

GalWerk

GAL-Editor und Brenner

GAL editor and programmer

Handbuch · Manual

Deutsch ab Seite 2 · English from page 10

Version 1v0 · gal.astro2match.de

created by AnnoBoy 2015–2026

Willkommen bei GalWerk

GAL-Bausteine sind kleine Chips, in die man seine eigene Logik „einbrennt“: aus ein paar Eingängen wird genau das, was man braucht – ein Adress-Decoder, ein Zähler, ein paar Gatter, die sonst drei oder vier TTL-Chips bräuchten. Früher brauchte man dafür einen teuren Programmer am Druckerport und ein DOS-Programm. GalWerk macht beides in einem: Sie schreiben die Logik, GalWerk prüft sie sofort, zeigt Ihnen, was im Chip passiert – und brennt sie mit einem kleinen Brenner am USB, den Sie selbst bauen können.

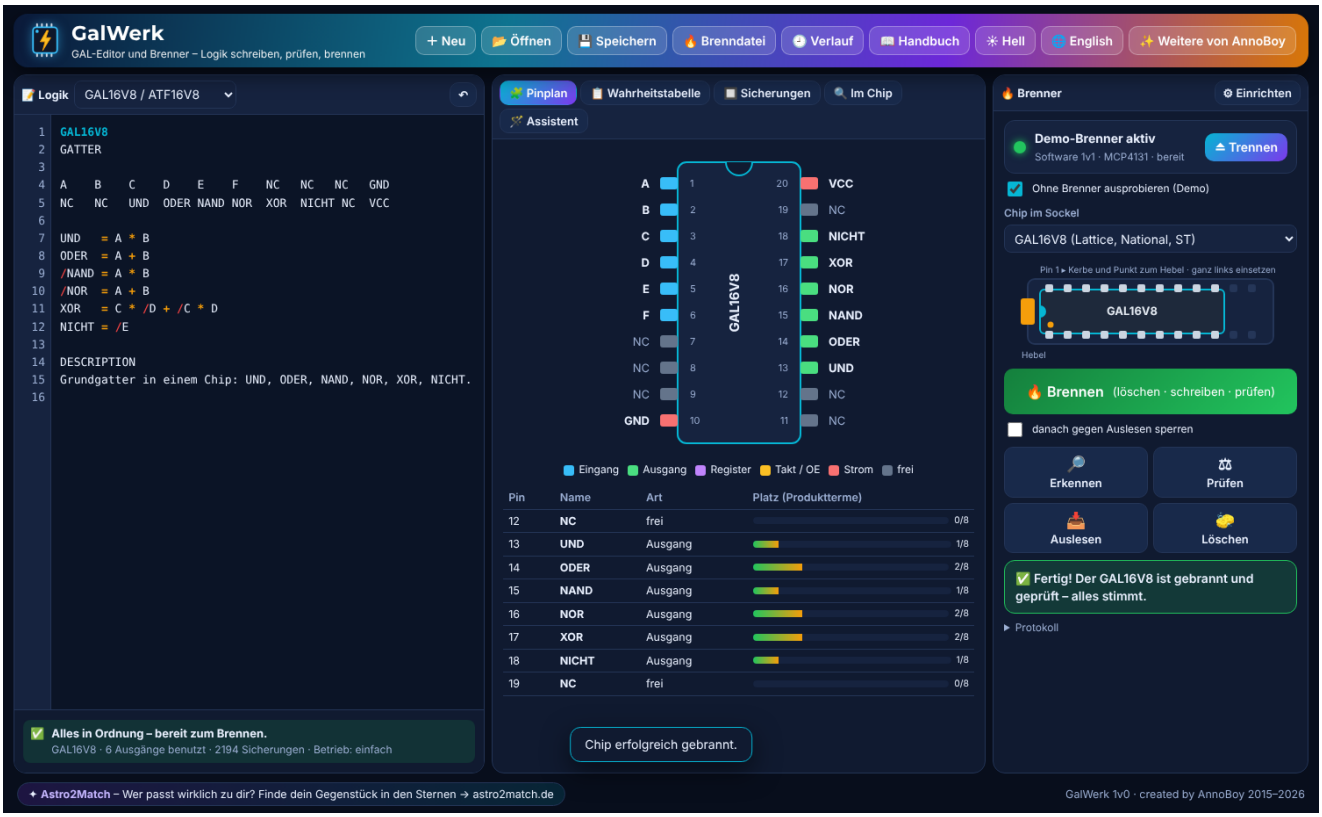
Das kann GalWerk

- Logik schreiben wie gewohnt (galasm/GALasm-Schreibweise) – mit Klammern, XOR und automatischer Vereinfachung
- Fehler werden beim Tippen gefunden und in normalem Deutsch erklärt, mit Tipp zur Lösung
- Pinplan, Wahrheitstabelle und Sicherungsplan – Sie sehen sofort, was der Chip tut
- Assistent: Wahrheitstabelle ankreuzen, GalWerk macht die Gleichung daraus
- Brenndatei (JEDEC, .jed) speichern und öffnen – passend zu anderen Brennern
- Mit dem GalWerk-Brenner: erkennen, löschen, brennen, prüfen, auslesen und gegen Auslesen sperren
- Ausgelesene Chips werden in lesbare Gleichungen zurückverwandelt („Im Chip“)
- Demo-Brenner zum Ausprobieren ganz ohne Hardware
- Deutsch und Englisch, hell und dunkel, am PC, Tablet und Handy

Welche Chips?

GAL16V8 · ATF16V8B/C	20 Beine – der Klassiker, am häufigsten
GAL20V8 · ATF20V8B	24 Beine – wie der 16V8, mehr Eingänge
GAL22V10 · ATF22V10B/C	24 Beine – zehn Ausgänge, Reset und Preset für alle Register
GAL20RA10	24 Beine – jedes Register mit eigenem Takt
GAL18V10 · GAL20XV10	Brennen, Prüfen und Auslesen von fertigen .jed-Dateien

Der Bildschirm



Links die Logik, in der Mitte die Ansichten, rechts der Brenner.

Oben finden Sie Neu (mit fertigen Vorlagen), Öffnen, Speichern, die Brenndatei, den Verlauf, dieses Handbuch, Hell/Dunkel und den Sprachknopf. **Links** schreiben Sie Ihre Logik, darüber wählen Sie den Chip. Unter dem Text steht immer, ob alles in Ordnung ist – oder was nicht stimmt. **In der Mitte** wechseln Sie zwischen Pinplan, Wahrheitstabelle, Sicherungen, „Im Chip“ und dem Assistenten. **Rechts** ist der Brenner.

Am Handy liegen die drei Bereiche untereinander; unten ist eine Leiste zum Umschalten.

Tipp: Ihre Arbeit wird laufend gemerkt. Schließen Sie GalWerk einfach – beim nächsten Mal ist alles wieder da. Ältere Stände finden Sie unter „Verlauf“, auch wenn Sie einen Brennvorgang abgebrochen haben.

Logik schreiben

Eine GalWerk-Datei sieht aus wie bei den bekannten DOS-Programmen GALasm und galasm: erst der Chip, dann ein Name, dann die Namen aller Beine der Reihe nach, dann die Gleichungen.

```
GAL16V8
DECODER

A15 A14 A13 RD WR NC NC NC NC GND
NC  NC  NC  NC NC NC /RAM /ROM /IO VCC

ROM  = /A15 * /A14
RAM  = /A15 * A14 + A15 * /A14 * /A13
IO   = A15 * A14 * A13 * (/RD + /WR)

DESCRIPTION
Adress-Decoder für einen kleinen Rechner
```

*****	UND – beide müssen 1 sein
+	ODER – einer reicht
/ !	NICHT – dreht um
^	XOR – genau einer
( )	Klammern – GalWerk rechnet sie selbst aus
.R	Ausgang mit Speicher (Flipflop, Takt an Pin 1)
.T .E	Ausgang abschaltbar: Y.T = Logik, Y.E = wann er treibt
AR SP	22V10: gemeinsamer Reset und Preset aller Register
NC	Bein ist nicht benutzt

Ein Schrägstrich vor dem Namen in der Bein-Liste heißt: dieses Bein ist „aktiv bei 0“. GalWerk kümmert sich um den Rest. Wählt man den Chip oben über der Logik, wird die erste Zeile mitgeändert.

Tipp: Unter „Neu“ gibt es fertige Vorlagen: Gatter, Adress-Decoder, Zähler und ein 22V10-Beispiel mit Klammern. Am schnellsten lernt man, indem man eine Vorlage ändert und zuschaut, was in der Mitte passiert.

Wenn etwas nicht stimmt

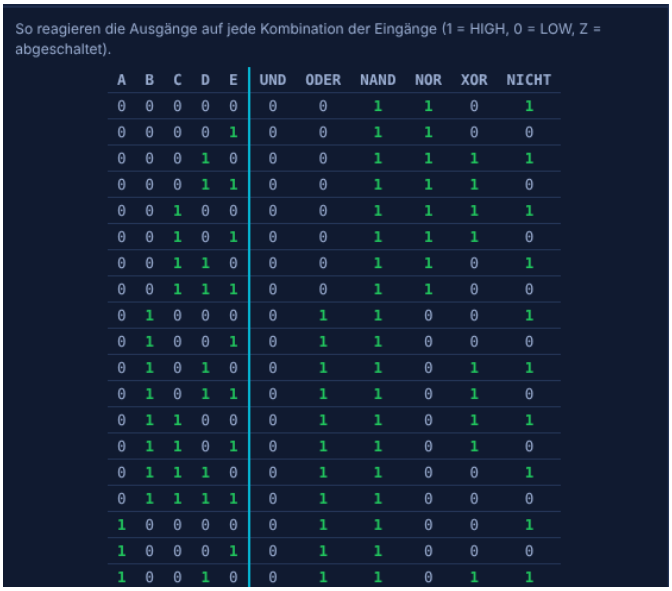
Die betroffene Zeile wird rot markiert und unten steht, was los ist – zum Beispiel „Für den Ausgang Y sind 9 Glieder nötig, es passen aber nur 8 hinein“, mit einem Tipp, wie man es löst (etwa einen anderen Ausgang mit mehr Platz nehmen). Ein Klick auf die Meldung springt in die Zeile.

Die Ansichten in der Mitte

Pinplan

Der Chip von oben, mit allen Namen und Farben (Eingang, Ausgang, Register, Takt, Strom). Darunter sehen Sie für jeden Ausgang, wie viel Platz schon belegt ist.

Wahrheitstabelle


So reagieren die Ausgänge auf jede Kombination der Eingänge (1 = HIGH, 0 = LOW, Z = abgeschaltet).

A	B	C	D	E	UND	ODER	NAND	NOR	XOR	NICHT
0	0	0	0	0	0	0	1	1	0	1
0	0	0	0	1	0	0	1	1	0	0
0	0	0	1	0	0	0	1	1	1	1
0	0	0	1	1	0	0	1	1	1	0
0	0	1	0	0	0	0	1	1	1	1
0	0	1	0	1	0	0	1	1	1	0
0	0	1	1	0	0	0	1	1	0	1
0	0	1	1	1	0	0	1	1	0	0
0	1	0	0	0	0	1	1	0	0	1
0	1	0	0	1	0	1	1	0	0	0
0	1	0	1	0	0	1	1	0	1	1
0	1	0	1	1	0	1	1	0	1	0
0	1	1	0	0	0	1	1	0	1	1
0	1	1	0	1	0	1	1	0	1	0
0	1	1	1	0	0	1	1	0	0	1
0	1	1	1	1	0	1	1	0	0	0
1	0	0	0	0	0	1	1	0	0	1
1	0	0	0	1	0	1	1	0	0	0
1	0	0	1	0	0	1	1	0	1	1

Jede Eingangs-Kombination und was dabei herauskommt.

Sicherungen

So sieht es im Chip wirklich aus: jedes Kästchen ist eine Sicherung. Hell heißt verbunden, dunkel offen, grau eine unbenutzte Zeile. Mit dem Mausrad zoomen, mit der Maus schieben, die Leertaste setzt die Ansicht wieder in die Mitte.

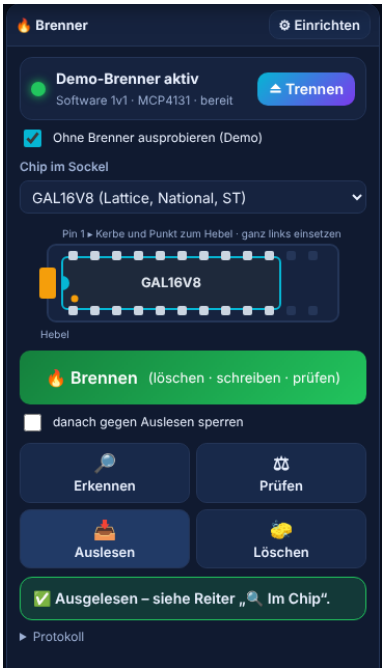
Im Chip

Nach dem Auslesen eines Chips steht hier, was in ihm steckt – als lesbare Gleichungen. Mit einem Klick übernehmen Sie sie in den Editor. So können Sie alte Platinen nachbauen, auch wenn die Original-Datei längst verloren ist (sofern der Chip nicht gesperrt wurde).

Assistent

Sie wissen, was herauskommen soll, aber nicht die Gleichung? Eingänge und Ausgang benennen, in der Tabelle 0, 1 oder „egal“ anklicken – GalWerk rechnet die kürzeste Gleichung aus und setzt sie in Ihre Logik ein.

Brennen



Der Brenner-Bereich.

1. Verbinden

- **Windows-Programm:** Brenner einstecken, Anschluss wählen (meist wird er von selbst gefunden), „Verbinden“.
- **Im Browser:** Chrome oder Edge am PC. Nach „Verbinden“ fragt der Browser, welches Gerät – den Arduino auswählen. Firefox, Safari und Handys können keine USB-Geräte ansprechen; dort geht alles außer dem Brennen.
- **Ohne Brenner:** Haken bei „Ohne Brenner ausprobieren (Demo)“. Alles läuft wie echt, nur ohne Chip.

2. Chip einsetzen

Chip-Typ auswählen. Das kleine Bild zeigt, wie er in den Sockel gehört: Kerbe und Punkt zum Hebel, ganz nach links geschoben. 20-beinige Chips lassen rechts vier Löcher frei. Dann den Hebel umlegen. Zum Schluss den Schalter „Sockel EIN“ einschalten (die LED leuchtet) – GalWerk sagt Bescheid, falls er falsch steht.

Tipp: „Erkennen“ liest die Kennung des Chips und stellt den Typ von selbst ein.

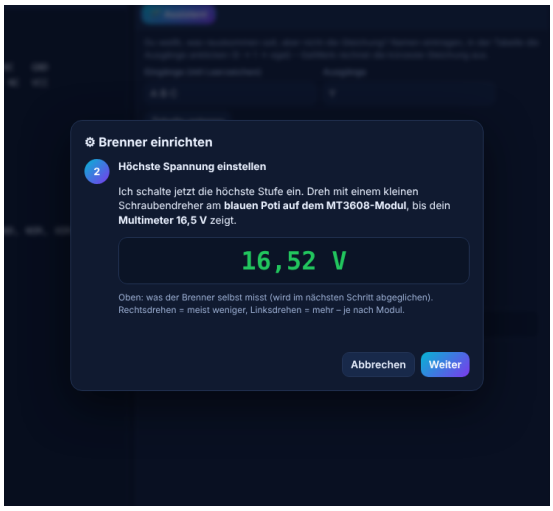
3. Brennen

Der große grüne Knopf macht alles in einem Rutsch: löschen, schreiben, prüfen. Ein Balken zeigt den Fortschritt, „Abbrechen“ hält sofort an und schaltet die Brennspannung ab. Am Ende steht klar da, ob alles stimmt. Wer möchte, setzt vorher den Haken „danach gegen Auslesen sperren“ – dann kann niemand den Inhalt mehr lesen (der Chip lässt sich aber wieder löschen und neu brennen).

Erkennen	Kennung lesen, Typ einstellen
Prüfen	Chip mit der aktuellen Logik vergleichen
Auslesen	Inhalt holen – siehe Reiter „Im Chip“
Löschen	Chip leer machen

Den Brenner einrichten

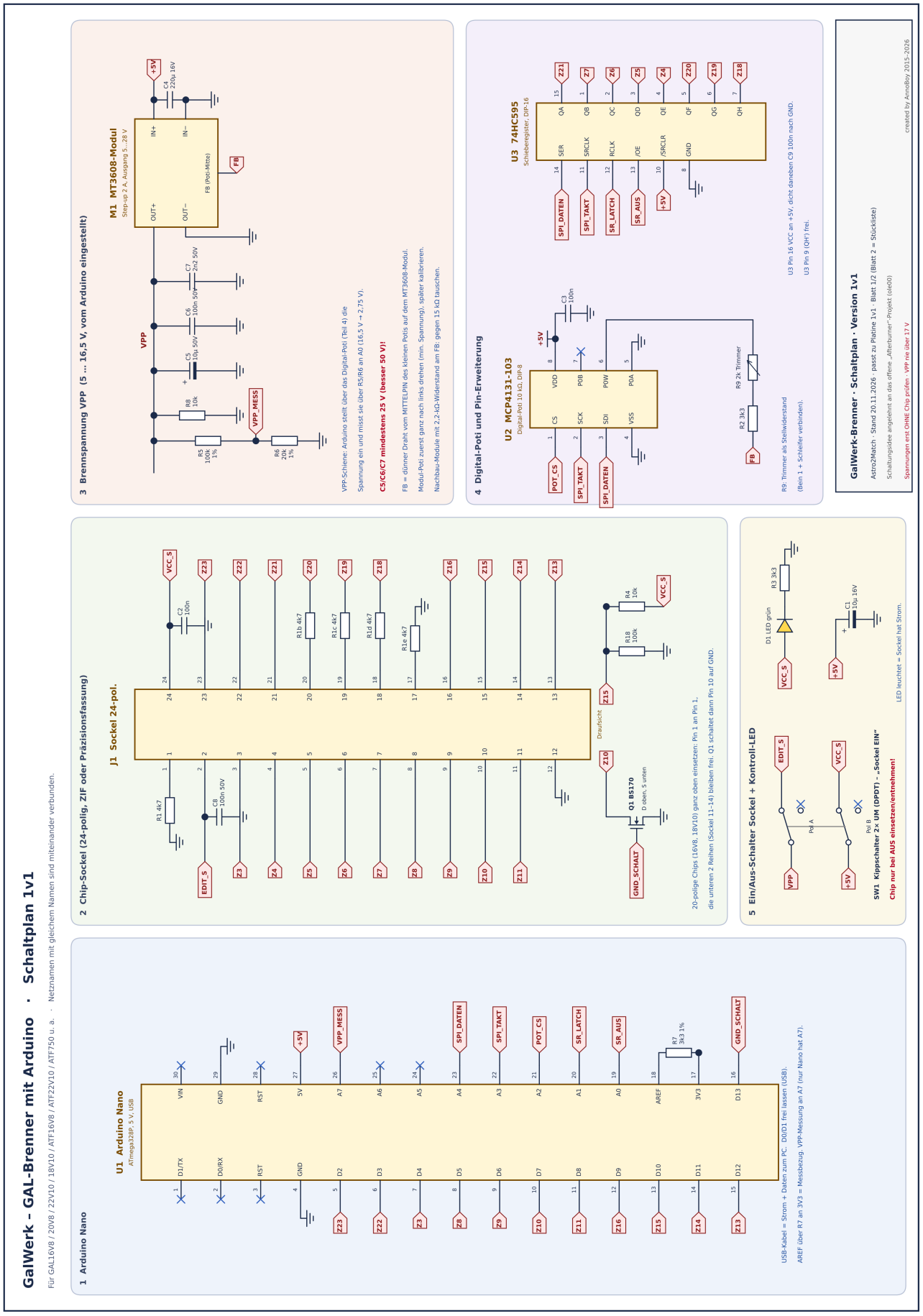
Damit die Brennspannung genau stimmt, misst GalWerk sie einmal nach. Oben rechts beim Brenner auf „Einrichten“ – ein kleiner Assistent führt in fünf Schritten durch: Schalter aus, Spannung messen, Wert vom Messgerät eintragen, fertig. Das Ergebnis bleibt im Brenner gespeichert.



Der Einricht-Assistent.

Den Brenner selbst bauen

Der Brenner besteht aus einem Arduino Nano, einem 24-poligen Hebelsockel, einem kleinen Spannungswandler und einer Handvoll Bauteilen. Schaltplan (nächste Seite), Stückliste und Platinenvorlage (einseitig mit Drahtbrücken oder zweiseitig) liegen dem Windows-Paket bei. Die Software für den Arduino ist ebenfalls dabei und wird einmal mit der Arduino-IDE aufgespielt.



Schaltplan des GalWerk-Brenners (zum Lesen das Blatt drehen).

Wenn es hakt

„Kein Anschluss gefunden“	USB-Kabel prüfen (manche Kabel laden nur). Beim Nano mit CH340 ggf. den Treiber installieren.
„Anschluss belegt“	Arduino-IDE oder serieller Monitor ist noch offen – schließen.
Schalter „Sockel EIN“	Zum Brennen an (LED leuchtet), zum Einsetzen und beim Einrichten aus.
„Spannung stimmt nicht“	Netzteil/USB prüfen, dann „Einrichten“ durchlaufen.
„Kein Chip erkannt“	Chip richtig herum und ganz links? Hebel zu? Typ von Hand wählen.
Prüfen findet Fehler	Einmal löschen und neu brennen. Bleibt es, ist der Chip oft verbraucht.
Im Browser kein „Verbinden“	Chrome oder Edge am PC nehmen – oder das Windows-Programm.

Tastatur

Strg+S / Ctrl+S	Logik speichern
Strg+O / Ctrl+O	Öffnen
Strg+Z / Ctrl+Z	Rückgängig
Esc	Laufende Aktion abbrechen / Fenster schließen
Leertaste	Sicherungs-Ansicht wieder mittig (nicht beim Tippen)

Dateien können Sie einfach ins Fenster ziehen: .pld für Logik, .jed für Brenndateien.

Testen und Freischalten

GalWerk können Sie 7 Tage ohne Anmeldung ausprobieren. Danach melden Sie sich mit Ihrer E-Mail-Adresse an. Ihre Logik und Ihr Verlauf bleiben dabei immer erhalten.

Viel Freude beim Basteln!

Welcome to GalWerk

GAL devices are small chips into which you "burn" your own logic: a handful of inputs become exactly what you need – an address decoder, a counter, a few gates that would otherwise take three or four TTL chips. In the old days this needed an expensive programmer on the printer port and a DOS program. GalWerk does both in one: you write the logic, GalWerk checks it immediately, shows you what happens inside the chip – and burns it with a small USB programmer you can build yourself.

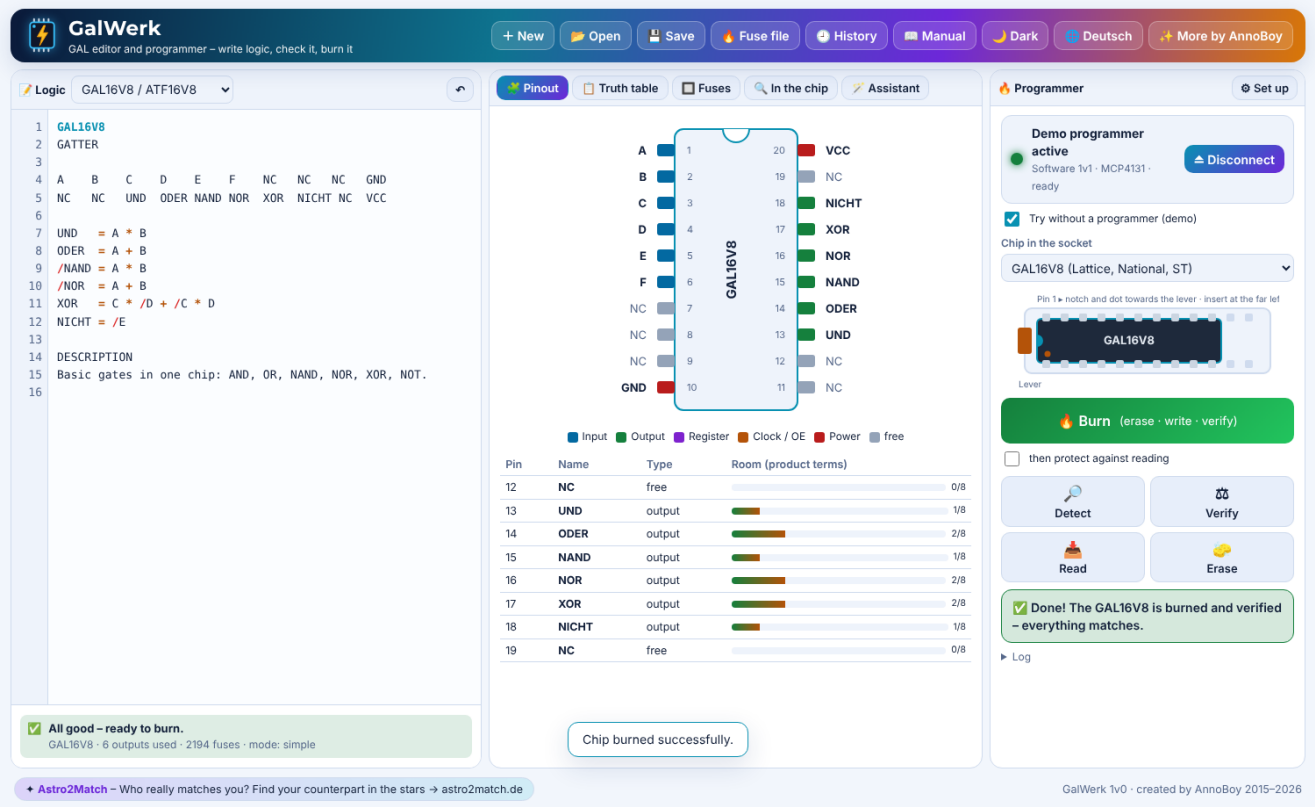
What GalWerk can do

- Write logic as usual (galasm/GALasm syntax) – with brackets, XOR and automatic simplification
- Mistakes are found while you type and explained in plain words, with a hint how to fix them
- Pin plan, truth table and fuse map – you see right away what the chip does
- Wizard: tick a truth table, GalWerk turns it into an equation
- Save and open fuse files (JEDEC, .jed) – compatible with other programmers
- With the GalWerk programmer: identify, erase, burn, verify, read back and protect against reading
- Chips you read back are turned into readable equations again ("In the chip")
- Demo programmer to try everything without hardware
- German and English, light and dark, on PC, tablet and phone

Which chips?

GAL16V8 · ATF16V8B/C	20 pins – the classic, most common
GAL20V8 · ATF20V8B	24 pins – like the 16V8, more inputs
GAL22V10 · ATF22V10B/C	24 pins – ten outputs, reset and preset for all registers
GAL20RA10	24 pins – every register with its own clock
GAL18V10 · GAL20XV10	Burn, verify and read back ready-made .jed files

The screen



Logic on the left, views in the middle, programmer on the right.

At the top you find New (with ready-made templates), Open, Save, the fuse file, the history, this manual, light/dark and the language button. **On the left** you write your logic and choose the chip above it. Below the text it always says whether everything is fine – or what is wrong. **In the middle** you switch between pin plan, truth table, fuses, "In the chip" and the wizard. **On the right** is the programmer.

On a phone the three areas are below each other; a bar at the bottom switches between them.

Tip: Your work is saved all the time. Simply close GalWerk – next time everything is back. Older states are under "History", even if you cancelled a burn.

Writing logic

A GalWerk file looks like the well-known DOS programs GALasm and galasm: first the chip, then a name, then the names of all pins in order, then the equations.

```
GAL16V8
DECODER

A15 A14 A13 RD WR NC NC NC NC GND
NC  NC  NC  NC NC NC /RAM /ROM /IO VCC

ROM  = /A15 * /A14
RAM  = /A15 * A14 + A15 * /A14 * /A13
IO   = A15 * A14 * A13 * (/RD + /WR)

DESCRIPTION
Address decoder for a small computer
```

*****	AND – both must be 1
+	OR – one is enough
/ !	NOT – inverts
^	XOR – exactly one
( )	Brackets – GalWerk expands them itself
.R	Registered output (flip-flop, clock on pin 1)
.T .E	Output can be switched off: Y.T = logic, Y.E = when it drives
AR SP	22V10: common reset and preset of all registers
NC	pin not used

A slash in front of a name in the pin list means: this pin is "active low". GalWerk takes care of the rest. If you choose the chip above the logic, the first line changes with it.

Tip: "New" offers ready-made templates: gates, address decoder, counter and a 22V10 example with brackets. The quickest way to learn is to change a template and watch what happens in the middle.

When something is wrong

The line concerned is marked red and below it says what is wrong – for example "Output Y needs 9 terms, but only 8 fit", with a hint how to fix it (e.g. use another output with more room). Clicking the message jumps to the line.

The views in the middle

Pin plan

The chip seen from above with all names and colours (input, output, register, clock, power). Below you see for each output how much room is already used.

Truth table

A	B	C	D	E	UND	ODER	NAND	NOR	XOR	NICHT
0	0	0	0	0	0	0	1	1	0	1
0	0	0	0	1	0	0	1	1	0	0
0	0	0	1	0	0	0	1	1	1	1
0	0	0	1	1	0	0	1	1	1	0
0	0	1	0	0	0	0	1	1	1	1
0	0	1	0	1	0	0	1	1	1	0
0	0	1	1	0	0	0	1	1	0	1
0	0	1	1	1	0	0	1	1	0	0
0	1	0	0	0	0	1	1	0	0	1
0	1	0	0	1	0	1	1	0	0	0
0	1	0	1	0	0	1	1	0	1	1
0	1	0	1	1	0	1	1	0	1	0
0	1	1	0	0	0	1	1	0	1	1
0	1	1	0	1	0	1	1	0	1	0
0	1	1	1	0	0	1	1	0	0	1
0	1	1	1	1	0	1	1	0	0	0
1	0	0	0	0	0	1	1	0	0	1
1	0	0	0	1	0	1	1	0	0	0
1	0	0	1	0	0	1	1	0	1	1
1	0	0	1	1	0	1	1	0	1	0
1	0	1	0	0	0	1	1	0	1	1

Every input combination and what comes out.

Fuses

This is what really happens inside the chip: every square is a fuse. Bright means connected, dark open, grey an unused row. Zoom with the mouse wheel, drag with the mouse, the space bar centres the view again.

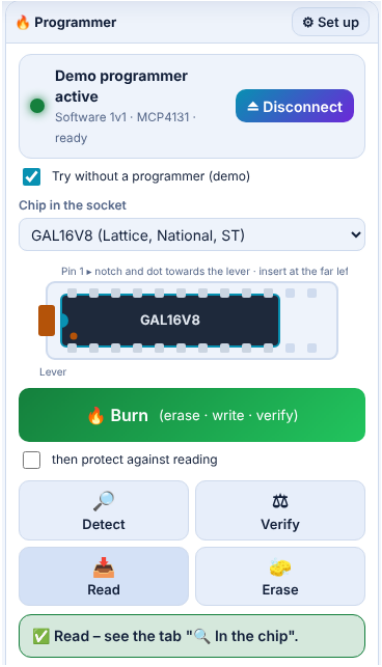
In the chip

After reading a chip this shows what is inside – as readable equations. One click copies them into the editor. This way you can rebuild old boards even if the original file was lost long ago (as long as the chip was not protected).

Wizard

You know what should come out, but not the equation? Name inputs and output, click 0, 1 or "don't care" in the table – GalWerk works out the shortest equation and puts it into your logic.

Burning



The programmer area.

1. Connect

- **Windows program:** plug in the programmer, choose the port (usually it is found by itself), "Connect".
- **In the browser:** Chrome or Edge on a PC. After "Connect" the browser asks which device – choose the Arduino. Firefox, Safari and phones cannot talk to USB devices; everything except burning works there.
- **Without programmer:** tick "Try without a programmer (demo)". Everything runs as if real, just without a chip.

2. Insert the chip

Choose the chip type. The small picture shows how it goes into the socket: notch and dot towards the lever, pushed fully to the left. 20-pin chips leave four holes free on the right. Then close the lever. Finally turn the "socket ON" switch on (the LED lights up) – GalWerk tells you if it is wrong.

Tip: "Detect" reads the chip's ID and sets the type by itself.

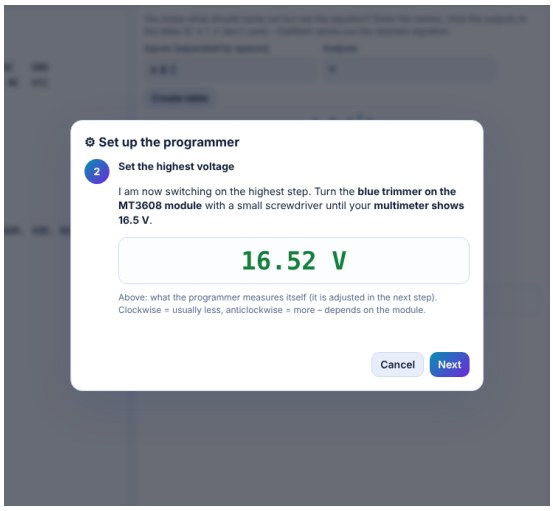
3. Burn

The big green button does everything in one go: erase, write, verify. A bar shows the progress, "Cancel" stops at once and switches off the programming voltage. At the end it clearly says whether everything matches. If you like, tick "then protect against reading" first – then nobody can read the contents any more (the chip can still be erased and burned again).

Detect	read ID, set type
Verify	compare chip with current logic
Read	fetch contents – see "In the chip" tab
Erase	make the chip blank

Setting up the programmer

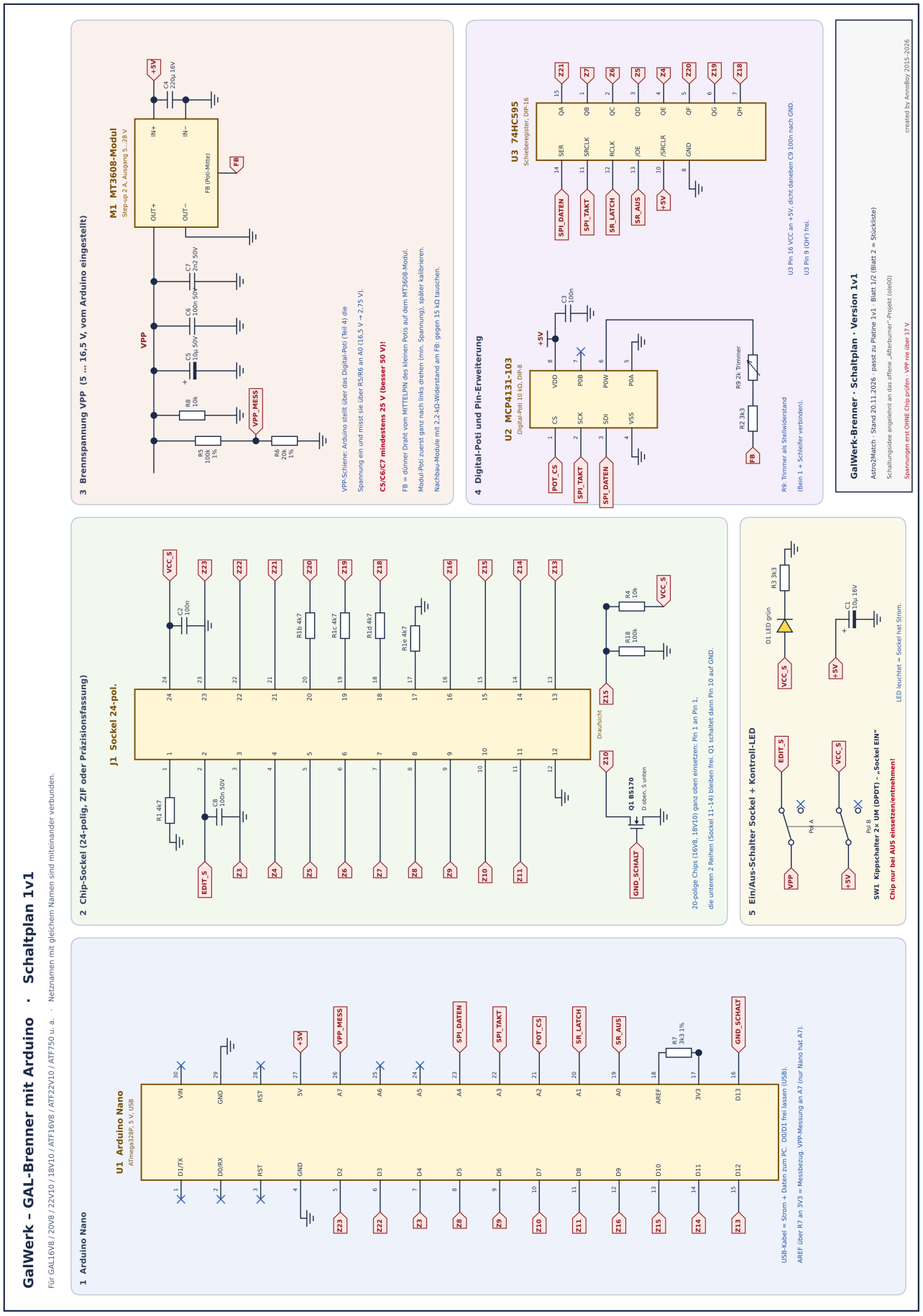
So that the programming voltage is exactly right, GalWerk measures it once. Click "Set up" at the top right of the programmer – a small wizard guides you in five steps: switch off, measure the voltage, enter the value from your multimeter, done. The result stays stored in the programmer.



The set-up wizard.

Building the programmer yourself

The programmer consists of an Arduino Nano, a 24-pin lever socket, a small voltage converter and a handful of parts. Schematic (next page), parts list and board layout (single-sided with wire links or double-sided) come with the Windows package. The Arduino software is included too and is uploaded once with the Arduino IDE.



Schematic of the GalWerk programmer (turn the page to read).

If something goes wrong

"No port found"	Check the USB cable (some cables only charge). For a Nano with CH340 install its driver if needed.
"Port in use"	Arduino IDE or serial monitor is still open – close it.
"socket ON" switch	On for burning (LED lit), off while inserting the chip and during set-up.
"Voltage is wrong"	Check power/USB, then run "Set up".
"No chip detected"	Chip the right way round and fully left? Lever closed? Choose type by hand.
Verify finds errors	Erase once and burn again. If it stays, the chip is often worn out.
No "Connect" in the browser	Use Chrome or Edge on a PC – or the Windows program.

Keyboard

Strg+S / Ctrl+S	save logic
Strg+O / Ctrl+O	open
Strg+Z / Ctrl+Z	undo
Esc	cancel running action / close window
Space bar	centre the fuse view again (not while typing)

You can simply drag files into the window: .pld for logic, .jed for fuse files.

Trial and activation

You can try GalWerk for 7 days without registering. After that you register with your e-mail address. Your logic and history are always kept.

Have fun building!