

ROM-Images des PC-1403 auslesen

Der Emulator braucht fünf Dateien (MAME-Namen):

Datei	Grösse	Inhalt	CRC32 (MAME-Romset)
introm.bin	8192	internes CPU-ROM	588c500b
extrom08.bin	16384	Bank 1 (Registerwert 8)	1fa65140
extrom09.bin	16384	Bank 2 (Registerwert 9)	4a7da6ab
extrom0a.bin	16384	Bank 3 (Registerwert 10)	9925174f
extrom0b.bin	16384	Bank 4 (Registerwert 11)	fa5df9ec

Andere ROM-Revisionen (Schulze erwähnt zwei Versionen) können abweichende Prüfsummen haben; ein brauchbarer Plausibilitätstest ist die Signatur &55 an Offset &3FFE der Bänke 2 und 4, die der Boot-Code des internen ROMs prüft.

Status

- introm.bin liegt vor (Dump vom Dezember 2025, Prüfsumme stimmt exakt).
- Die vier externen Bänke fehlen. Der Dump vom Dezember 2025 („ROM ext“) war ein Adressraum-Dump mit Übertragungsverlusten und enthält nur eine (defekte) Bank – siehe sharp-pc1403-analyse.md.

Auslesen mit Arduino UNO (Verfahren von Walter Fischer, cavefisher.at, 2026)

Quelle und vollständige Beschreibung inkl. Steckbrett-Foto und Arduino-Sketch: <http://www.cavefisher.at> (Abschnitt „PC-1403“). Das Verfahren überträgt jedes Byte als zwei Nibbles parallel über EXT1–EXT4 mit Handshake über Fo3 (Sharp → Arduino: „Daten gültig“) und IN1 (Arduino → Sharp: „bereit“). Dadurch gehen – anders als beim alten Dump – keine Bytes verloren.

Verdrahtung (11-polige Buchsenleiste ↔ Arduino UNO)

Sharp Pin	Signal	Arduino	Bemerkung
3	GND	GND	
11	EXT1	D2	Pull-down 10 kΩ nach GND
10	EXT2	D3	Pull-down 10 kΩ
5	EXT3	D4	Pull-down 10 kΩ
8	EXT4	D5	Pull-down 10 kΩ
4	Fo3	D6	Pull-down 10 kΩ
9	IN1	D7	Ausgang des Arduino

Ablauf

1. Arduino-Sketch von der Seite laden (Ausgabe mit 115200 Baud: Kopfzeile `Adr. ;Bk1;Bk2;Bk3;Bk4`, dann eine Zeile pro Adresse 16384...32767 mit den vier Bankwerten).
2. Auf dem PC-1403 das BASIC-Programm der Seite eingeben (es legt ein kurzes Maschinenprogramm bei `&F100` ab, das nacheinander die Bankregisterwerte 0, 1, 10, 11 setzt, das Byte am aktuellen Offset liest und über EXT1–4 ausgibt). Mit RUN einmal die DATA-Zeilen poken (ca. 14 s), danach mit DEF A die Übertragung starten (ca. 38 s, endet mit Piepton).
3. Seriellen Monitor als Textdatei speichern (z. B. `pc1403_extrom.csv`).
4. Konvertieren und prüfen:

```
python tools/csv2rom.py pc1403_extrom.csv roms/
```

Das Skript schreibt `extrom08.bin ... extrom0b.bin` sowie `extrom.bin` (64 KB) und meldet die CRC32-Werte. Fehlende Adressen werden gemeldet – dann Übertragung wiederholen.

5. Dateien in den Emulator laden: entweder über den Dialog „ROMs...“ (werden im Browser gespeichert), oder nach `public/roms/` legen (werden beim Start automatisch geladen; `roms/` ist für den Headless-Runner und ist wie `public/roms/*.bin` per `.gitignore` vom Commit ausgeschlossen).

Internes ROM

Das interne ROM liegt bereits vor. Falls es neu ausgelesen werden soll: cavefischer.at, „Auslesen des internen ROM's“ (`IntROM.html`) beschreibt das Verfahren mit dem DATA-Befehl (das interne ROM ist nur so lesbar). Eine „`addr;value;mnemonic`“-CSV wandelt `tools/csv2rom.py` ebenfalls um.

Rechtliches

Die Firmware ist urheberrechtlich geschützt (Sharp Corporation). ROM-Images gehören nicht in das Repository und werden vom Emulator ausschliesslich lokal (IndexedDB des Browsers bzw. lokaler Ordner) verwendet.