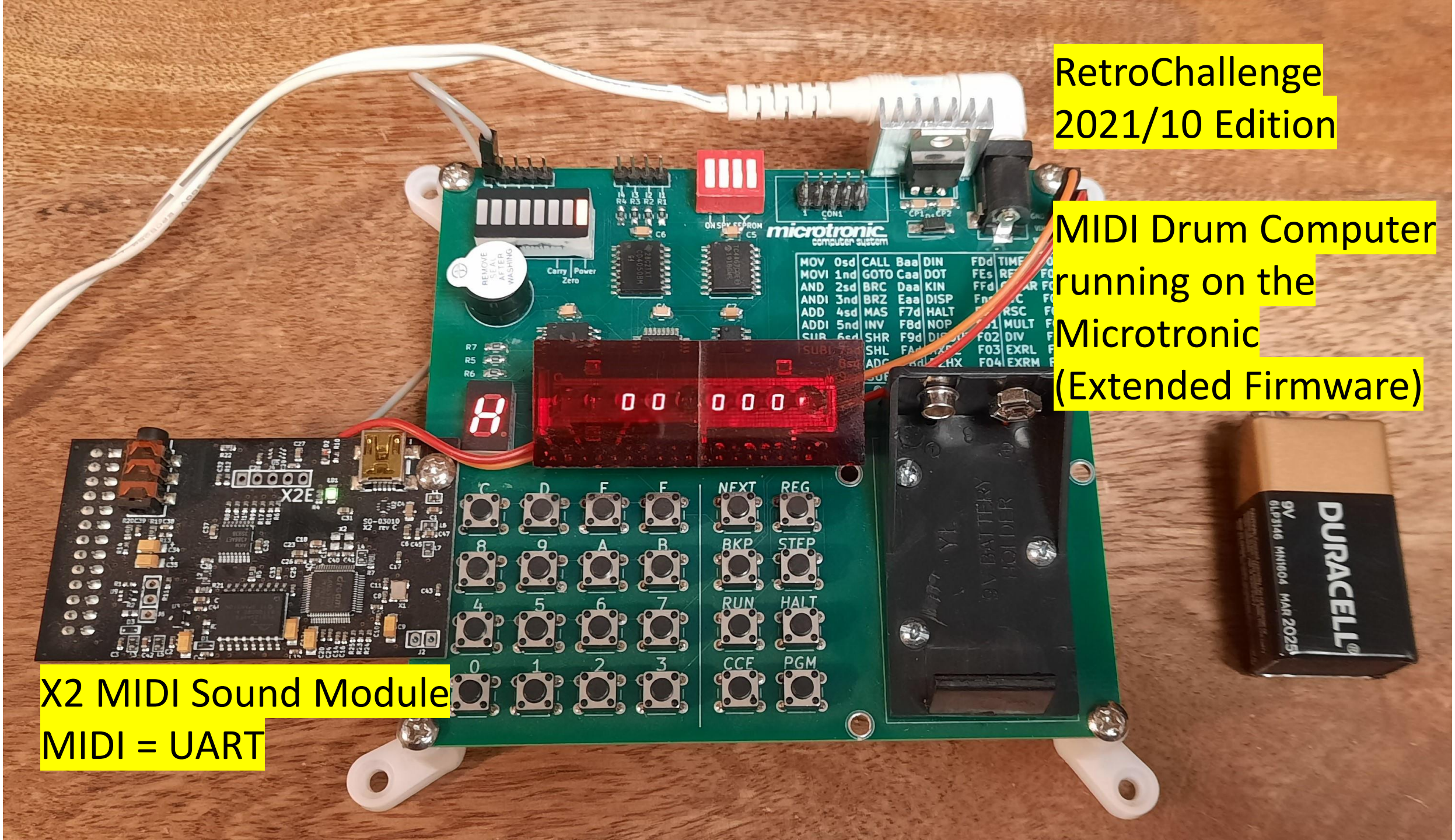
My SMD PCB.

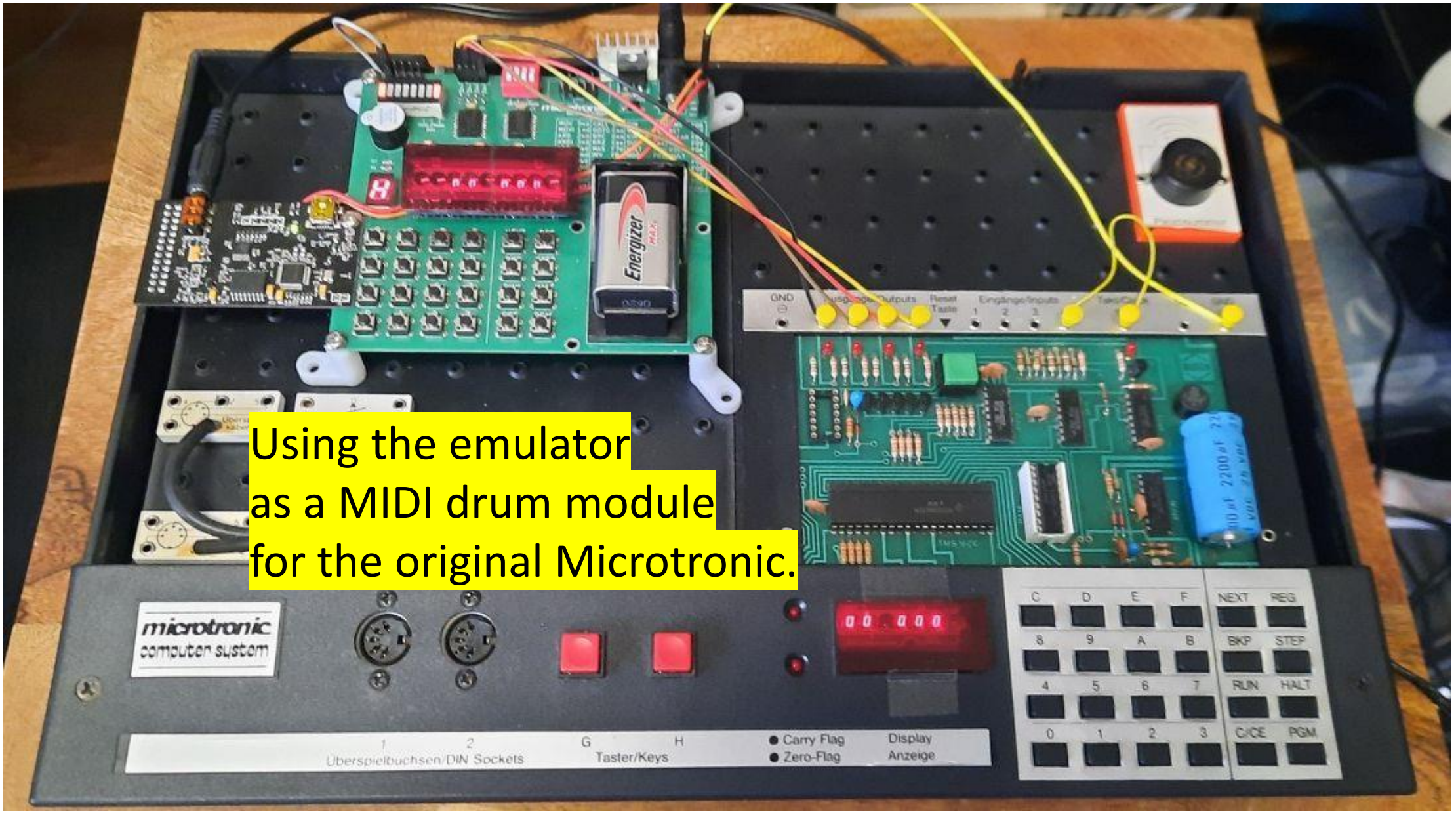
Technically, this is no longer an Arduino, but an ATmega 328-powered SMD SBC.

The firmware is still programmed using the Arduino IDE / software.



The final "retro-authentic" Pocket Microtronic ATmega 328-powered SMD PCB.


RetroChallenge
2021/10 Edition
MIDI Drum Computer
running on the
Microtronic
(Extended Firmware)
X2 MIDI Sound Module
MIDI = UART
microtronic
computer system
NEXT
REG
BKP
STEP
RUN
HALT
CCE
PGM
DURACELL

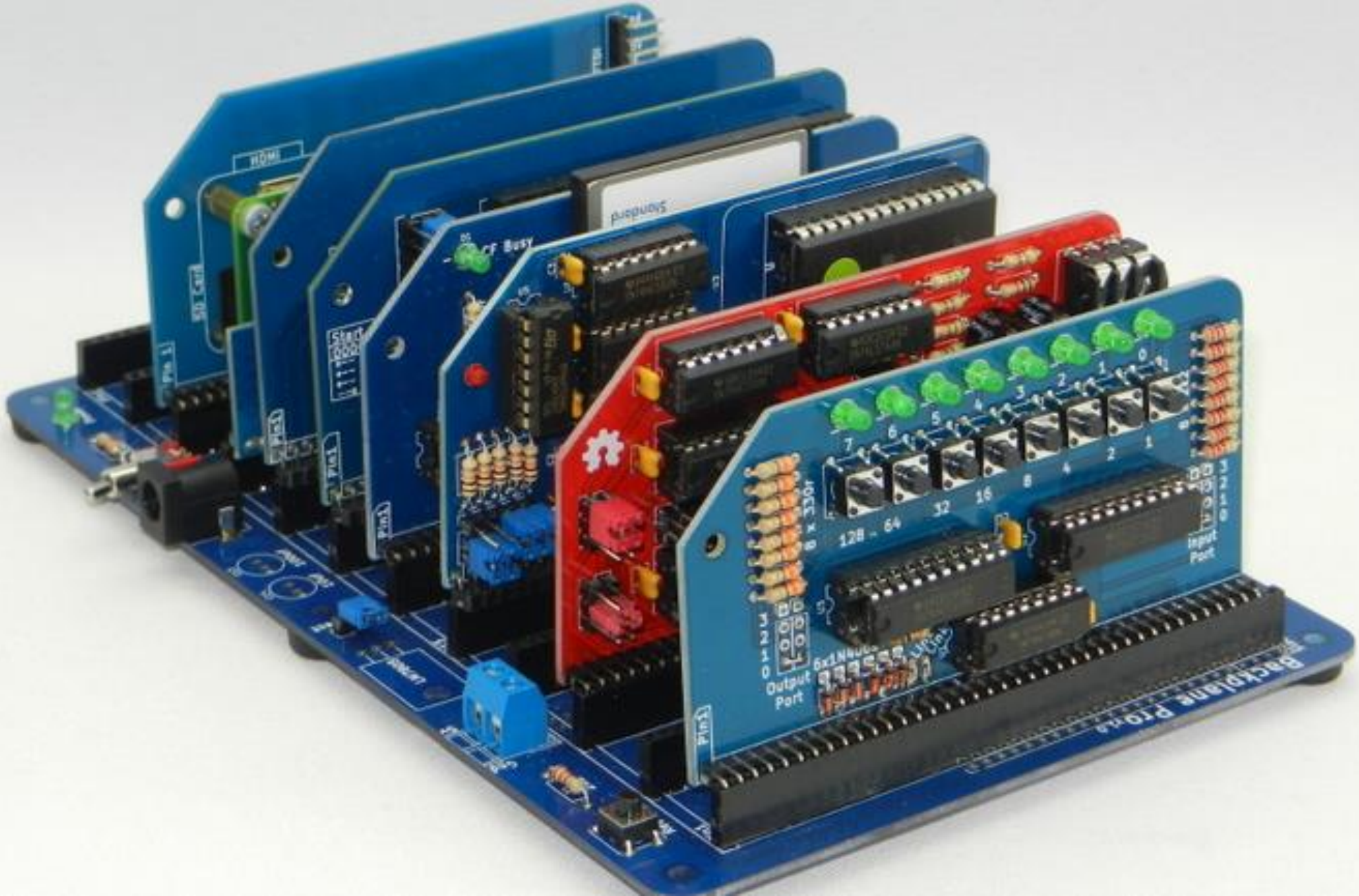

Using the emulator
as a MIDI drum module
for the original Microtronic.
microtronic
computer system
Überspielbuchsen/DIN Sockets
Taster/Keys
Carry Flag
Zero-Flag
Display
Anzeige
NEXT
REG
BKP
STEP
RUN
HALT
C/CE
PGM

RetroChallenge
Home « RC2014
https://rc2014.co.uk/what-is-rc2014/
For quick access, place your bookmarks here on the bookmarks toolbar. Manage bookmarks...

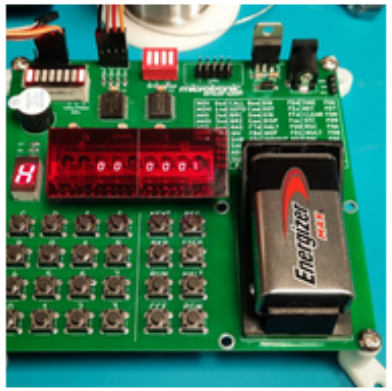
Home

The RC2014 is a simple 8 bit Z80 based modular computer originally built to run Microsoft BASIC. It is inspired by the home built computers of the late 70s and computer revolution of the early 80s. It is not a clone of anything specific, but there are suggestions of the ZX81, UK101, S100, Superboard II and Apple I in here. It nominally has 8K ROM, 32K RAM, runs at 7.3728MHz and communicates over serial at 115,200 baud.




HDMI
Output Port
Input Port
128 64 32 16 8 4 2 1
Pin1

Michael Wessel



Platform: Busch Microtronic

Project : A fully retro-authentic yet modern reimplemenation of the classic 1981 4bit educational single board computer. Hex in machinecode with a 7segment bubble display like its 1981!

Halftime Update: *MSW: one of the leading contenders this, the blog photos are beautiful and he's making real progress with some wonderful videos... definitely one to watch.*

Final Comments: *4 bit microprocessor, MIDI syth, drum machine, bubble LED displays, great videos, and what really sold me: using his emulated Microtronic as a drum module for an original 1981 Busch 2090 Microtronic!! Awesome job!*
MSW: Can't beat that last video for wow factor, I need to include more drum modules in my projects!

Final Status: *Prize "Winner"!!*


RETROCHALLENGE 2021/10
LET'S BOSS THIS!

What can be learned from a Microtronic?

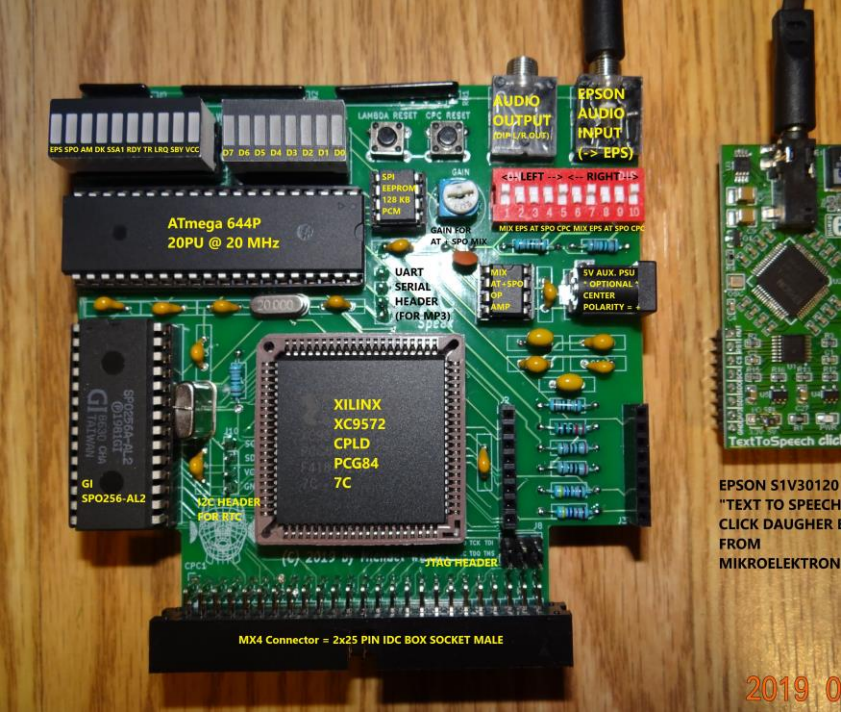
- A lot!
- Hex & Binary
- Basics of programming / machine language / algorithms
- GPIO / basics of physical computing
- Basic computer terminology (CPU, RAM, ROM, IO, …)
- However…

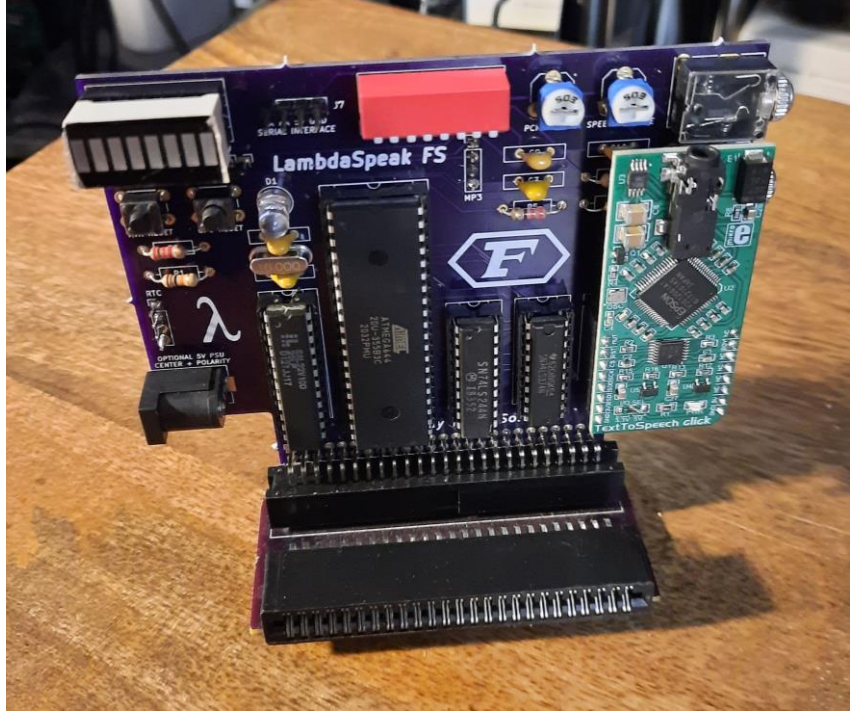
What can not be learned from a Microtronic?

- **Advanced addressing modes (indirect, indexed)**
  - I added program writeable RAM
  - Vacuous op-code extensions for using the extra memory
  - Needed for pattern sequencer in the drum computer
- **Recursion**
  - Like in the TMS itself, there is **no return stack**, only a single **return register** (no CALL-RET nesting)
  - I added a **return stack**, but still **no way to store and restore invocation contexts**
    - **Well, you can do it ONCE or TWICE using the EXTRA REGISTER sets –**
      makes a lot of sense given that there are no nested subroutine invocations possible in the original!
    - **The designers of the instruction set were very insightful**
    - **No way to implement a stack in memory with the original**
- **Self-modifying programs (Harvard architecture)**
- **Not the fastest…**
  - The emulator is quite a bit faster than the original
  - E.g., 999999 **DIV** 1 takes ~ 40 seconds with the original!
- **Only 4bit architecture**
- **Interrupts, DMA, modern communication protocols (I2C, SPI, UART, …)**
- **High-Level Programming Languages**

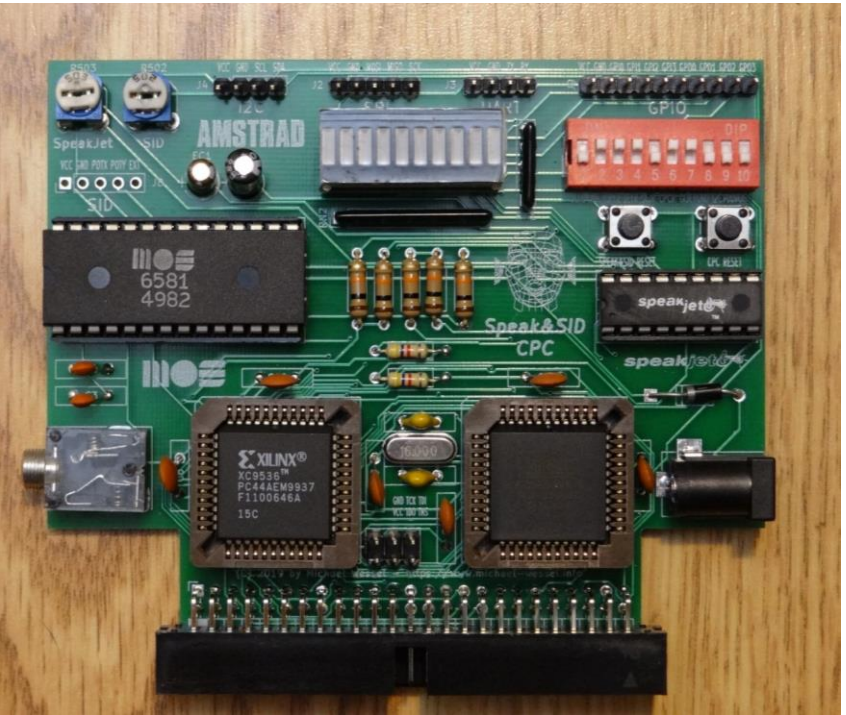
Conclusion & take away

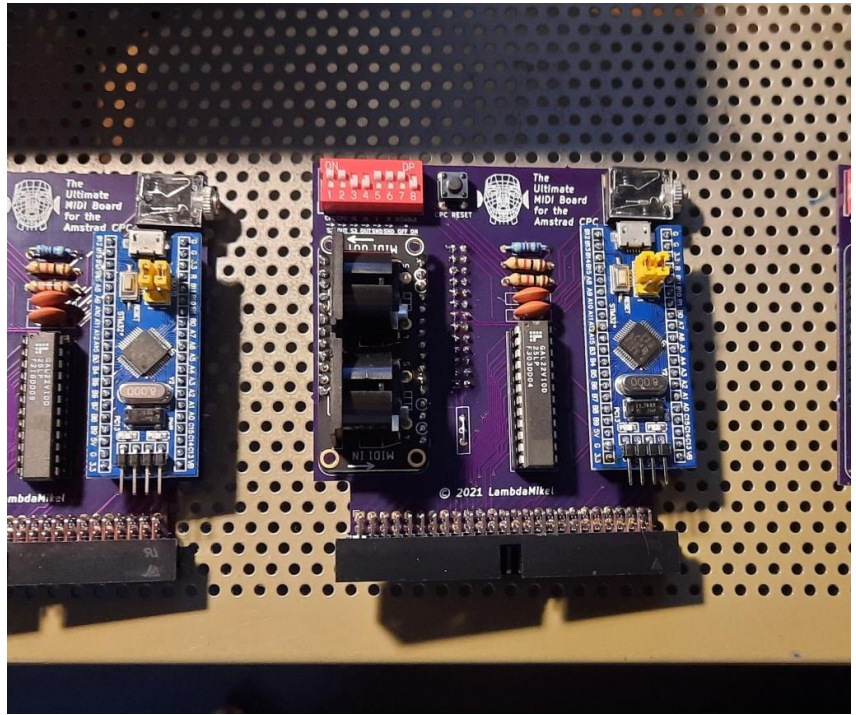
- **Hardware is frequently just software these days, even on low levels**
  - **And, in many cases, has been since the mid / late 70s**
  - **Both the original Microtronic and my homebrew versions are just complex firmware running on Microcontrollers**
  - The TMS1600 and the AVR / ATmega (328, 2560)
  - Many CPUs are software “IP” (Intellectual Property) Cores these days (Verilog designs etc.)
  - ARM, the world’s dominant CPU architecture, is running an entirely IP-based licensing business
- **Arduino is a great learning platform and steppingstone**
  - Variety - quite inexpensive boards and shields / modules (modular electronics)
  - Arduino IDE, drivers, tutorials, community
  - **Allows you to step up your game – from BMOWs over bread boards to PCBs via “Shrimping”**
- **Electronic Design Automation (EDA)**
  - KiCAD, OshPark, PCB Way, Seeed
  - Receiving your first self-designed PCB back from a PCB printing service in the mail is a magical moment!
  - There is always one more feature, one more little fix, one more improvement, one more thing to try, …
- **Hobbyists can now achieve what would have been impossible 10 / 15 years ago**
  - **I don’t have a formal EE background (CS PhD only); if I succeeded at this, so can you!**
- **Community**
  - **Share / OpenSource your designs (GitHub, Hackaday, OshPark / Seeed gallery)**
  - Collaborators can find you, very motivational

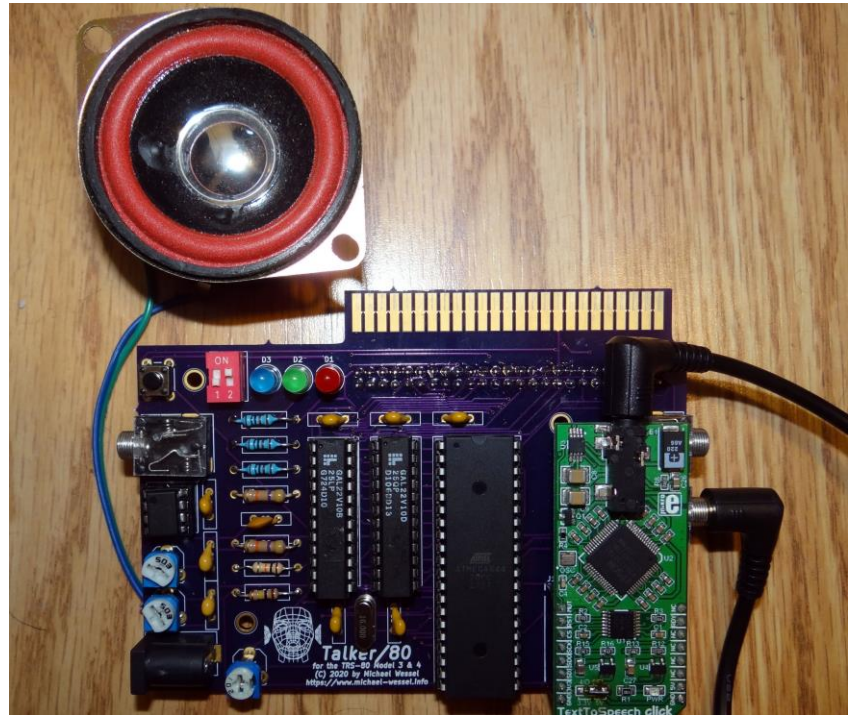

AUDIO OUTPUT
EPSON AUDIO INPUT (-> EPS)
ATmega 644P
20PU @ 20 MHz
XILINX
XC9572
CPLD
PCG84
7C
GI
SPO256-AL2
I2C HEADER FOR RTC
JTAG HEADER
UART SERIAL HEADER (FOR MP3)
MX4 Connector = 2x25 PIN IDC BOX SOCKET MALE
EPSON S1V30120
"TEXT TO SPEECH
CLICK DAUGHER
FROM
MIKROELEKTRON


LambdaSpeak FS
TextToSpeech click


Talker/80


AMSTRAD
Speak&SID
CPC
XILINX
XC9536
PC44AEM9937
F1100646A
15C


The Ultimate MIDI Board for the Amstrad CPC
© 2021 LambdaMikel


Talker/80
for the TRS-80 Model 3 & 4
(C) 2020 by Michael Vessel
https://www.michael-vessel.info
TextToSpeech click

Backup slides

- Other educational computers in Germany

Kosmos

Computer-Praxis

CP 1

Der spielerische Einstieg in ein Gebiet, dem die Zukunft gehört.
Geeignet für Schule und Fortbildung.



Mit dem Computer auf Du und Du – mit dem KOSMOS-Computer ein faszinierendes, großes Hobby beginnen.

kosmos®

Wichtig! Nur hier die Stromversorgung (Gleich- oder Wechselspannung) anschließen!

P .000

0 1 2 3 4 5 6 7 8 9

STEP Schritt | STP Stopp | RUN Lauf | CAL Cass. lesen

CLR Irrtum | ACC Akku | CAS Cass. speichern

PC Programmzähler | OUT auslesen | INP eingeben

7 8 9
4 5 6
1 2 3
0


MICROCOMPUTER
MASTER LAB
PHILIPS
Programmieren ohne Vorkenntnisse
Leichter Einstieg in die neueste Computer-Technik.
Microprozessor mit 38000 Transistor-Funktionen.
Integrierte Cassetten-Elektronik, 12 Festprogramme.
Hochintegrierte Schaltkreise, 8-stelliges Leuchtdisplay, 12 Leuchtdioden, Lautsprecher,
27 Eingabe- und Funktionstasten.
1024 x 8 Bits freie Speicherplätze.
Netzteil 220/8 Volt (VDE).
Über 150 Programmier-Experimente:
Lauflicht, Elektronen-Rechner,
Reaktionstester, Spiele, Lichtschrankenzähler, Radiowecker,
Digitaluhr.....
Beiliegende Speicher-Cassette
als Programm-Bibliothek mit bereits
überspielten Programmen.
Ausführliches Anleitungsbuch.
PHILIPS
EXPERIMENTIER-
TECHNIK
PHILIPS


GAMA
Anschluß für Zusatzbausteine
Netzgerät
Recorder
+ 5 Volt
Ausgang F 1
Ausgang F 2
Ausgang F 3
0 Volt/Masse
Eingang SA
Eingang SB
Start/Spiele
• Netzgerät anschließen
• Betriebsschalter auf 1
• Anzeige: HALLO
• Taste SP drücken
• Anzeige: SPIELE
• Spieltaste 0 bis B
• Spiel erscheint auf der Anzeige
• Spielende Taste RS drücken
• Neues Spiel
Programm/Spiele
0 = Laufschrift
1 = Ampelsteuerung
2 = Blinklicht, gelb
3 = Lauflicht → LED
4 = Dezimal-/Dualzahlen → LED
5 = Sekundenuhr
6 = Sirene
7 = Würfel → mit SB würfeln
8 = 6 aus 49 → mit SA auswählen
9 = Potenzen von 2
A = Rechner
PHILIPS
SSA
LED
7 6 5 4 3 2 1 0
0
1
RS
F3
F2
F1
SA
SB
C D E F
8 9 A B
4 5 6 7
0 1 2 3
A↔D SV
RUN LD
ME− CPU
ME+ SP
LED
Ta
SV

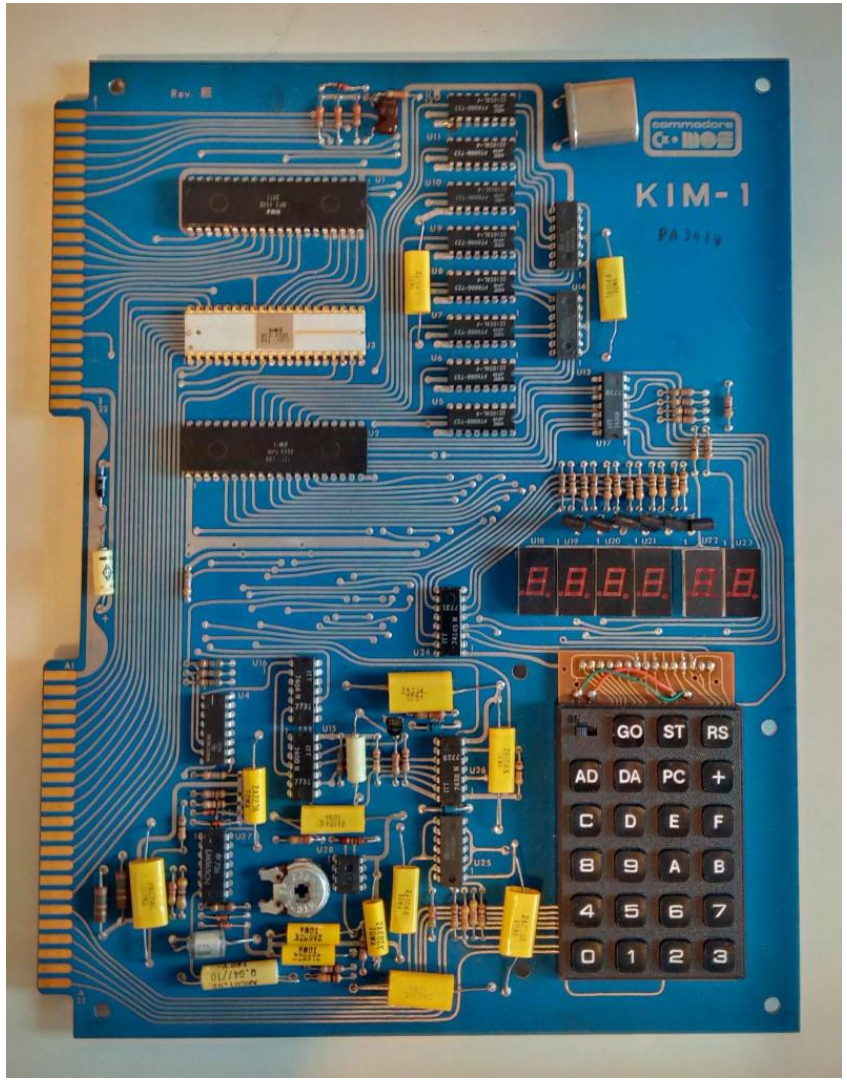
Other well-known single board computers

- KIM-1
- SC/MP
- RCA’s COSMAC
- HEATHKIT ET-3400

- Microprofessor MPF-1B / MPF-IP

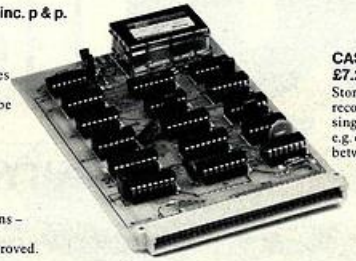


1976, KIM-1, 6502, US $245,



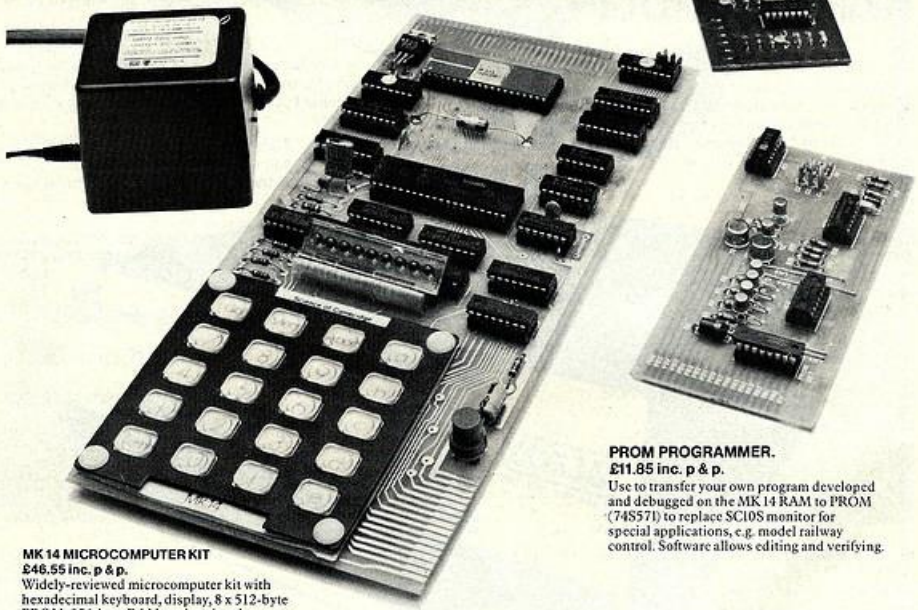
Now, the complete MK 14 micro-computer system from Science of Cambridge

VDU MODULE. £33.75
(£26.85 without character generator) inc. p & p.
Display up to ½K memory (32 lines x 16 chars, with character generator; or 4096 spot positions in graphics mode) on UHF domestic TV. Eurocard-sized module includes UHF modulator, runs on single 5 V supply. Complete ascii upper-case character set can be mixed with graphics.



CASSETTE INTERFACE MODULE.
£7.25, inc. p & p.
Store and retrieve programs on any cassette recorder. Use for serial transmission down single line at up to 110 baud (teletype speed), e.g. over telephone line, and to communicate between two or more MK 14s.

POWER SUPPLY. £6.10 inc. p & p.
Delivers 8 V at 600 mA from 220/240 V mains – sufficient to drive all modules shown here simultaneously. Sealed plastic case, BS-approved.



PROM PROGRAMMER.
£11.85 inc. p & p.
Use to transfer your own program developed and debugged on the MK 14 RAM to PROM (74S571) to replace SC10S monitor for special applications, e.g. model railway control. Software allows editing and verifying.

MK 14 MICROCOMPUTER KIT
£46.55 inc. p & p.
Widely-reviewed microcomputer kit with hexadecimal keyboard, display, 8 x 512-byte PROM, 256-byte RAM, and optional 16-lines I/O plus further 128 bytes of RAM.
Supplied with free manual to cover operations of all types – from games to basic maths to electronics design. Manual contains programs plus instructions for creating valuable personal programs. Also a superb education and training aid – an ideal introduction to computer technology.
Designed for fast, easy assembly; supplied with step-by-step instructions.

Science of Cambridge Ltd

6 Kings Parade, Cambridge, CAMBS., CB2 1SN.
Tel: 0223 311488.

To order, complete coupon and post to Science of Cambridge
Return as received within 14 days for full money refund if not completely satisfied.

To: Science of Cambridge Ltd, 6 Kings Parade, Cambridge, Cambs., CB2 1SN.

Please send me:
- ☐ MK 14 standard kit @ £46.55.
- ☐ Extra RAM @ £4.14 per pair.
- ☐ RAM I/O device @ £8.97.
- ☐ VDU module including character generator @ £33.75.
- ☐ VDU module without character generator @ £26.85.
- ☐ Cassette interface module @ £7.25.
- ☐ PROM programmer @ £11.85.
- ☐ Power supply @ £6.10.
- ☐ Full technical details of the MK 14 System, with order form.

All prices include p+p and VAT.

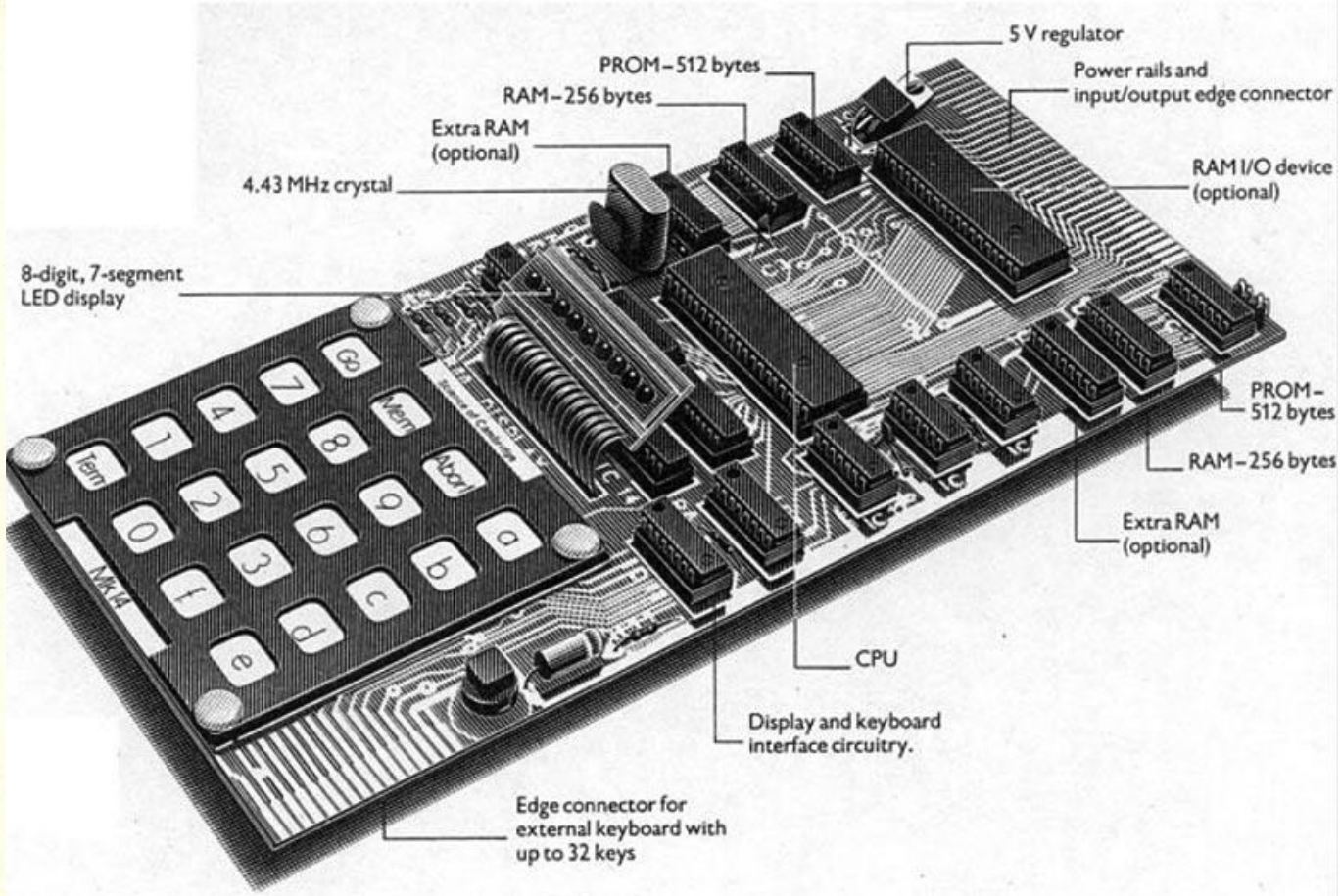
I enclose cheque/MO/PO for £_____(total).

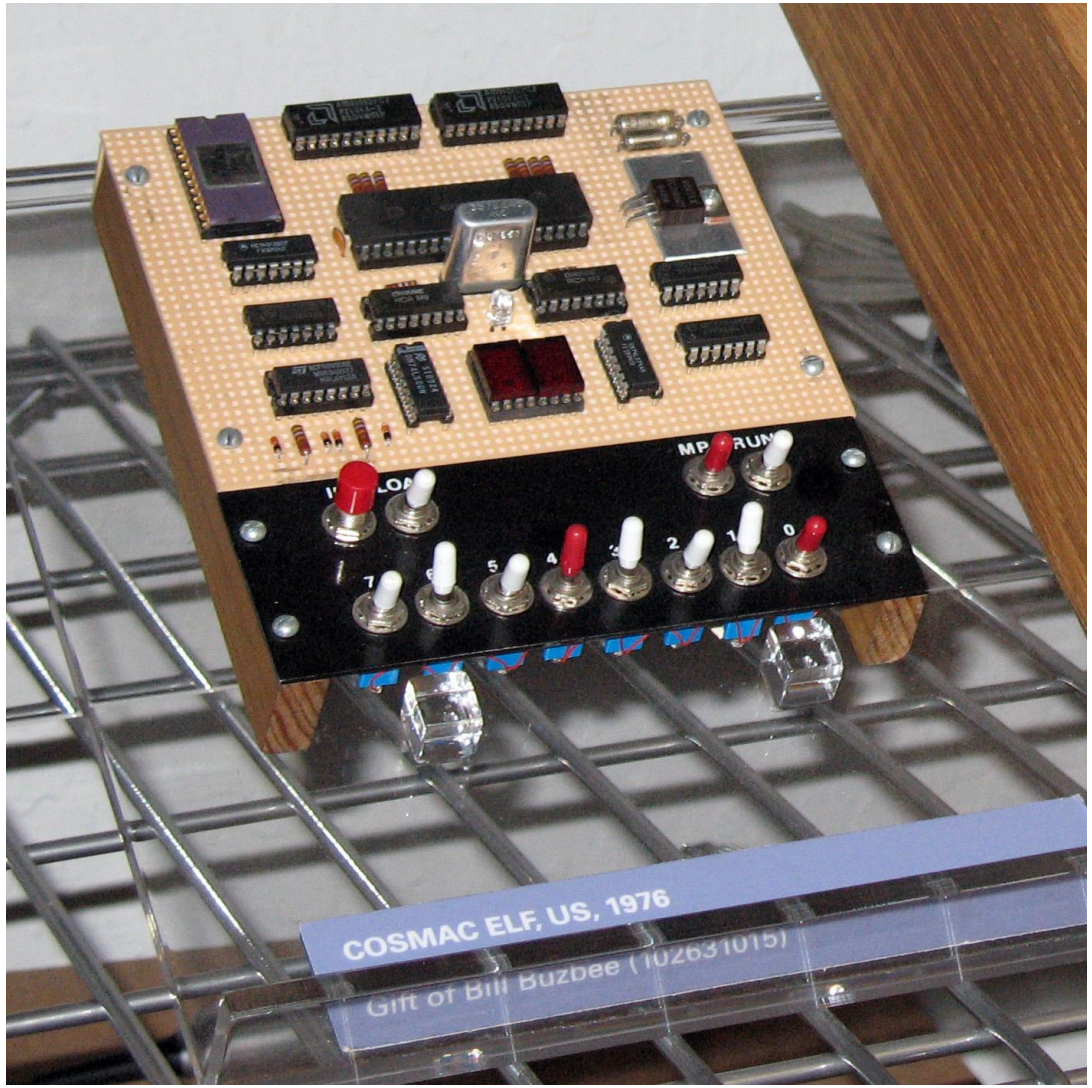
Name_____

Address (please print)_____

PCW/11/80

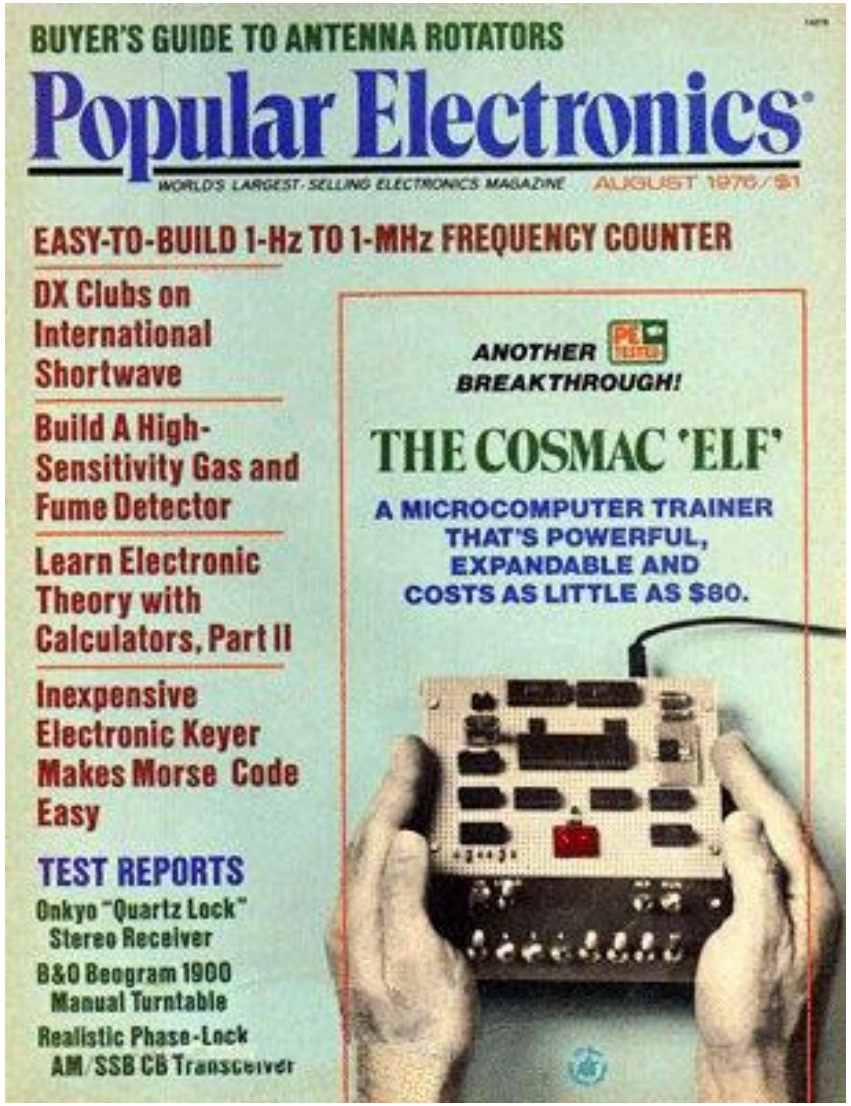
1978, SC/MP, £39.95,

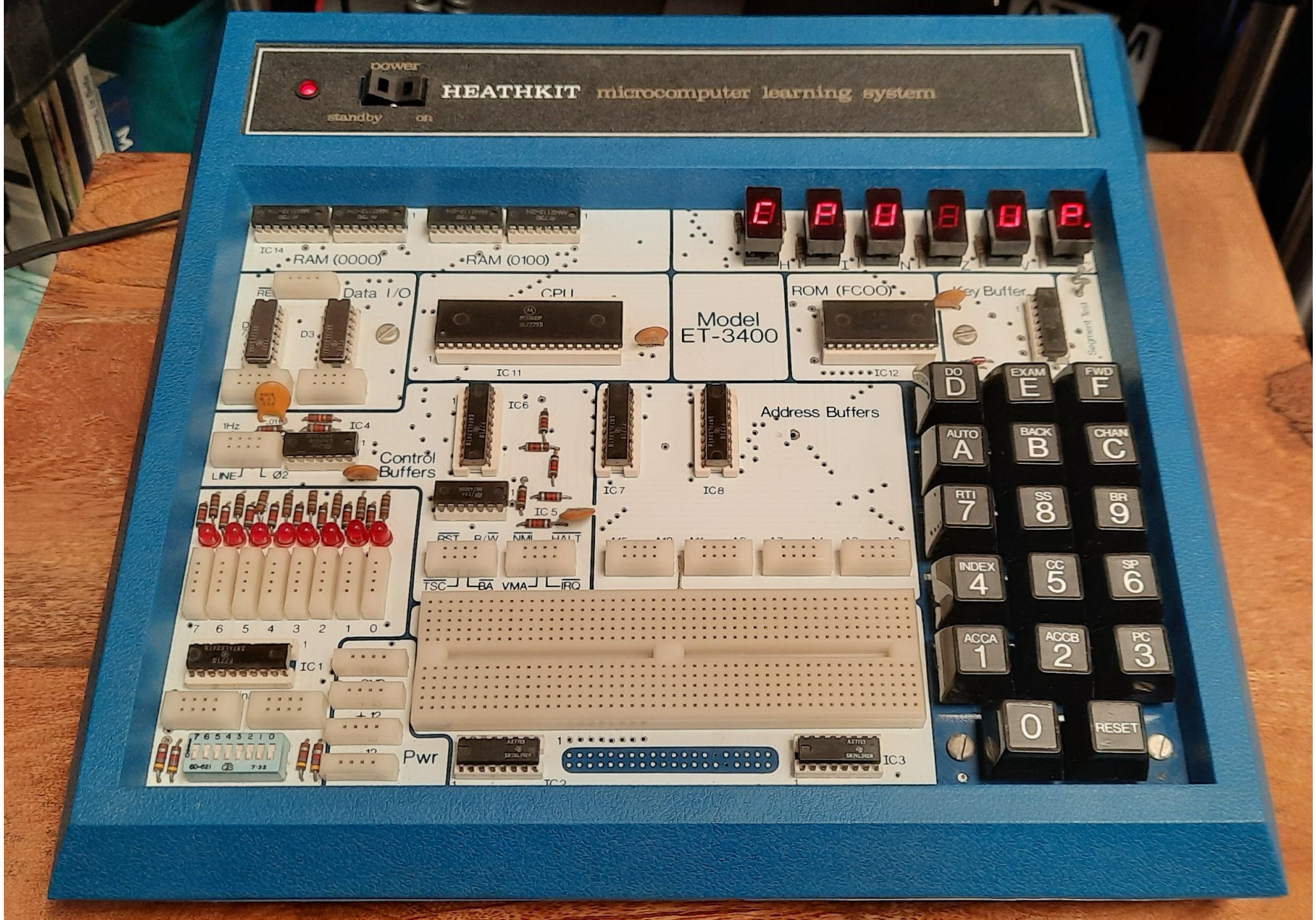


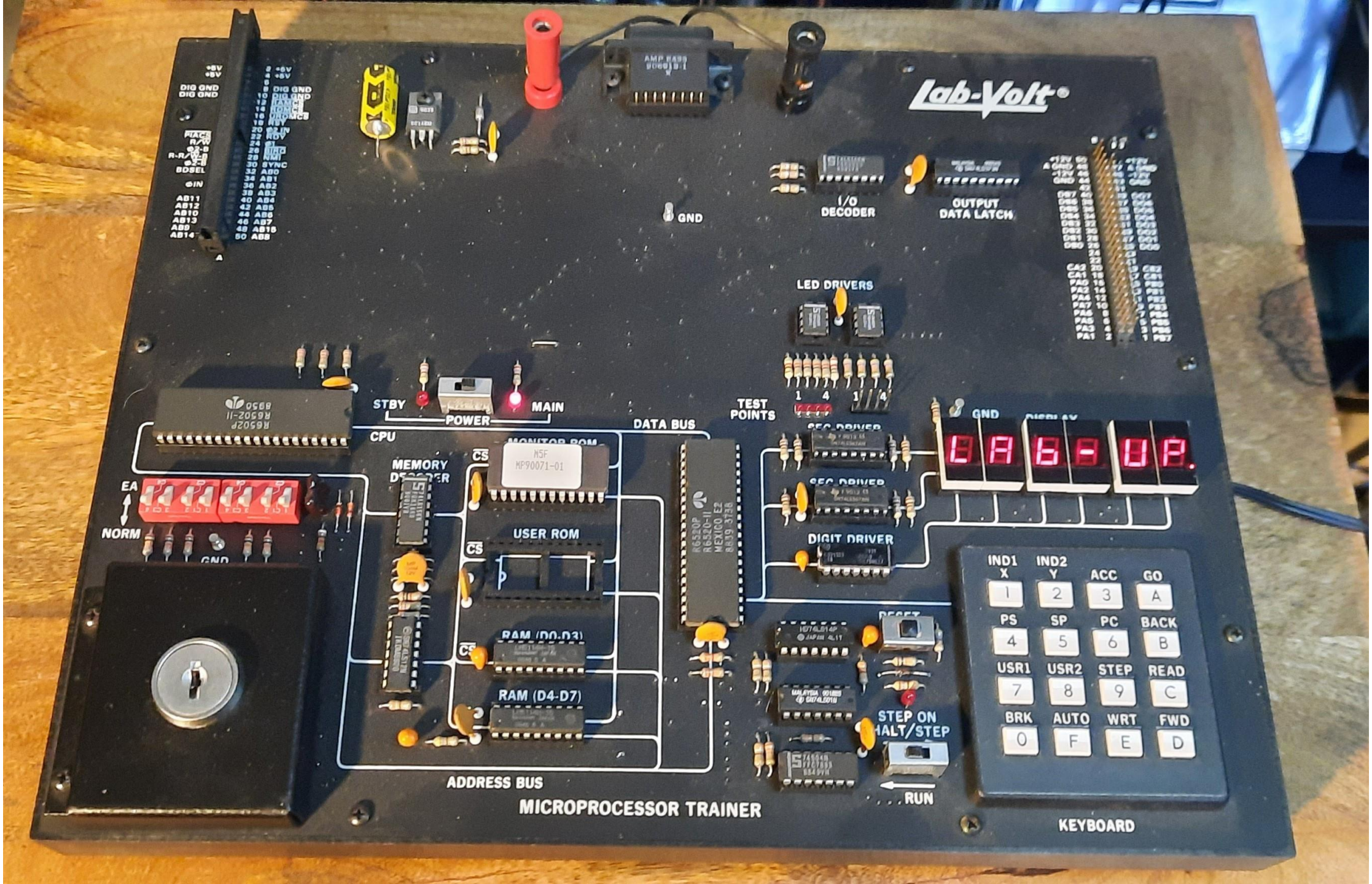


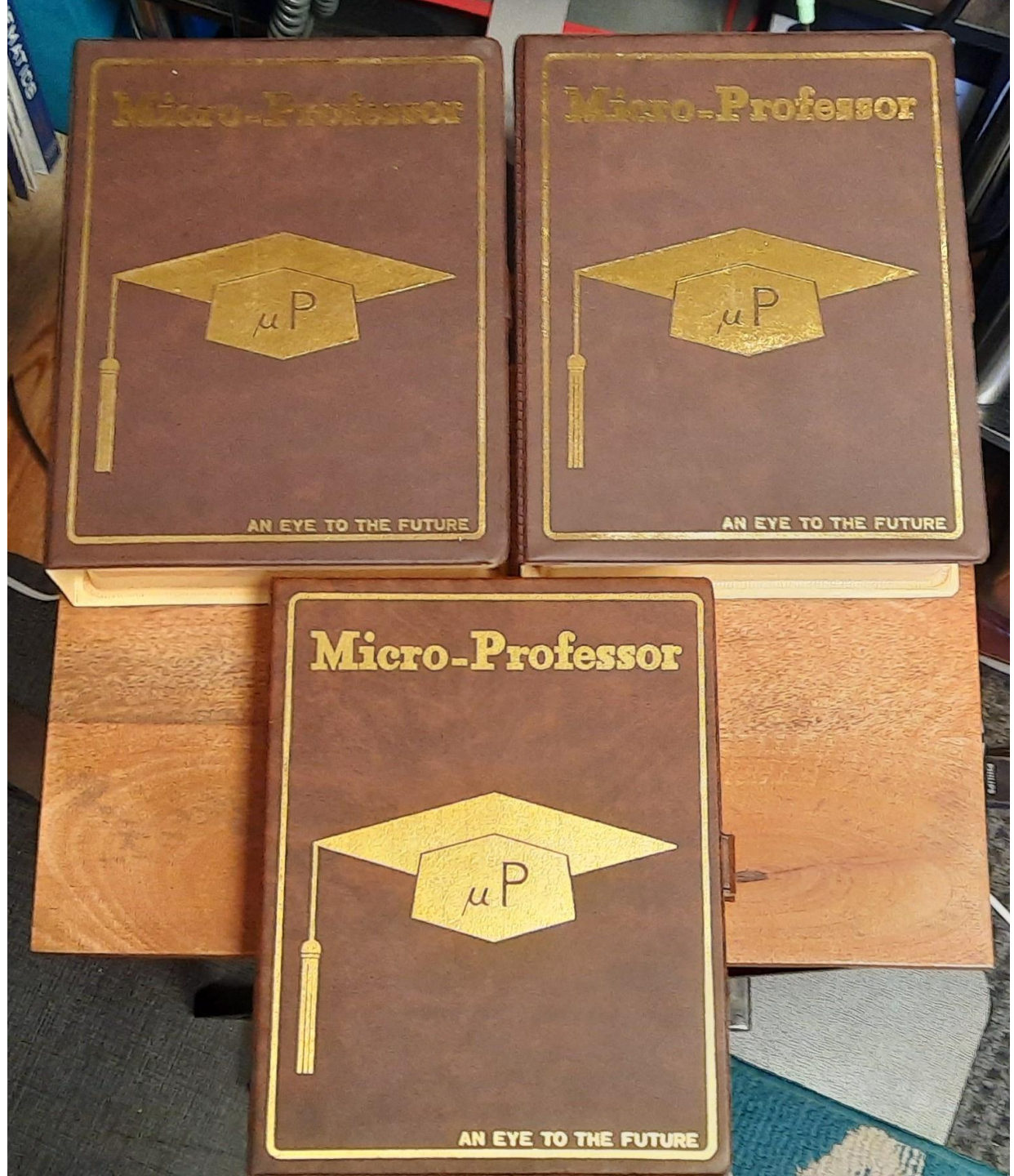
The **COSMAC Elf** was an RCA 1802 microprocessor-based computer described in a series of construction articles in *Popular Electronics* magazine in 1976 and 1977.

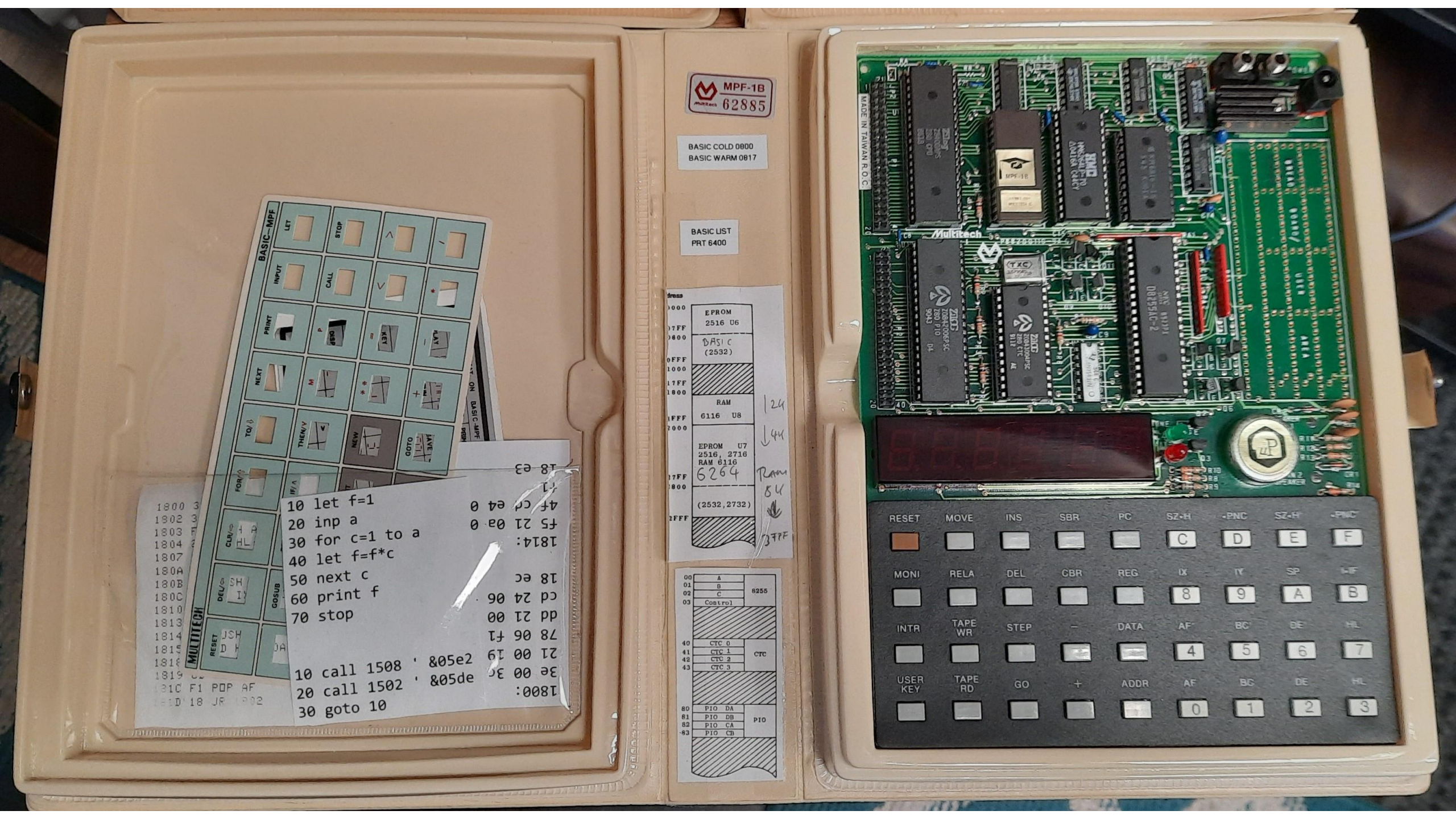
RCA CDP1802

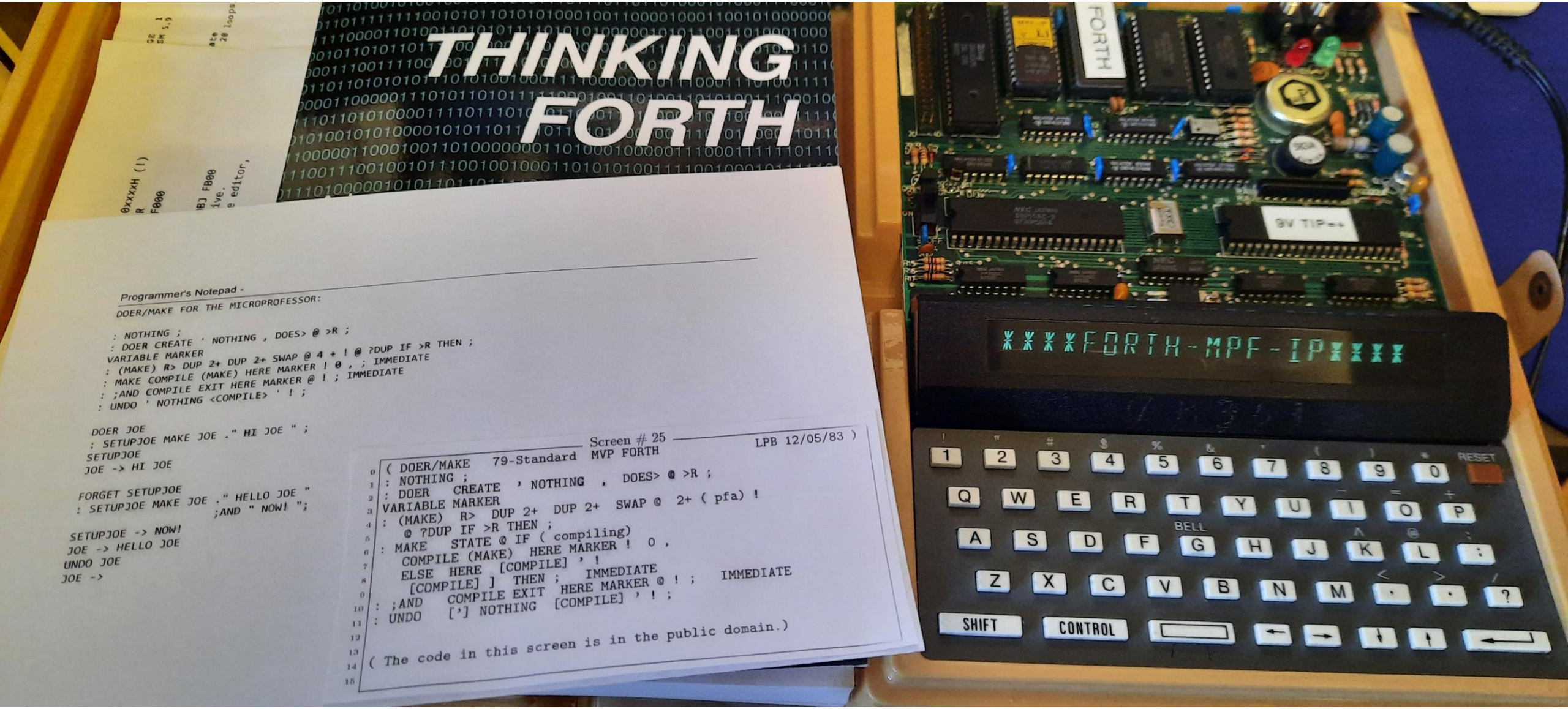



power
standby
on
HEATHKIT microcomputer learning system
RAM (0000)
RAM (0100)
Data I/O
CPU
Model ET-3400
ROM (FC00)
Key Buffer
Control Buffers
Address Buffers
Pwr
DO
EXAM
FWD
AUTO
BACK
CHAN
RTI
SS
BR
INDEX
CC
SP
ACCA
ACCB
PC
RESET


Lab-Volt
I/O DECODER
OUTPUT DATA LATCH
GND
LED DRIVERS
TEST POINTS
STBY
MAIN
POWER
CPU
DATA BUS
MEMORY DECODER
USER ROM
RAM (D0-D3)
RAM (D4-D7)
EA
NORM
DIGIT DRIVER
RESET
STEP ON HALT/STEP
RUN
ADDRESS BUS
MICROPROCESSOR TRAINER
IND1 X
IND2 Y
ACC
GO
PS
SP
PC
BACK
USR1
USR2
STEP
READ
BRK
AUTO
WRT
FWD
KEYBOARD

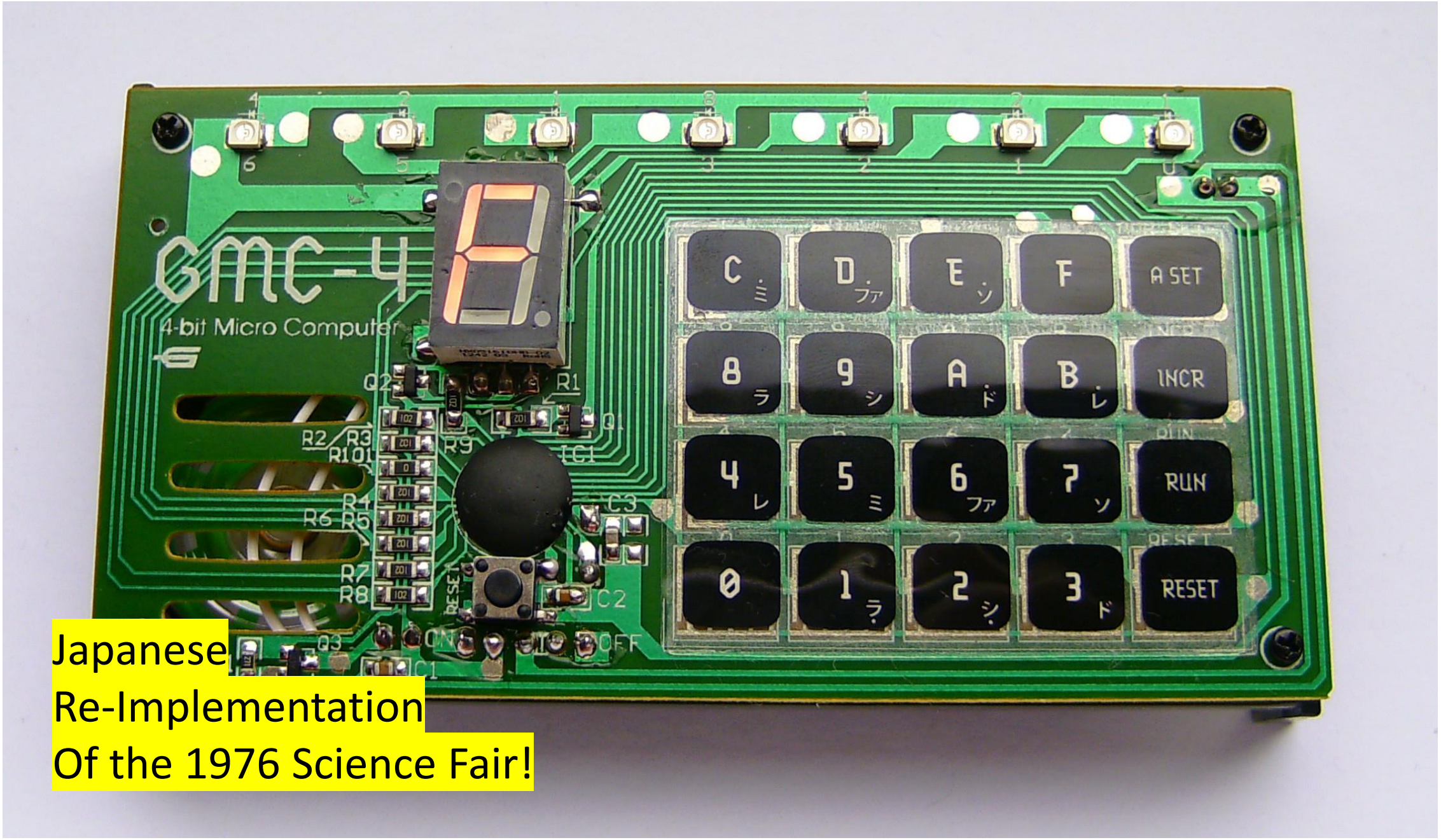

Micro-Professor
μP
AN EYE TO THE FUTURE
Micro-Professor
μP
AN EYE TO THE FUTURE
Micro-Professor
μP
AN EYE TO THE FUTURE


MPF-1B
62885
BASIC COLD 0800
BASIC WARM 0817
BASIC LIST
PRT 6400
MADE IN TAIWAN R.O.C.
Multitech
BASIC-MPF
MULTITECH
10 let f=1
20 inp a
30 for c=1 to a
40 let f=f*c
50 next c
60 print f
70 stop
10 call 1508 ' &05e2
20 call 1502 ' &05de
30 goto 10
EPROM
2516 U6
BASIC
(2532)
RAM
6116 U8
EPROM U7
2516, 2716
RAM 6116
6264
(2532,2732)
8255
CTC
PIO
RESET
MOVE
INS
SBR
PC
MONI
RELA
DEL
CBR
REG
INTR
TAPE WR
STEP
DATA
USER KEY
TAPE RD
GO
ADDR

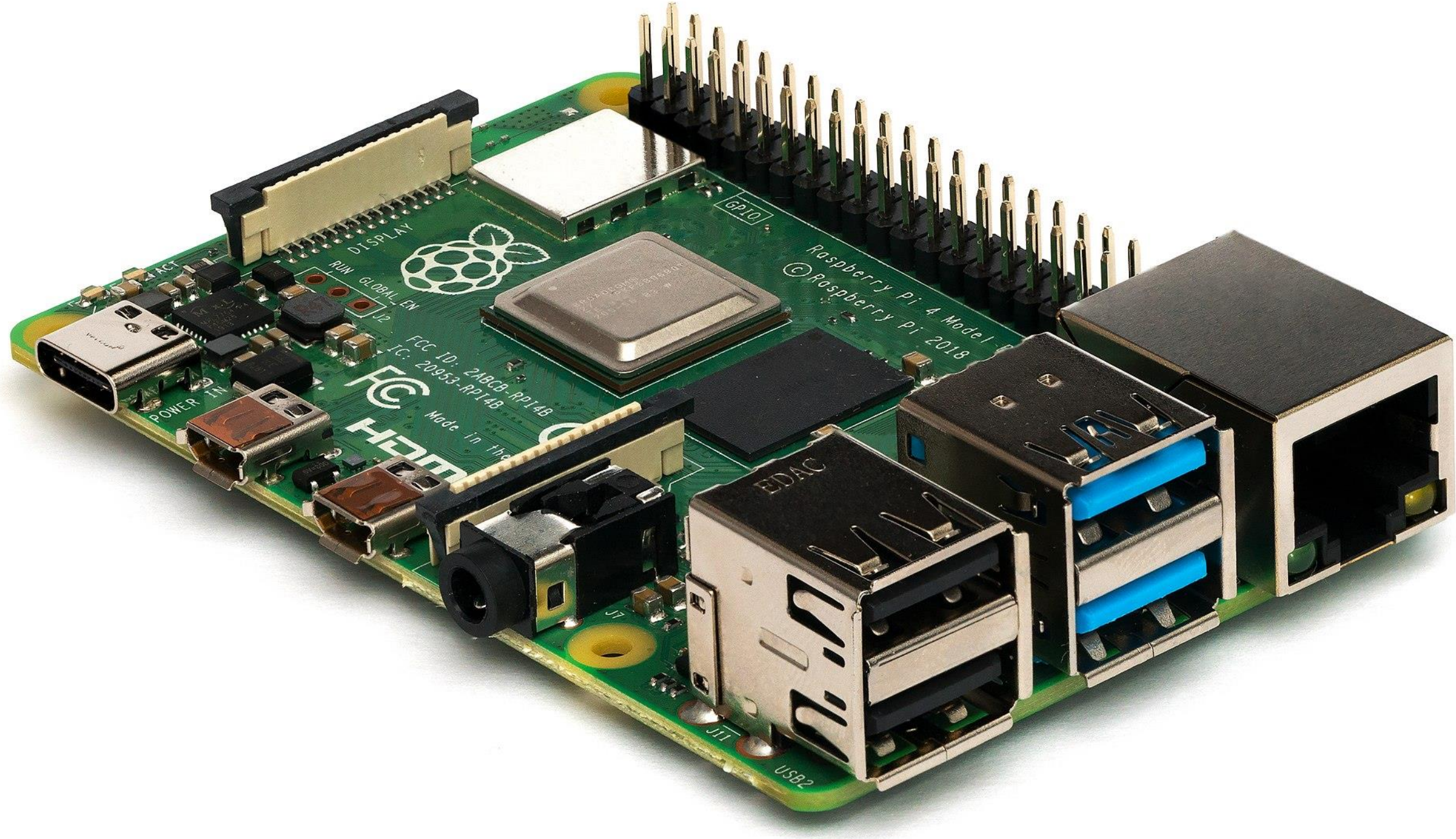

THINKING
FORTH
Programmer's Notepad -
DOER/MAKE FOR THE MICROPROFESSOR:
: NOTHING ;
: DOER CREATE ' NOTHING , DOES> @ >R ;
VARIABLE MARKER
: (MAKE) R> DUP 2+ DUP 2+ SWAP @ 4 + ! @ ?DUP IF >R THEN ;
: MAKE COMPILE (MAKE) HERE MARKER ! 0 , ; IMMEDIATE
: ;AND COMPILE EXIT HERE MARKER @ ! ; IMMEDIATE
: UNDO ' NOTHING <COMPILE> ' ! ;
DOER JOE
: SETUPJOE MAKE JOE ." HI JOE " ;
SETUPJOE
JOE -> HI JOE
FORGET SETUPJOE
: SETUPJOE MAKE JOE ." HELLO JOE "
;AND " NOW! ";
SETUPJOE -> NOW!
JOE -> HELLO JOE
UNDO JOE
JOE ->
Screen # 25
LPB 12/05/83)
(DOER/MAKE 79-Standard MVP FORTH
: NOTHING ;
: DOER CREATE ' NOTHING , DOES> @ >R ;
VARIABLE MARKER
: (MAKE) R> DUP 2+ DUP 2+ SWAP @ 2+ (pfa) !
@ ?DUP IF >R THEN ;
: MAKE STATE @ IF (compiling)
COMPILE (MAKE) HERE MARKER ! 0 ,
ELSE HERE [COMPILE] ' !
[COMPILE]] THEN ; IMMEDIATE
: ;AND COMPILE EXIT HERE MARKER @ ! ; IMMEDIATE
: UNDO ['] NOTHING [COMPILE] ' ! ;
(The code in this screen is in the public domain.)
FORTH
8V TIP++
FORTH-MPF-IP
RESET
BELL
SHIFT
CONTROL

Contemporary educational SBCs

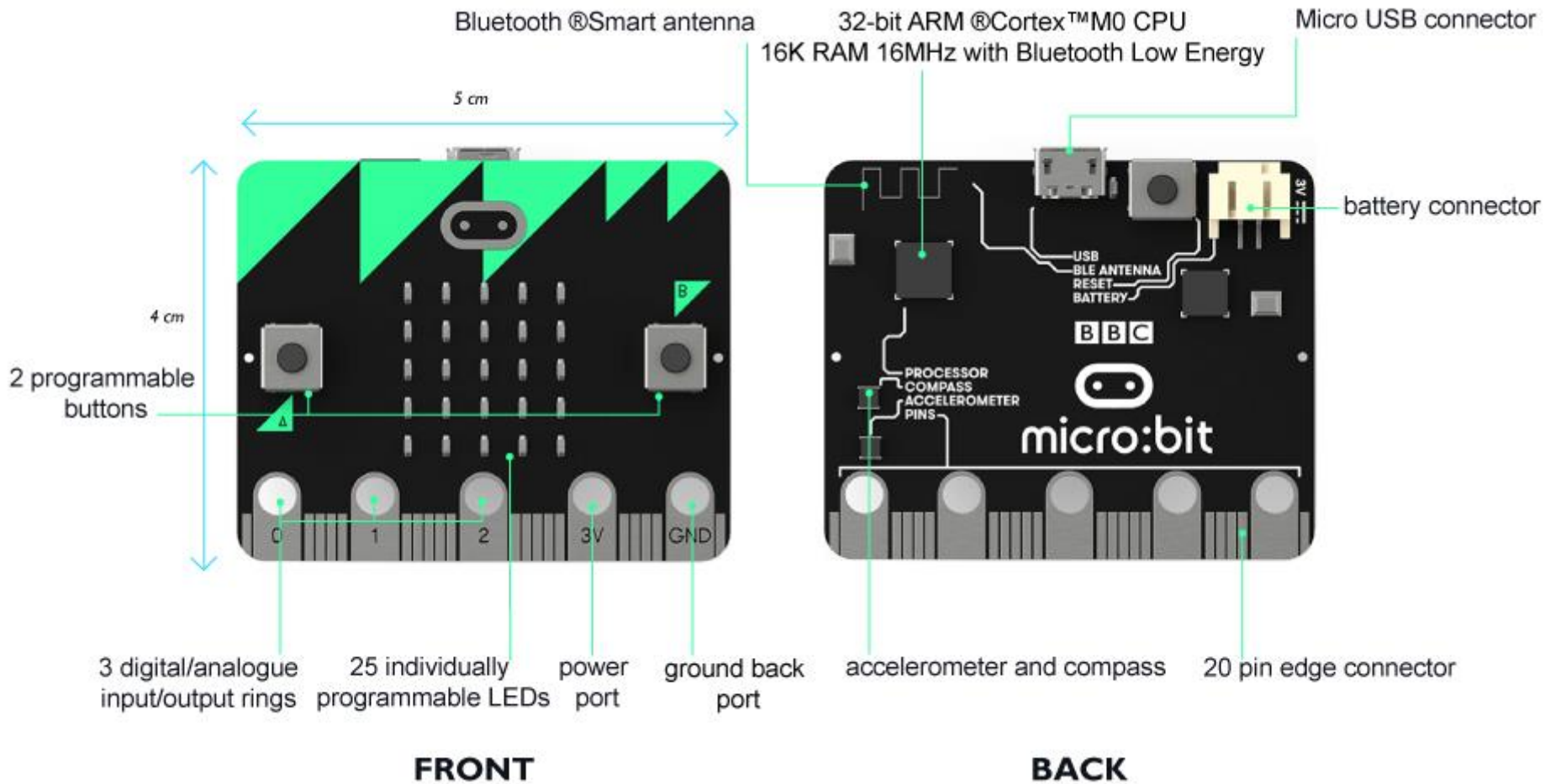
- BBC MicroBit
- Arduino
- Raspberry Pi
- Gaken reimplementation of the Science Fair Trainer



Japanese
Re-Implementation
Of the 1976 Science Fair!


DISPLAY
RUN
GLOBAL_EN
POWER IN
FCC ID: 2ABCB-RPI4B
IC: 20953-RPI4B
Made in the
GPIO
Raspberry Pi 4 Model
Raspberry Pi 2018
EDAC
USB2


BBC
micro:bit
go
0
1
2
3V
GND
BBC micro:bit


Bluetooth ®Smart antenna
32-bit ARM ®Cortex™M0 CPU
16K RAM 16MHz with Bluetooth Low Energy
Micro USB connector
5 cm
4 cm
2 programmable buttons
battery connector
USB
BLE ANTENNA
RESET
BATTERY
PROCESSOR
COMPASS
ACCELEROMETER
PINS
micro:bit
0
1
2
3V
GND
3 digital/analogue input/output rings
25 individually programmable LEDs
power port
ground back port
accelerometer and compass
20 pin edge connector
FRONT
BACK

News

2022

- November
- October
- August
- July
- June
- May
- April
- March
  - 57,000 BBC micro:bits to be donated to primary schools
  - New primary lessons promote unplugged and cross-curricular computing
  - Free teaching resource, Life Below Water
  - Life on Land teaching guide
- February
- January

57,000 BBC micro:bits to be donated to primary schools

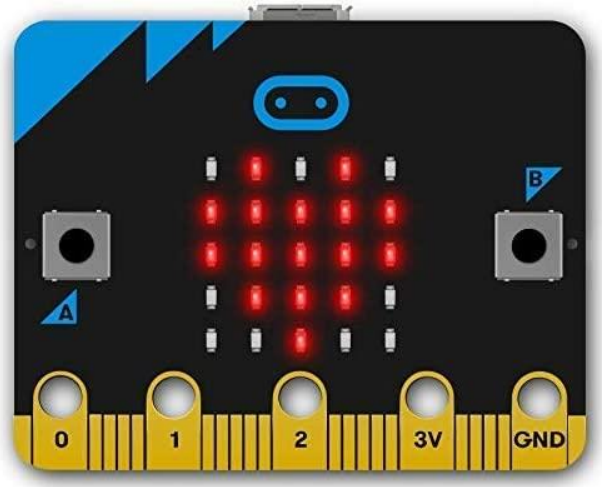
30 Mar 2022

Donation of thousands of micro:bits to support the teaching of digital literacy in UK primary schools

Plans to help even more UK primary school children take their first steps into digital creativity and computing will launch this April. In partnership with Nominet and the Scottish Government, 57,000 micro:bits will be donated to UK primary schools, alongside free comprehensive teaching resources and online continuing professional development courses.









KEYESTUDIO Programming Robot Car Starter Kit for BBC Microbit (No Board&Battery)

Condition: New

Bulk savings:

Buy 1	Buy 2	Buy 3
$37.99/ea	$36.85/ea	$36.09/ea

4 or more for $35.33/ea

Quantity: 1 More than 10 available / 15 sold

Price: **US $37.99/ea**

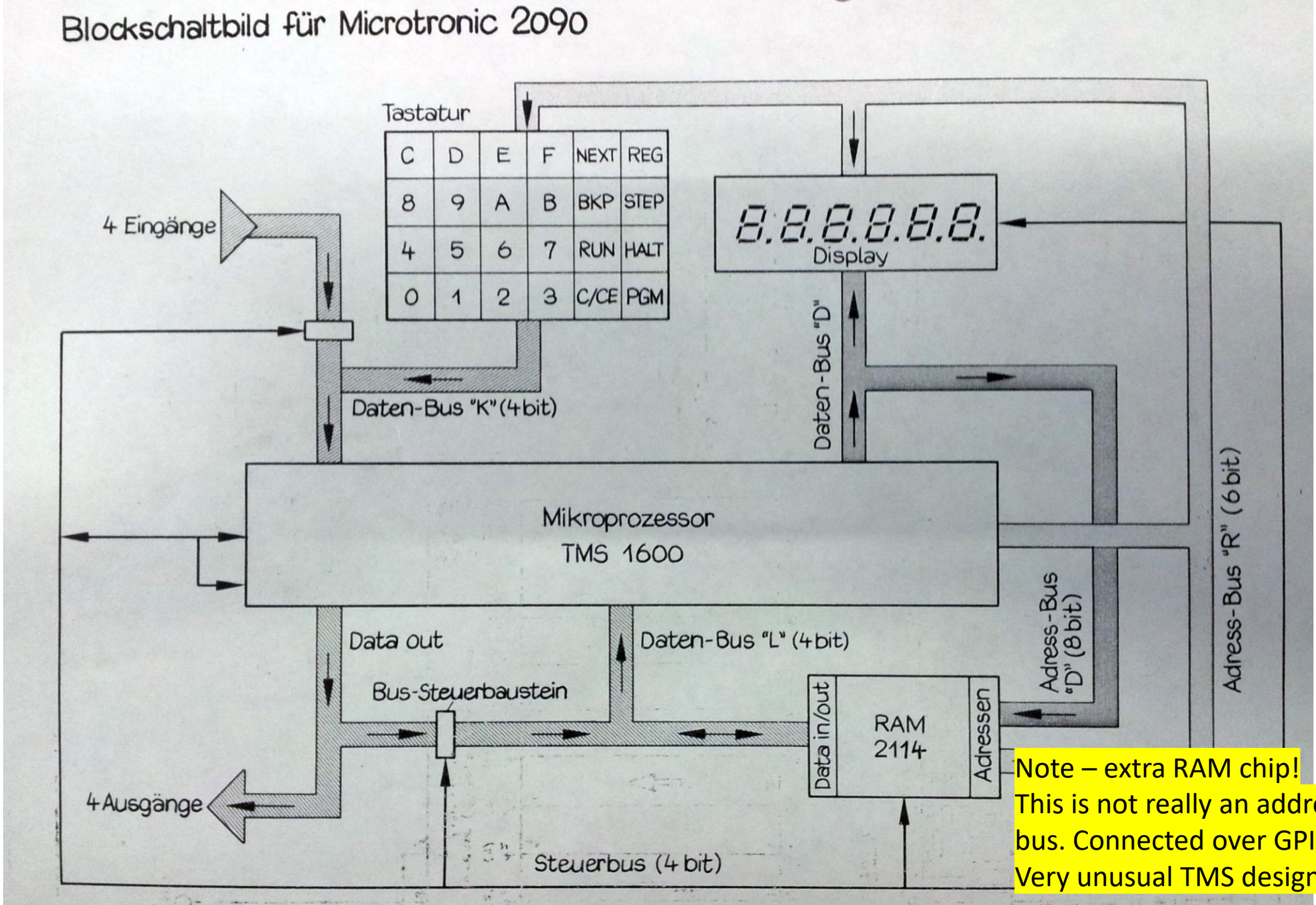
Add to cart

♡ Add to Watchlist

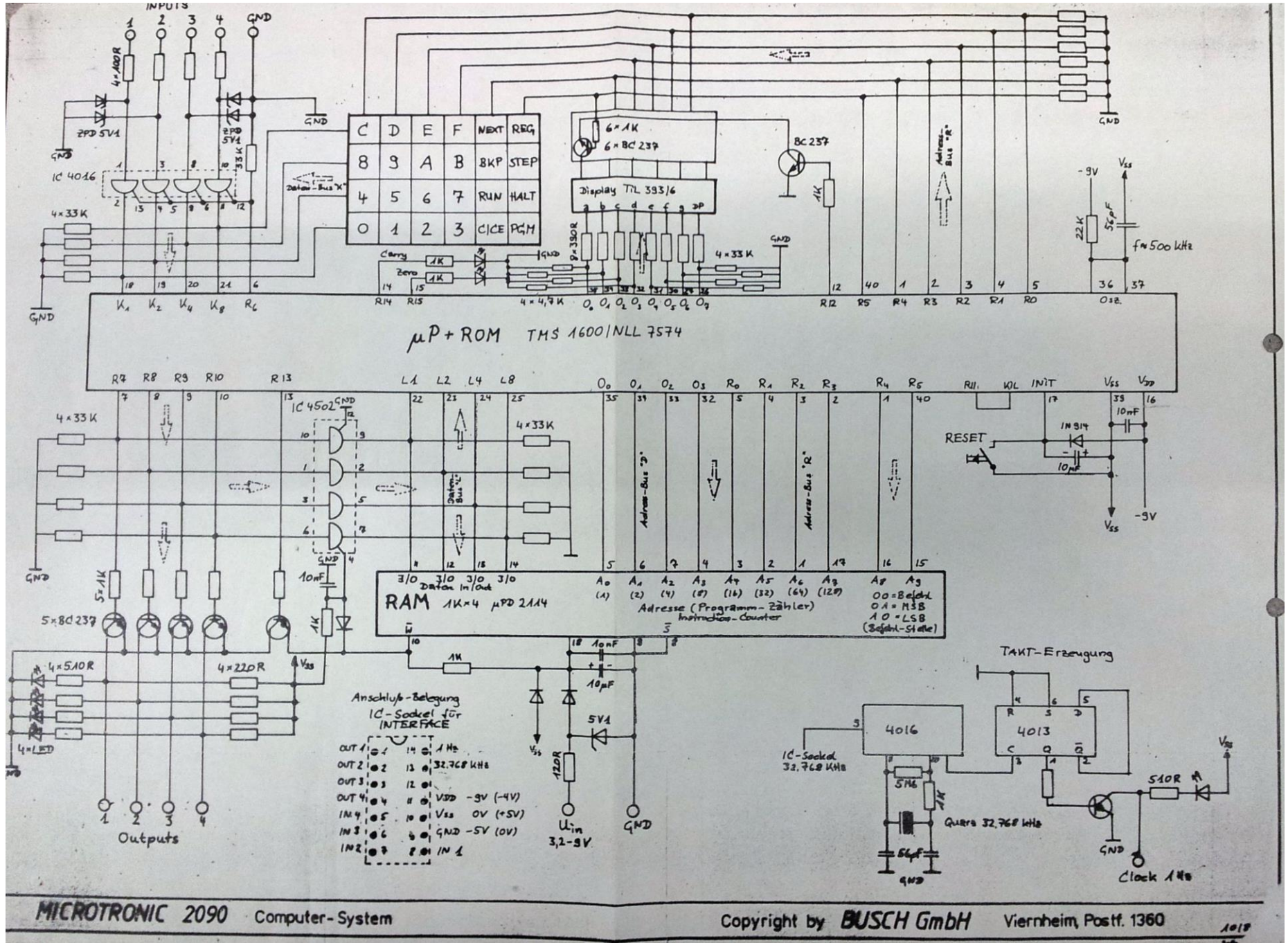
Additional service available

☐ 3-year protection plan from Allstate - $3.34

Returns accepted | 24 watchers



Note – extra RAM chip!
This is not really an address bus. Connected over GPIOs.
Very unusual TMS design.



Copyright by BUSCH GmbH Viernheim, Postf. 1360