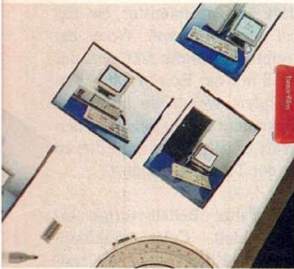


Super-6502

Wer geglaubt hat, daß mit den Prozessoren der 65XX-Reihe seit dem Ableben des Apple-II nichts mehr los ist, irrt sich gewaltig. Die Japaner haben einen Super-6502 entwickelt, der im EMUF-734 seine Muskeln zeigt.

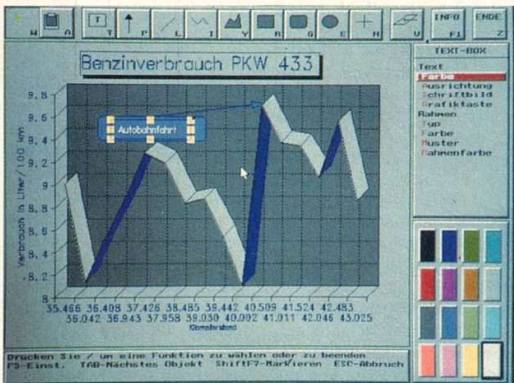
Seite 104



Durchleuchtet

Jeder preist die EISA-486er als ideale Fileserver für Netzwerke an, mc hat sie unter Novell Netware im Netz getestet. Ob mit dem Etikett EISA schon alles klar ist, erfahren Sie auf

Seite 44



Neuling gegen Klassiker

Quattro Pro 3.0 oder Lotus 1-2-3: Welche Tabellenkalkulation bringt Zahlenkolonnen besser in Form? Anhand mehrerer Beispiele aus der Praxis sehen Sie die wichtigsten Unterschiede zwischen den beiden Programmen.

Seite 184

MS-EXTRA

| | |
|--|-----|
| Kompatibles Zweiergespann | 138 |
| OS/2 und DOS auf einer Platte | |
| Bibliothek für Windows | 142 |
| mc-Windows-DLL für Visual Basic und Turbo Pascal | |

PROGRAMMIEREN

| | |
|------------------------------------|-----|
| Welcher Typ ist der Co ...? | 152 |
| CPU und Coprozessor identifizieren | |

TRICKKISTE

| | |
|--|-----|
| Programmieren professionell: Quick-Format | 162 |
| Drucker bereit: Turbo-Pascal-Unit | 166 |
| Europäisches Dilemma: Umlaute in Winword | 166 |
| Grau ist schick: Farben in Grauwerte umwandeln | 167 |

mc-PAPERDISK

| | |
|---|-----|
| Futter für Scanner | 168 |
| Mit der mc-Paperdisk Software direkt einlesen | |

EXKLUSIV

REPORT

| | |
|-------------------------------------|-----|
| Bunter die Blätter nie waren | 174 |
| Neue Farbdrucktechnik von Tektronix | |
| Wahre Typen für Home-DTP | 180 |
| Truetype als Postscript-Ersatz | |

TESTLABOR

| | |
|--|-----|
| Im Netz durchleuchtet | 44 |
| Fünf EISA-486er im Fileserver-Test | |
| Hochdruck für Windows | 60 |
| Supergrafikkarte von ATI | |
| Knapp kalkuliert | 184 |
| Quattro Pro 3.0 contra Lotus 1-2-3 2.3 | |
| Spaß mit C | 194 |
| Quick für Windows | |

TAIWAN-SPECIAL

| | |
|---|-----|
| Taiwan überall: Neue Produkte aus Fernost | 213 |
|---|-----|

BLITZLICHT

| | |
|------------------------------------|-----|
| Postscript-Tuning: Pacific Page XL | 226 |
|------------------------------------|-----|

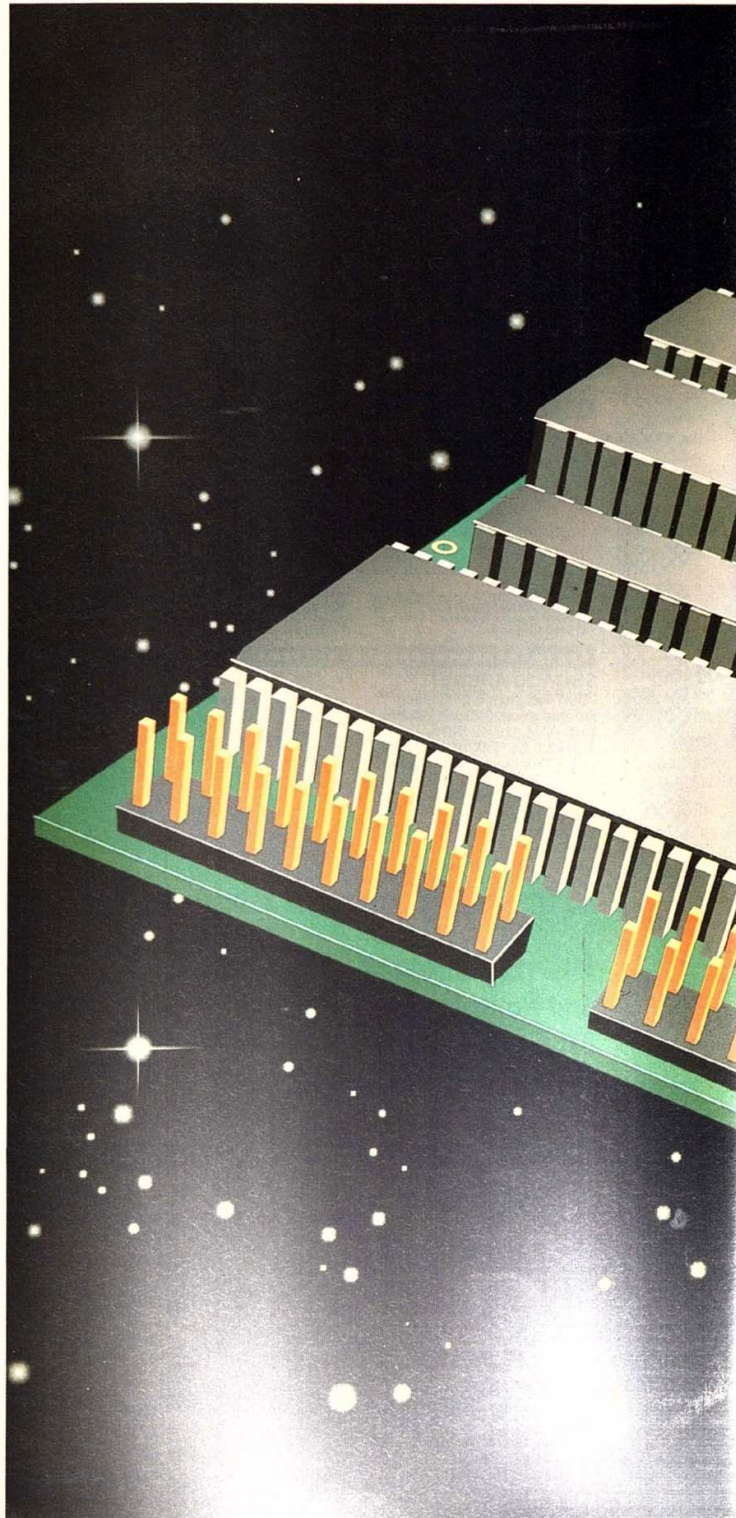
STÄNDIGE RUBRIKEN

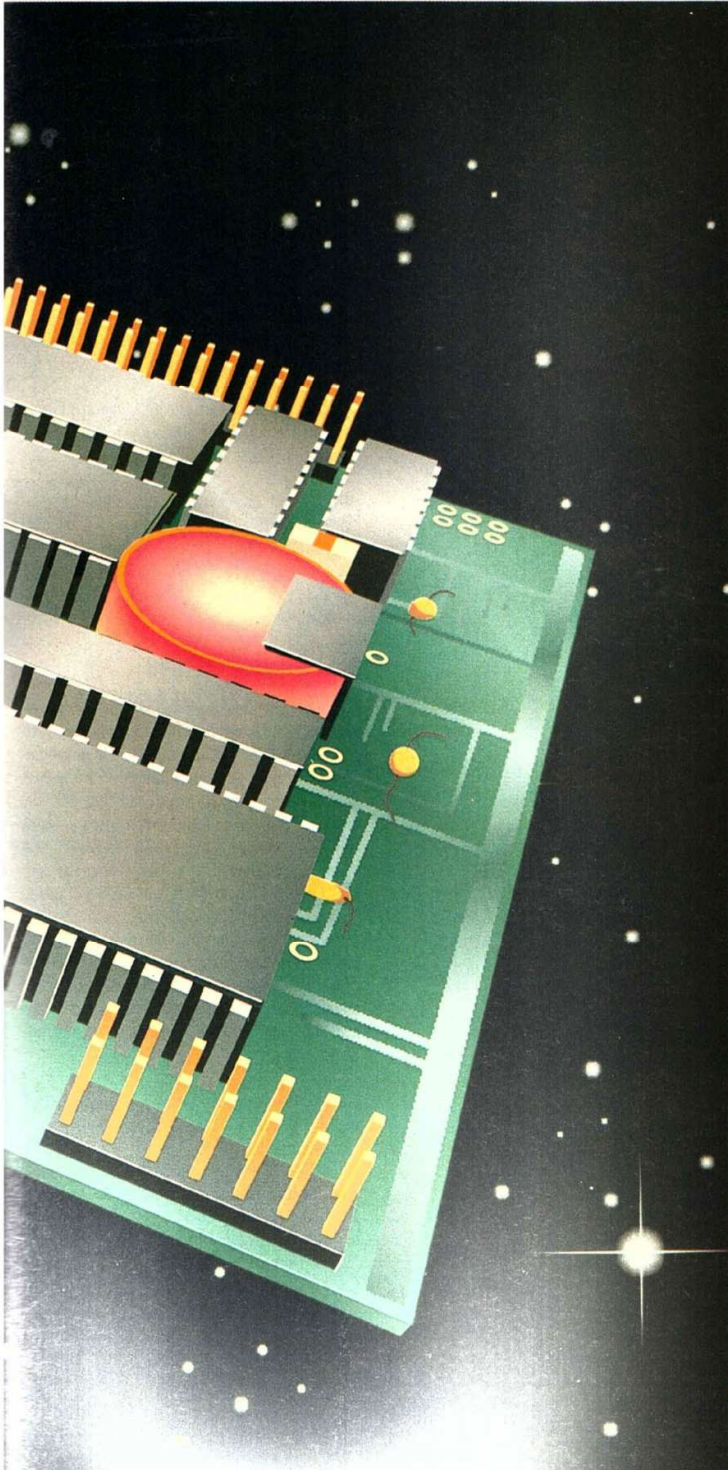
| | | | |
|-----------|-----|-----------------------|-----|
| Editorial | 3 | Briefe | 225 |
| Kolumne | 6 | Bücher | 228 |
| Online | 34 | Impressum | 230 |
| Das noch | 40 | Inserentenverzeichnis | 233 |
| Produkte | 198 | Vorschau | 234 |

Blick zurück nach vorn

*Ein würdiger 65er-
Nachfolger bietet alles,
was EMUF-Fans noch
vermißten*

Das hier vorgestellte Projekt ist die Idee einiger schon etwas älterer Freunde der 6502-CPU und trug während der Entwicklungszeit den (heimlichen) Projektnamen „Der Enkel“. Nun gibt es seit Jahren im politischen Bereich eine Menge von Enkeln, sei es von Brandt, Adenauer, Bismarck oder sonstwem, aber damit hatte es nichts zu tun. Schon allein deswegen, weil im menschlichen Leben und Tun die Enkel selten besser als die Großväter sind, im technischen Dasein jedoch ein Nachfolger nur Berechtigung hat, wenn er über wesentlich mehr Fähigkeiten als sein Ahn verfügt.





Der Urahn der Einplatinencomputer auf der Basis des 6504 wurde 1981 von Herwig Feichtinger erschaffen. Die Familie der leistungsfähigen Einplatinenrechner wuchs, doch ein echter 65er-Nachfolger war außer zwei 6502ern, die Thienel und Sauer auf die Platine brachten, nicht darunter.

Auch seitens der Chiphersteller gab es kaum Hoffnung: Es schien, als habe sich die gute alte 65er-CPU im Kinderkram der C16 und C64 derart verausgabt, daß sie als CPU einfach niemand mehr ernstnehmen wollte. Einzig Rockwell tat noch ein wenig, als es eine leicht verbesserte CMOS-Version der 6502-CPU auf den Markt brachte. Aber so etwas Tolles wie der 84C15 für die Z80-Freunde war nicht in Sicht.

Neuer 65er aus Japan

Wenn es die Amerikaner nicht hinbekommen, dann schaffen es vielleicht die Japaner – und sie schaffen es. Mitsubishi bietet seit einiger Zeit eine grandiose Familie von Microcontrollern mit 65er-Kern an. Ein Blick ins neue Datenbuch wirkte wie eine Frischzellen-Therapie: aus diesem Ding ließe sich ein EMUF machen – aus diesem Ding mußte ein EMUF gemacht werden.

Alle Freunde der 65er-Familie werden unsere Aufregung verstehen, wenn sie einen Blick auf die Möglichkeiten dieses Superchips werfen. Der Controller hat die Bezeichnung M 50734 und weist folgende integrierte Funktionen auf:

- Parallele Ein- und Ausgänge
- Serieller synchroner Kommunikationskanal
- Serieller asynchroner Kommunikationskanal
- Timer, auch als Zähler verwendbar
- Baudratengeneratoren
- Analog-Digitalwandler
- Pulsbreitenmodulationsausgänge
- Schrittmotorlogik mit programmierter Ausgabesequenz
- Watchdog-Überwachung
- und vieles mehr.

Alle angegebenen Funktionen sind natürlich im „Enkel“, dessen offizieller Name EMUF-734 lautet, enthalten. Zusätzlich ist auf der 96 mm × 73 mm großen Platine aber auch sonst alles drauf, was Sie normalerweise von einem EMUF erwarten (*Tabelle 1*). Die äußeren Abmessungen der Platine wurden so gewählt, daß der EMUF-734 auf eine 100 mm × 160 mm-Europakarte als „Subsy-

PROJEKT

Tabelle 1: Merkmale des Enkel-EMUF

- Mikrocontroller M50734, mit 6502-CPU-Kern
- Taktfrequenz 10 MHz
 - Zykluszeit ca. 0,8 µs bei 10 MHz
 - Seriell I/O 1x synchron, 1x asynchron (UART)
 - 3 freie Ports
 - Pulsbreiten-Modulations-Ausgang (PWM)
 - 32 KByte CMOS-RAM
 - 32 KByte CMOS-EPROM
 - Echtzeituhr (RTC)
 - Watchdog (MAX690)
 - Batterie für RAM und Uhr
 - Adreß-/Datenbus rausgeführt
 - Adreßdecoder mit GAL16V8 realisiert
 - 5 Selects rausgeführt und frei
 - 2 Ausgangskanäle Schrittmotorlogik
 - 4 analoge Eingänge (8 Bit), Referenzdiode für A/D-Wandler
 - Anschluß von IF-Modulen möglich, wenn kabelseitig nur die Leitungen +5V, GND, RXD und TXD benutzt werden
 - Ausbau über Motherboard möglich
 - Größe ca. 73 mm x 96 mm
 - Stromversorgung 5 V

stem“ aufgesteckt werden könnte, sich also hier Lösungen mit kundenspezifischer Grundplatine und EMUF-734 denken lassen. Das alles in Vier-Lagen-Multilayer, mindestens aber Feinstleiter auf die Karte, SMD-RAMs und ... Nein, so nicht! Den Status eines wirklich würdigen Enkels hatten sich die Entwickler anders vorgestellt: Zum Vermächtnis des ersten aller EMUFs gehörte der „Bausatz-Gedanke“, der da lautet, „Ein EMUF muß als Bausatz erhältlich und mit einem ganz normalen 20-Watt-Kolben von einem ganz normalen Techniker zu löten sein.“ Die einzige Konzession an „moderne Dichte“ ist das Rastermaß des Controller-Sockels, der 64 Pins unterbringen muß, wo sonst nur 40 sind. Der Rest der Karte ist technischer Standard.

Ein Blick auf die Tabellen zeigt, was für ein Wunder-Chip da von Mitsubishi produziert worden ist. Ein echter Controller. Und so schnell, daß selbst getunten 65ern schwindelig werden kann.

Eigentlich, wird einigen jetzt durch den Kopf gehen, müßte man bei den Möglichkeiten, die das Ding für die Motorensteuerung hat, doch gut einen Plotter damit steuern können, oder einen Drucker, so etwas „Elektro-Mechanisches“. Na dann schrauben Sie Ihren Drucker mal auf! Wenn der Printer aus Japan kommt, haben Sie eine gute Chance, „unsere CPU“ an ihrem Arbeitsplatz zu entdecken.

Soweit erst einmal zur CPU. Wie sieht denn der Rest der Karte aus? Natürlich EMUF-üblich. Alles rund um die CPU ist nicht dazu

angetan, einen vom Stuhl zu hauen. Die Begleitелеktronik ändert sich doch nicht so schnell, wie es manchmal den Anschein hat. Wir wollen jedoch die einzelnen Funktionsblöcke kurz erläutern:

MAX 690: Dieser Baustein ist den meisten sicherlich bereits bekannt und vertraut, da er schon auf mehreren EMUFs gute Dienste leistet. Er ist der Baustein für die Power-on-

Reset-Schaltung (mit einstellbarer Schwellenspannung), er ist Watchdog-Timer (einstellbar über einen Kondensator) und schaltet die Versorgungsspannung für RAM und Uhr, wenn nötig, auf die Lithium-Batterie um. Hier beim EMUF-734 werden sämtliche Funktionen des MAX 690 genutzt, ein MAX im MAXimal-Einsatz.

RTC 72421: Noch ein Bekannter von vielen EMUFs. Eigentlich die Standard-Uhr auf den meisten Industriekarten, die mittlerweile auch von OKI gefertigt und angeboten wird. Hier zur Auffrischung einige Grunddaten:

- Einchip-Echtzeituhr mit integriertem Quarz 32,768 kHz
- Register für Sekunden, Minuten, Stunden, Wochentag, Datum, Monat, Jahr, Status, Korrektur, Test, Reset, Interrupt usw.
- Direkter Busanschluß, also schneller Datenzugriff
- Sehr geringer Stromverbrauch (Batteriebetrieb)
- Über Jumper J1 wird ein Schreiben verhindert.

Der Baustein GAL16V8

Der aufmerksame Beobachter der Mikroprozessor-Technik hat in den letzten Jahren feststellen können, daß die GALs eine ungeheure Verbreitung gefunden haben – unterstützt beispielsweise auch durch die in der mc erfolgten Veröffentlichung des mit 98 Mark sehr preiswerten GAL-Programmierers von Thomas Schlenger-Klink. Und seit es für weitere knapp hundert Mark mit GDS 1.3 einen phantastischen GAL-Assembler gibt, ist die Entwicklungsumgebung für diese programmierbaren Vielkötner so preiswert geworden, daß die kleinen Dinger heute fast überall zu finden sind – so auch hier als Decoder. Für Neueinsteiger: GAL bedeutet hier „Generic Array Logic“.

Die weiteren verwendeten ICs sind:

- 74HC573 als Zwischenspeicher für die Adressen 0 bis 7,
- 74HC00 für die Verknüpfung der Signale WR, RES, CS2, um ein falsches Schreiben in RAM und Uhr bei Spannungswerten außerhalb der zugelassenen Pegel zu verhindern.
- EPROM 27C256 als 32-KByte-Programmspeicher
- 32 KByte RAM mit CMOS oder Low-power
- Referenzdiode ZN456 (ca. 2,5 V) für den AD-Wandler

Tabelle 2: Anschlußbelegung

| ST-Verbinder 1, 40polig | |
|-------------------------|---------|
| Pin | Pin |
| 1 +5V | 2 +5V |
| 3 GND | 4 GND |
| 5 UEXT | 6 PH1 |
| 7 /CS7 | 8 /CS6 |
| 9 NC | 10 /CS5 |
| 11 /CS4 | 12 /RD |
| 13 A15 | 14 SYNC |
| 15 /WR | 16 A12 |
| 17 A14 | 18 A7 |
| 19 A13 | 20 A6 |
| 21 A8 | 22 A5 |
| 23 A9 | 24 A4 |
| 25 A11 | 26 A3 |
| 27 /WR | 28 A2 |
| 29 A10 | 30 A1 |
| 31 /CS1 | 32 A0 |
| 33 AD7 | 34 AD0 |
| 35 AD6 | 36 AD1 |
| 37 AD5 | 38 AD2 |
| 39 AD4 | 40 AD3 |

| ST-Verbinder 2, 20polig | |
|-------------------------|---------|
| Pin | Pin |
| 1 P0.3 | 2 P0.2 |
| 3 P0.1 | 4 P0.0 |
| 5 P0.4 | 6 P0.5 |
| 7 P0.6 | 8 P0.7 |
| 9 P1.0 | 10 P1.1 |
| 11 P1.2 | 12 P1.3 |
| 13 P1.4 | 14 P1.5 |
| 15 P1.6 | 16 P1.7 |
| 17 P2.0 | 18 P2.1 |
| 19 P2.2 | 20 P2.3 |

| ST-Verbinder 3, 10polig | |
|-------------------------|--------|
| Pin | Pin |
| 1 +5V | 2 P2.4 |
| 3 P2.5 | 4 P2.6 |
| 5 P2.7 | 6 P3.0 |
| 7 P0.6 | 8 P0.7 |
| 9 P3.1 | 10 GND |

| ST-Verbinder 4, 14polig | |
|-------------------------|---------|
| Pin | Pin |
| 1 P3.2 | 2 P3.3 |
| 3 P3.4 | 4 P3.5 |
| 5 +5V | 6 P3.6 |
| 7 /RES | 8 P3.7 |
| 9 GND | 10 GND |
| 11 P4.3 | 12 P4.2 |
| 13 P4.1 | 14 P4.0 |

PROJEKT

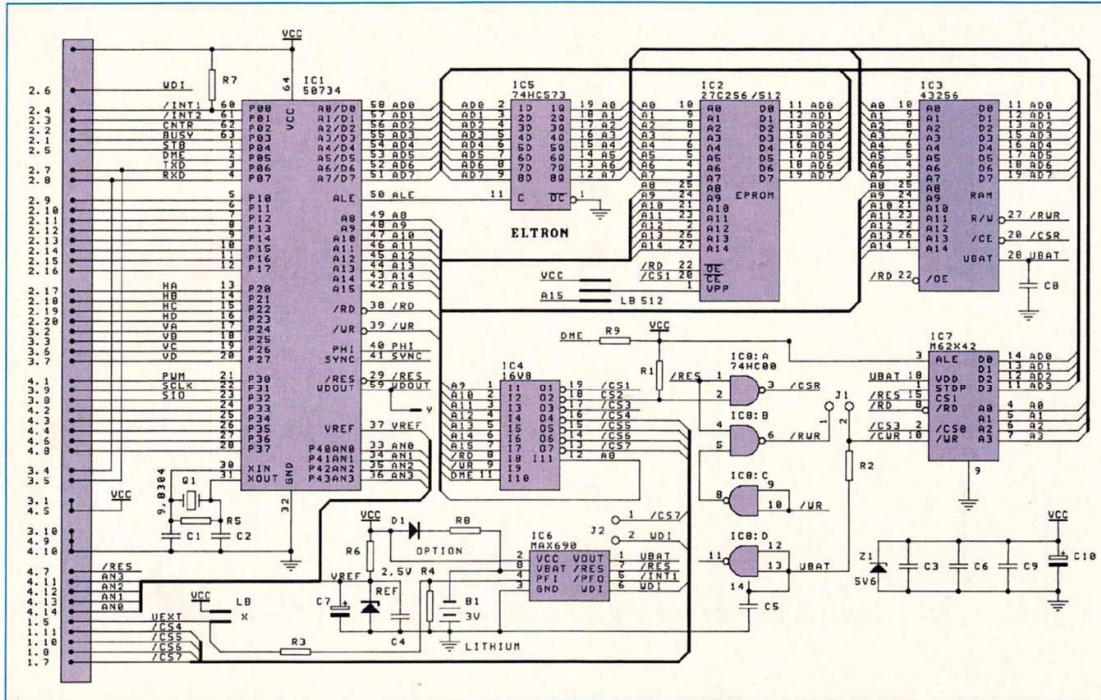


Bild 1. Das Schaltbild: Man glaubt es kaum, aber das alles ist auf gerade mal 70 Quadratcentimetern untergebracht

Nicht unwichtig für die Aufrechterhaltung der Hintergrundinformationen und zur Sicherung bei Spannungsausfall ist die Lithium-Batterie.

Die Schaltung

Das Schaltbild zeigt uns den EMUF-734 in seiner ganzen Einfachheit. Sie sehen, wir haben nichts Überflüssiges mit auf die Karte gepackt, aber auch nichts Grundsätzliches fehlen lassen. Als größter Baustein ist der M50734-CPU-Chip mit seinen vier freien Ports zu erkennen. Wie Sie der Tabelle 2 entnehmen können, sind diese Ports teilweise mit Zweitfunktionen belegbar. An der rechten Seite sieht man den gemultiplexten A/D 0 - 7-Bus, der zuerst die niedrigen Adreßbits A0 bis A7 ausgibt, die mit der negativen Flanke des ALE-Signals in den Zwischenspeicher 74HC573 geschrieben werden und danach die Datenbits D0 bis D7 führt. Die höheren Adreßbits haben einen eigenen statischen Ausgabeport.

Weitere CPU-Signal-Ein- und -ausgänge:
 RD = Lesesignal
 WR = Schreibsignal
 PHI = Systemtakt

SYNC = Synchronsignal
 RES = Reseteingang
 WDOUT = Watchdogausgang

Unterhalb dieser Leitungen sehen wir V_{REF} , also den Spannungsreferenz-Eingang und die vier Analog-Eingänge AN0 bis AN3. Zu guter Letzt befindet sich auf der linken Seite der Taktgeber, ein Quarz mit einer Schwingfrequenz von 9,8304 MHz. Der scheinbar „krumme“ Wert wurde mit Rücksicht auf möglichst viele Baud-Raten gewählt. Wir erreichen so immerhin elf Standardwerte. Nach dem bereits erwähnten Zwischenspeicher 74HC573 folgen in gleicher Reihe das EPROM 27C256 und das 32-KByte-RAM. Diese Bausteine weisen, vielleicht abgesehen von der Batteriestromversorgung des RAM, keine Besonderheiten auf. Unterhalb des Programm- und Datenspeichers befindet sich der GAL 16V8, der, wie bereits erwähnt, hier nur als programmierbarer Decoder eingesetzt ist. Übrigens verbrauchen die neuen GALs (wie die S- und AS-Typen) schon viel weniger Strom als die alten, die als Stromfresser berüchtigt waren. Vier freie Ausgänge (CS4 bis CS7) wurden noch für externe Erweiterungen auf dem Steckverbinder zur Verfügung gestellt. Jetzt

folgt nach unten gesehen der MAX690, der hier optimal genutzt wird:

- er erzeugt beim Einschalten der Versorgungsspannung ein verzögertes Resetsignal, dessen Schwellenwert durch R3 und R4 bestimmt ist,
- er erkennt hierdurch auch einen Ausfall der Betriebsspannung, da R3 entweder über die Lötbrücke direkt an +5 V oder über die Steckverbindung SV1.5 an eine externe (wie die gesiebte Vorspannung für +5 V) Spannung angeschlossen wird. Die externe Lösung bietet den Vorteil, daß über den Ausgang \overline{PFO} ein frühzeitiger Interrupt ausgelöst werden kann und somit für die CPU Zeit bleibt, Status und Daten zu retten.
- er spielt hier den Watchdog! So bleiben alle Leitungen unbeschaltet, wenn die Leitung WDI beschaltet und bedient wird.
- zuletzt übernimmt das 8-beinige-Mäxchen auch noch das Umschalten von +5 V auf „Notstrom“ aus der Lithium-Batterie, das heißt, er schaltet U_{BAT} immer auf die höhere zur Verfügung stehende Spannung um. Damit versorgt er dann das RAM und die Uhr. Links vom MAX befindet sich die Referenzdiode ZN458, die mit Hilfe von R6 und C7 eine temperaturstabile Referenzspannung

PROJEKT

von 2,5 V für den A/D-Wandler erzeugt. Der 74HC00 verknüpft die Signale CS2 mit RES zum CSR = RAM-Select, damit, wenn das RES-Signal low ist, das RAM nicht fälschlich beschrieben werden kann. Das Gatter B erzeugt durch die Verbindung mit WR und RES ein geschütztes RWR-Schreibsignal für das RAM und die Uhr.

Der Aufbau des EMUF-734

Der Aufbau des EMUF-734 ist nahezu ein Kinderspiel, – wenn man löten kann. Die „alten Hasen“ sollten die nächsten Punkte mal ausnahmsweise nicht einfach überschlagen, Sie sollten alle Punkte vor dem Bestücken kennen.

Die Reihenfolge des Aufbaus:

Alle Bauteile rein – mit dem Kolben drüber – Versorgungsspannung dran und Luft anhalten: Rollt er nach links... oder qualmt er schon rechts?

Nein! So nicht. Beschreiten Sie lieber den richtigen Weg – und nehmen Sie sich für jeden Schritt Zeit:

- Öffnen Sie den Bausatz – und nehmen Sie den MAX690 und die Lithiumbatterie heraus. Dazu kommen wir später. Stecken Sie nun alle flachen Bauteile, die verlötet werden sollen (z. B. Widerstände), auf die Leiterkarte (Achtung: auf die Bestückungsseite) Nun die Leiterkarte (Vorsicht, damit nichts rausfällt) umdrehen und alles verlöten. Jetzt die nächst höheren Bauteile (z. B. IC-Fassungen) einstecken, Platine drehen und löten. Arbeiten Sie ruhig langsam, aber dafür voll konzentriert und aufmerksam. Vom Einsatz 100wättiger Schmoreisen wird hier dringend abgeraten. Verwenden Sie (wie hoffentlich immer) eine feine Spitze und gutes Zinn. Die Steckverbinder SV1 bis SV4 können Sie, je nach Anwendung, oben oder unten bestücken, die Numerierung im Schaltplan geht von der Bestückungsseite/Oberseite aus.

- Nach dem sorgfältigen Kontrollieren wirklich aller (!) Lötstellen (die Fassung der CPU hat 64), und wenn Sie wirklich keinen Fehler (mehr) entdecken, dann können Sie nun die +5 V-Versorgungsspannung anlegen. Die +5 V kommen am besten aus einem strombegrenzten Netzteil. Jetzt sollte sich der EMUF-734 von seiner sparsamsten Seite zeigen. Wenn nicht, steckt irgendwo ein Löt- oder Bestückungsfehler. Ist er aber sparsam, das heißt, verbraucht er wirklich nur einige Milliampere, dann können Sie die +5 V abnehmen und den MAX690 einlöten. Er steckt unter der CPU und wird (deswegen) als einziges IC ohne Fassung eingelötet.
- Sie sollten sich schon entschieden haben,

Tabelle 3: Monitor-Grundbefehle

| | |
|---|---|
| A | = Analog-Input lesen (im 0,1-Sekunden-Zyklus) |
| B | = Breakpoint setzen |
| C | = Inhalt des Speichers vergleichen |
| D | = Inhalt des Speichers ausgeben |
| E | = Inhalt des Speichers anzeigen und verändern |
| F | = Inhalt des Speichers füllen |
| G | = Programmausführung starten |
| H | = Parameter für den Schrittmotor setzen (X-Achse) |
| I | = Input-Port lesen (im 0,1-Sekunden-Zyklus) |
| J | = Benutzer-Unterprogramme aufrufen |
| K | = Extended Memory anwählen |
| L | = Datei in den Speicher laden – (Datei, HEX) |
| M | = Inhalt des Speichers verschieben |
| N | = Interrupt-Control-Register lesen und setzen |
| O | = Port-Ausgabe schreiben |
| P | = Parameter des PWM-Output setzen |
| Q | = Port-Status lesen und schreiben |
| R | = Register anzeigen und ändern |
| S | = Speicher in eine Datei schreiben – (Datei, OBJ) |
| T | = Programmschritt ausführen mit Registerausgabe |
| U | = Programmschritt ausführen |
| V | = Parameter für den Schrittmotor setzen (Y-Achse) |
| W | = Real-Time-Clock setzen |
| X | = Schrittmotor starten (X-Achse) |
| Y | = Schrittmotor starten (Y-Achse) |
| Z | = Real-Time-Clock anzeigen |

Die Sonderbefehle:

| | |
|-----|---|
| F1 | = Allgemeine Hilfe Ctrl-F1 = gezielte Hilfe |
| F2 | = COM-Port wechseln |
| F3 | = Datei.PRN – oder Text-Datei laden |
| F4 | = Adressen-Suche ein- bzw. ausschalten |
| F5 | = Fenstergröße verändern |
| F6 | = Terminal-Modus ein- bzw. ausschalten |
| F7 | = Programm-Counter-Adresse suchen |
| F8 | = Assembler |
| F9 | = Disassembler |
| F10 | = Betriebssystem |

Bitte beachten Sie:

Der Monitor belegt die nachfolgenden Speicherbereiche:

| | |
|--------|---------------------------------|
| RAM: | 00C9 – 00CF Hex: Zero-Page-RAM |
| | 0100 – 011F Hex: Stack-RAM |
| | 0200 – 02FF Hex: Arbeits-RAM |
| | 7FF4 – 7FFF Hex: Vektor-RAM |
| EPROM: | E800 – FFFF Hex: Programm EPROM |

Für den Benutzer stehen die nachfolgenden Bereiche zur Verfügung:

| | |
|--------|---------------------------------|
| RAM: | 0000 – 00C8 Hex: Zero-Page-RAM |
| | 0120 – 01FF Hex: Stack-RAM |
| | 0300 – 7FF3 Hex: Arbeits-RAM |
| EPROM: | 8000 – 7FF3 Hex: Programm-EPROM |

ob MAXe nun mit 5 V oder mit externer Spannung im Schwellenwert für RESET versorgt werden soll. Also R3 richtig gewählt und die Lötbrücke entsprechend geschlossen oder auch nicht.

- Jetzt montieren Sie, ohne einen Kurzschluß zu verursachen, die Lithium-Batterie (eventuell auf Stiften).

- Wird in diesem Bestückungszustand die 5-V-Versorgungsspannung angelegt, kann man jetzt schon das RESET-Signal auf einwandfreie Funktion überprüfen (+5 V ± 5%). Die Referenzspannung von 2,5 V ist jetzt ebenfalls schon zu messen.

- Wenn jetzt auch schon das Umschalten von +5 V auf Batteriespannung am Mäxchen bei einer Eingangsspannung von kleiner 3 V (entsprechend der Batteriespannung) funktioniert, dann können Sie nun (aber bei abgeschalteter Versorgungsspannung!) die ICs in die Fassungen stecken. Achten Sie auf zwei Dinge besonders: Stecken Sie die ICs richtig herum in die Fassungen und malträtiert Sie die Pins nicht. Wie immer: ruhig und konzentriert.

- Eine anschließende Stromverbrauchsmessung sollte nur Milliampere-Werte ergeben, sonst geht es an die Fehlersuche.

Nun könnte es sofort losgehen, denn mit dem EMUF-734 haben Sie wahrscheinlich den MONI-734 erhalten, und der hat es ganz gewaltig in sich, aber zunächst müssen Sie den EMUF-734 jetzt mit einem Terminal (also Ihrem PC) verbinden. Verwenden Sie eines der IF-Module (z. B. IF232), dann brauchen Sie noch ein Kabel, das das IF-Modul mit COM1 oder COM2 Ihres Rechners verbindet. Wenn Sie das haben, dann kann es wirklich abgehen.

Das Monitorprogramm des EMUF-734 ist auf jedem IBM-Kompatiblen ablauffähig und ist eigentlich weniger ein schlichter Monitor, sondern eher eine mehr als passable Entwicklungsumgebung. Wir unterscheiden zwei Arten von Befehlen: Grund- und Sonderbefehle. Beide Befehlsarten können mit dem Programm MTM.EXE bearbeitet werden. Über die Taste F1 können Hilfe-Funktionen zu allen Monitorbefehlen sowie Informationen zum Betrieb des Entwicklungsrechners abgerufen werden (Tabelle 3). Das ermöglicht eine sehr komfortable Bedienung und ist, wir ziehen unsern Hut vor allen Pullup-und-down-Menüs, selbst bei EMUFs nun möglich. Alle Hilfetexte sind in der Datei MTM-HLP gespeichert, die vom Kunden bei Bedarf sogar erweitert werden kann. Jetzt wissen Sie schon, ohne die Befehlsliste gesehen zu haben, daß dieser Monitor, der noch Assembler und Disassembler enthält, etwas ganz Außergewöhnliches und Beson-

Tabelle 4: Erklärung der zusätzlichen Befehle

BBC A, i, REL = BRANCH ON BIT CLEAR, im ACCU, verzweigt bei BIT=0 innerhalb - 128 + 127 Byte
 BBC ZP, i, REL = BRANCH ON BIT CLEAR, in ZEROPAGE, sonst wie oben
 BBS A, i, REL = BRANCH ON BIT SET, im ACCU, sonst wie oben
 BBS ZP, i, REL = BRANCH ON BIT SET, in ZEROPAGE, sonst wie oben
 CLB i, A = CLEAR BIT im ACCU
 CLB i, ZP = CLEAR BIT in ZEROPAGE
 CLT = CLEAR TRANSFER-FLAG, das T-Flag bestimmt, ob Daten innerhalb der ZEROPAGE unter Umgehung des ACCU ausgetauscht werden.
 COM ZP = COMPLEMENT in ZEROPAGE, komplementiert Daten und speichert sie zurück.
 DEC A = DECREMENT, Inhalt des ACCU -1
 INC A = INCREMENT, Inhalt des ACCU +1
 LDM, ZP = LOAD IMMEDIATE DATA TO MEMORY, schreibt Daten unter Umgehung des ACCU direkt in ZEROPAGE.
 RRF = ROTATE RIGHT OF FOUR BITS, tauscht die 2 NIBBLES eines ZEROPAGE-Byte aus
 SEB i, A = SET BIT, im ACCU
 SEB i, ZP = SET BIT, in ZEROPAGE
 SET = SET TRANSFER-FLAG (siehe CLT)
 STP = STOP, stoppt den Oszillator. Nur ein externer Interrupt oder ein RESET können den Controller wieder starten.
 TST ZP = TEST FOR NEGATIVE OR ZERO, setzt N- oder Z-FLAG
 WIT = WAIT, Controller wird in den WAIT-Modus gesetzt, wartet auf internen/externen Interrupt oder RESET.

Ist das Transferflag gesetzt, ermöglicht das Operationen durchzuführen, ohne dabei den Akkumulator zu benutzen oder zu verändern. Folgende Befehle lassen dies zu: ADC, SBC, CMP, AND, EOR, ORA und LDA. Hierbei wird nicht, wie gewohnt, der Inhalt aus dem Akkumulator mit einem angegebenen Speicherinhalt bearbeitet, verknüpft oder geladen, sondern die Zero-Page-Speicherstelle, die durch das Register X adressiert wird.

T-Flag = 0 - A < A * Mem2
 T-Flag = 1 - Mem1 < Mem1 * Mem2

Dabei ist A der Akkumulator, Mem1 die Zero-Page-Speicherstelle, auf die Register X zeigt, Mem2 ist die Speicherstelle, mit der operiert wird und * ist die auszuführende Operation. So kann das Umschalten der Transfer-Flag eine sehr sinnvolle und hilfreiche Unterstützung bei der Programmierung sein.

deres ist. Und deswegen soll an dieser Stelle den beiden Entwicklern dieses Monitors der Leistungsklasse „Erste Sahné“ gedankt werden. Es handelt sich um das tschechoslowakische Software-Expertenduo Jiri Sloupensky und Otakar Müller, die mit dieser Arbeit einen neuen Standard an Leistungsfähigkeit und Komfort für EMUF-Monitore gesetzt haben.

Einige Besonderheiten sollten alte 65er-Hasen unbedingt beachten, wenn sie mit der Mitsubishi-CPU arbeiten, denn sie kann in der Tat ein ganzes Eckchen mehr, als die „alte 65er“. Sie macht auch einiges ganz anders als der Opa wie Sie der *Tabelle 4* entnehmen können.

Eine weitere Besonderheit der CPU verdient Beachtung: Wenn Speicher- und/oder Peripherie-Bausteine in der zweiten Bank liegen, muß zuerst das DME-Signal aktiv geschaltet werden. Alle Zugriffe auf eine Erweiterung können nur mit einer indirekten Adressierung über die Zero-Page in Verbindung mit Register Y erfolgen. Hier zwei Beispiele: LDA (name),Y oder STA (name),Y

Auch bei den Adressierungsmodi hat sich einiges geändert, was der Benutzer unbedingt in Mitsubishis „Series 740 User's Manual, Software“ nachlesen sollte, einen klei-

nen Überblick wollen wir aber jetzt schon geben:

Folgende Adressierungsarten sind möglich:

- IMMEDIATE
- IMPLIED
- ACCUMULATOR BIT
- ACCUMULATOR
- RELATIVE
- ZERO PAGE
- INDIRECT
- ZERO PAGE X
- INDIRECT X
- ZERO PAGE Y
- INDIRECT Y
- ABSOLUTE
- INDIRECT ZERO PAGE
- ABSOLUTE X
- SPECIAL PAGE
- ABSOLUTE Y
- ZERO PAGE BIT

Daran sollten Sie denken: alle ZEROPAGE-modifizierenden Befehle und Adressierungsarten betreffen ebenso die Ports.

Tja, und dann sind uns zu unserem großen Schrecken folgende Abnormalitäten und Macken aufgefallen, mit denen der Anwender, wenn er sie kennt, allerdings gut leben kann:

- NOPS (und zwar je einer) sollten jeweils zwischen die Folge von einigen Befehlen eingefügt werden. Und zwar nach ADC und SBC im Decimal-Mode vor SEC, CLC, SED, CLD, und nach Ausführung von PLP.

- Die Adressen 00D0 bis 00D9 sind für eine Systemerweiterung vorgesehen und dürfen daher nicht anderweitig benutzt werden.

- Mindestens ein NOP muß eingefügt werden zwischen der Modifikation der Interrupt Request Bits und der Ausführung von BBC und BBS.

Glänzende Zukunft

Der 65er lebt! Laut Mitsubishi lebt er sogar ganz vorzüglich, denn der japanische Hersteller hat mit der 740er-Familie einen saten Marktanteil unter den 8-Bit-Controllern. Besonders Drucker, Plotter aber auch kleine Handhabungsautomaten schreien geradezu nach solchen Controllern, die sich direkt mit Schrittmotoren beschäftigen können. Bei Mitsubishi gibt es sogar einen 16-Bit-65er-Controller, so daß wir nach diesem Enkel des alten EMUF sogar einen Ururenkel ins Auge fassen, der dann mit einem C-Compiler bearbeitet werden könnte. Sollte es an diesem Enkel, dem EMUF-734, genügend Interesse geben, was wir hoffen, dann könnten wir auch noch ein schnelles Basic aus der Schublade holen. Den allermeisten wird jedoch der Super-Monitor reichen.

Wir hoffen, daß der neue EMUF viele (alte und junge) Freunde findet. Uns hat die Entwicklung viel Freude gemacht und mal wieder eins gezeigt: Eine einfache Steuerung braucht einen einfachen Rechner und nicht komplexe Gebilde wie PCs. Eine Maschine, mit dem EMUF-734 in Assembler programmiert, wird mit dem gebotenen Speicher mit Sicherheit effektiver versorgt sein, als mit jeder PC-Steuerung. Diese hochgelobten PC-Dinger besitzen im Normalfall noch nicht mal eine Programmüberwachung (Watchdog), von der EMV-Verträglichkeit mal ganz zu schweigen. Solche kleinen Schaltungen, wie dieser Enkel, haben Zukunft.

Es sollte grundsätzlich eine Symbiose zwischen PCs und Mikrocontrollern angestrebt werden. Der PC sollte verwalten – und der Mikrocontroller ausführen.

P. Thews, A. Korn, U. Meyer/ed

Literatur

- [1] User's Manual M50734, Mitsubishi Series 740 User's Manual Software
- [2] mc-Extra, Einplatinen-Sonderheft