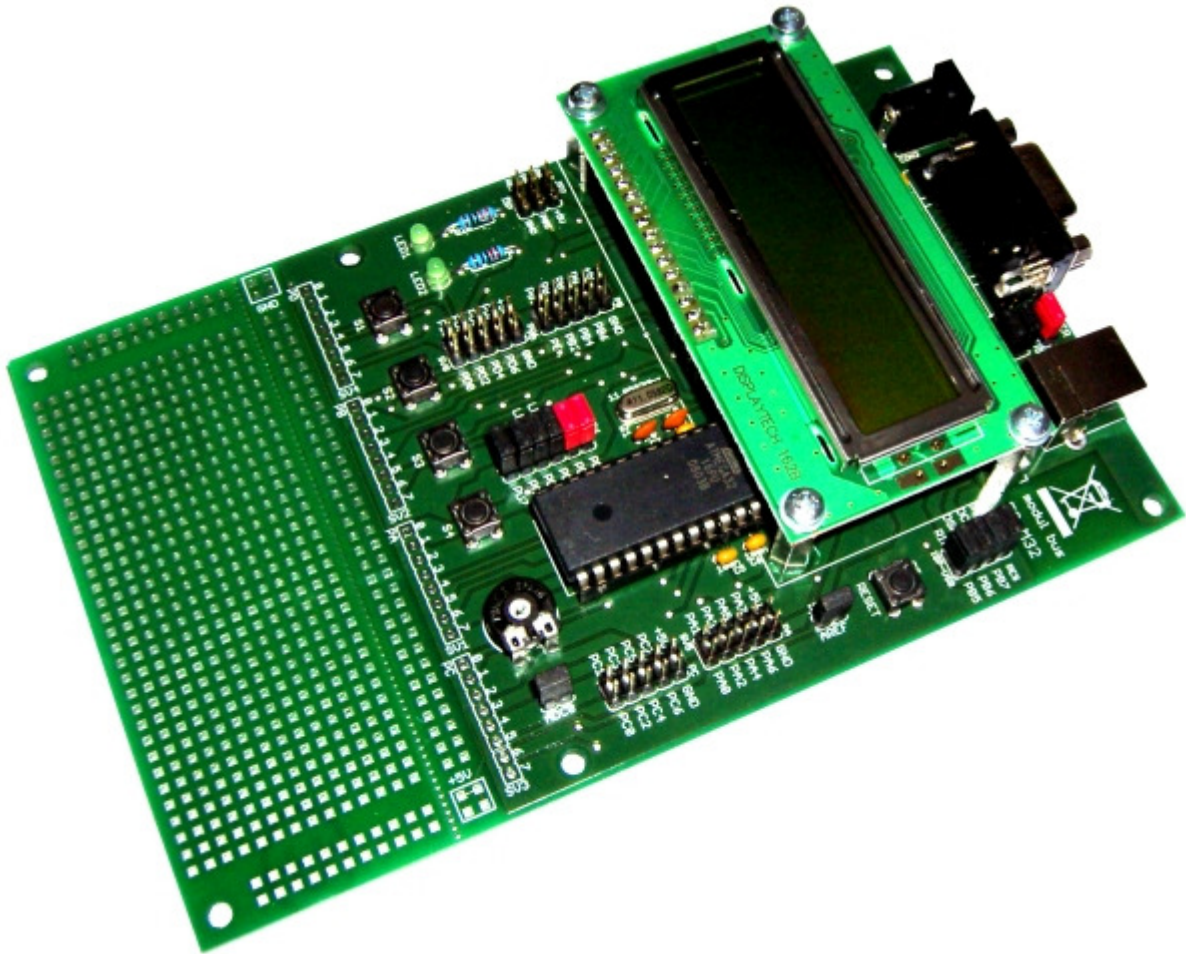


ES-M32

Entwicklungs- und Ausbildungssystem für ATmega32



Bei Modul-Bus wurde lange überlegt: Noch ein ATmega-System? Es gibt doch schon so viele. Dann haben wir uns doch dafür entschieden. Ausschlaggebend waren folgende Überlegungen:

Es gibt eine Reihe von Entwicklungssystem rund um die 8051er Familie: ES51, ES52-Flash, ES535. Diese wurden oft auch für den internen Gebrauch, also für kundenspezifische Entwicklungen verwendet. Inzwischen hat sich der Schwerpunkt jedoch mehr in Richtung AVR verschoben. Neuere Entwicklungen verwenden überwiegend ATmega und ATtiny. Und wir wollen weiterhin auch den Bereich Ausbildung unterstützen. Das System soll also vor allem auch für Schulen und Unis praktikabel sein.

ATmega

Deshalb haben wir uns überlegt, was könnte man noch besser machen, welche Eigenschaften braucht so ein System, damit es möglichst bequem und universell einsetzbar ist. Dabei kamen folgende Elemente heraus:

Gewählt wurde der Mega32, weil er noch im DIL40-Gehäuse geliefert wird, genügend Ports hat und auch sonst für die meisten Anwendungen ausreicht. Man kann aber auch den Mega16

oder den Mega644 mit mehr Speicher und zusätzliche Peripherie verwenden.

Das System hat ein LCD und einige Tasten, damit es sofort einsetzbar ist. Eine typische Anwendung wäre z.B. das Bedienteil eines autonomen Kurzwellenempfängers.

Alle Ports sind über Pfostenstecker (kompatibel zum STK500) und über einreihige Anschlüsse am Rand zugänglich. Ein großes Rasterfeld ermöglicht den Anschluss zusätzlicher Peripherie.

Schnittstellen

Das System verfügt über eine RS232 und über einen USB-Anschluss. Ein FT232R auf dem Board dient als USB/Seriell-Umsetzer. Man kann über Jumper auswählen, ob die Leitungen RXD und TXD des Controllers zum USB oder zur RS232 verbunden sind.

Die Stromversorgung kann wahlweise vom USB oder von einem separaten Netzteil kommen. Die USB-Betriebsspannung ist über eine Polyswitch-Sicherung geschützt, damit auch bei einem Fehler keine Gefahr für den PC besteht.

Auch für die Programmierung des ATmega gibt es zwei Schnittstellen. Der sechspoliger ISP-Anschluss ist kompatibel zum STK500 bzw. zum ISP mkII von Atmel. Alternativ kann das System auch über ein spezielles Programmierool von Modul-Bus über den USB-Anschluss programmiert werden, das in Kürze vorgestellt wird.

Eigentlich braucht man nur noch ein USB-Kabel für Stromversorgung, Programmierung und serielle Kommunikation mit dem System. Wenn aber andere Anschlüsse benötigt werden sind sie schon vorhanden.

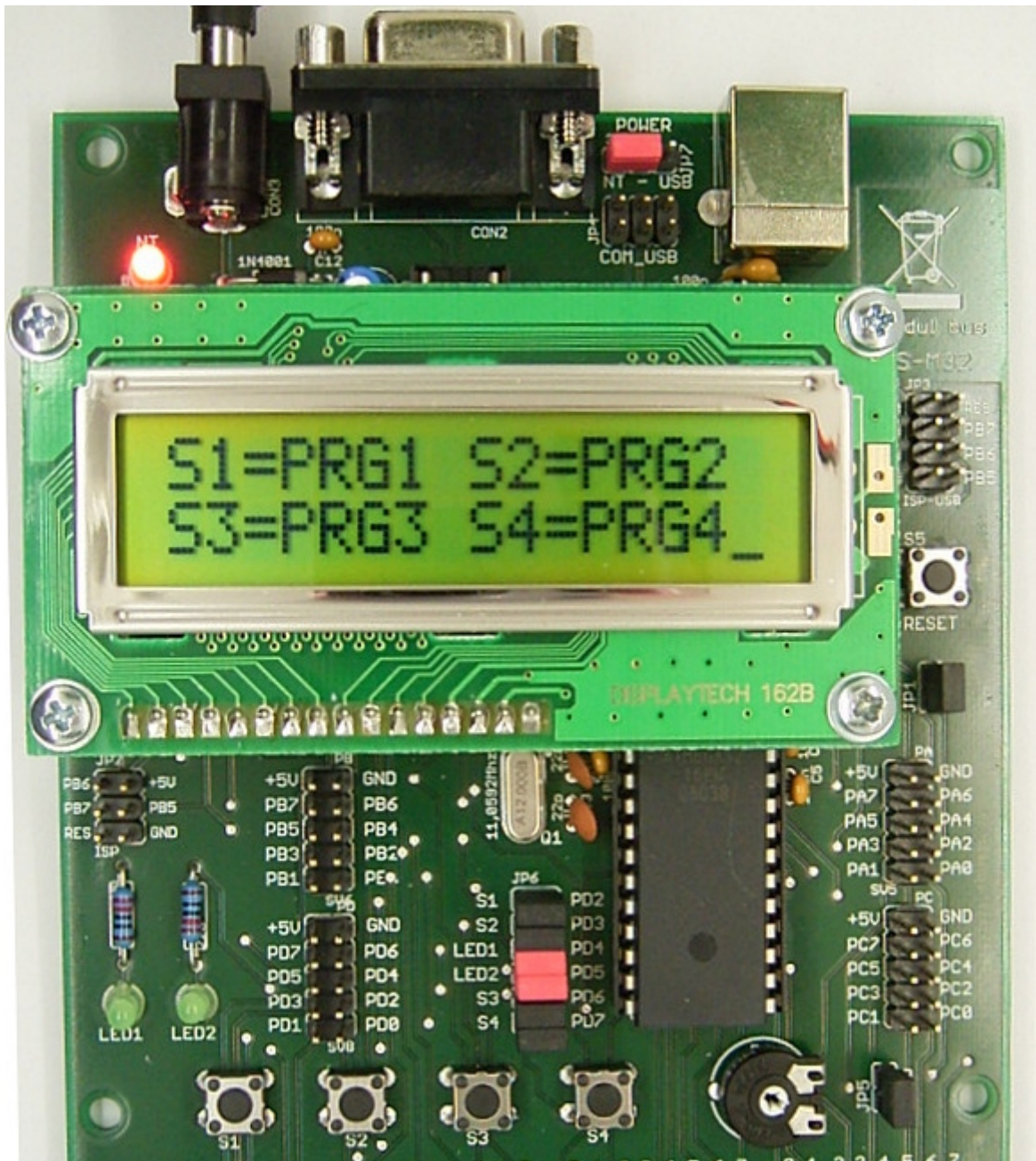
Periperie

Auf dem Board befinden sich zwei LEDs und vier Taster, die man wahlweise über Jumper mit den Ports verbinden kann. Auch die beiden PWM-Ausgänge des Mega32 können direkt die LED-Helligkeiten steuern. Außerdem gibt es ein Poti, das eine einstellbare Spannung über einen Jumper an einen Analogeingang liefert. Zusammen mit dem LCD lassen sich deshalb bereits viele Testprogramme ohne zusätzliche Hardware ausführen.

Geplant ist, dass hier in ELEXS laufend über praktische Anwendungen des Systems berichtet wird. Der Schwerpunkt liegt dabei auf Bascom-AVR, weil dieser Basic-Compiler den leichtesten Einstieg verspricht.

ES-M32

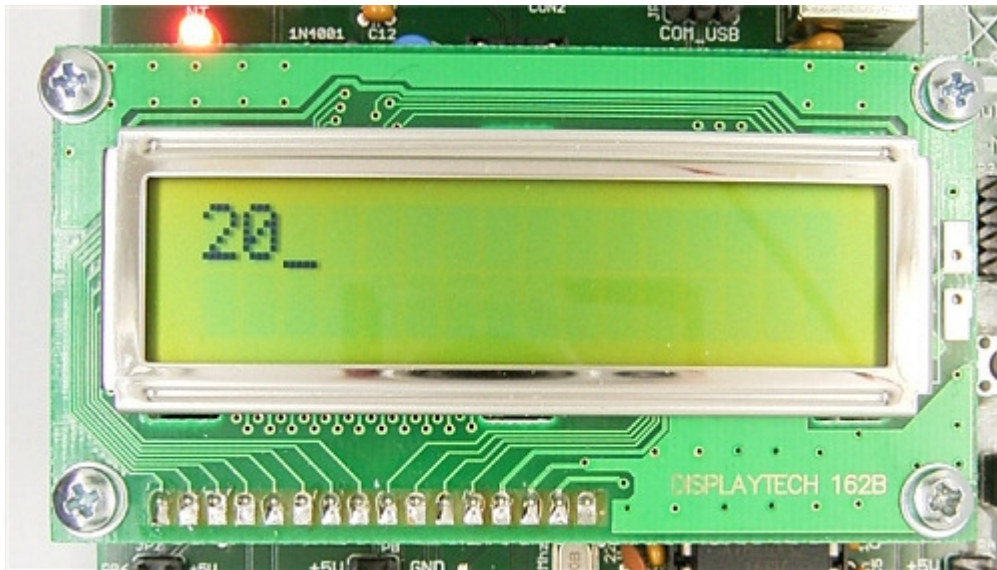
Der erste Test



Wenn man ein neues Entwicklungssystem in die Hand bekommt, muss man sich erst mal orientieren. Welche Anschlüsse gibt es, welche Jumper müssen gesteckt sein, wie wird das System mit Spannung versorgt und wie lädt man ein Programm. Damit die ersten Schritte leicht fallen wird das System mit einem schon geladenen Testprogramm ausgeliefert. Dasselbe Programm dient auch zur Endkontrolle in der Produktion. Das Programm wurde mit Bascom erstellt und soll in ELEXS noch genauer erläutert werden.

Die Spannungsversorgung soll zunächst von einem externen Netzteil kommen. Dazu muss der Power-Jumper JP7 in Richtung NT (Netzteil) gesteckt werden. Das Netzteil mit einem Hohlstecker (Pluspol innen) soll 9 V ... 13,6 V liefern. Wenn der Netzteilstecker verbunden ist, erscheint auf dem Display eine Einschaltmeldung und dann ein kleines Bedien-Menü für vier Testfunktionen.

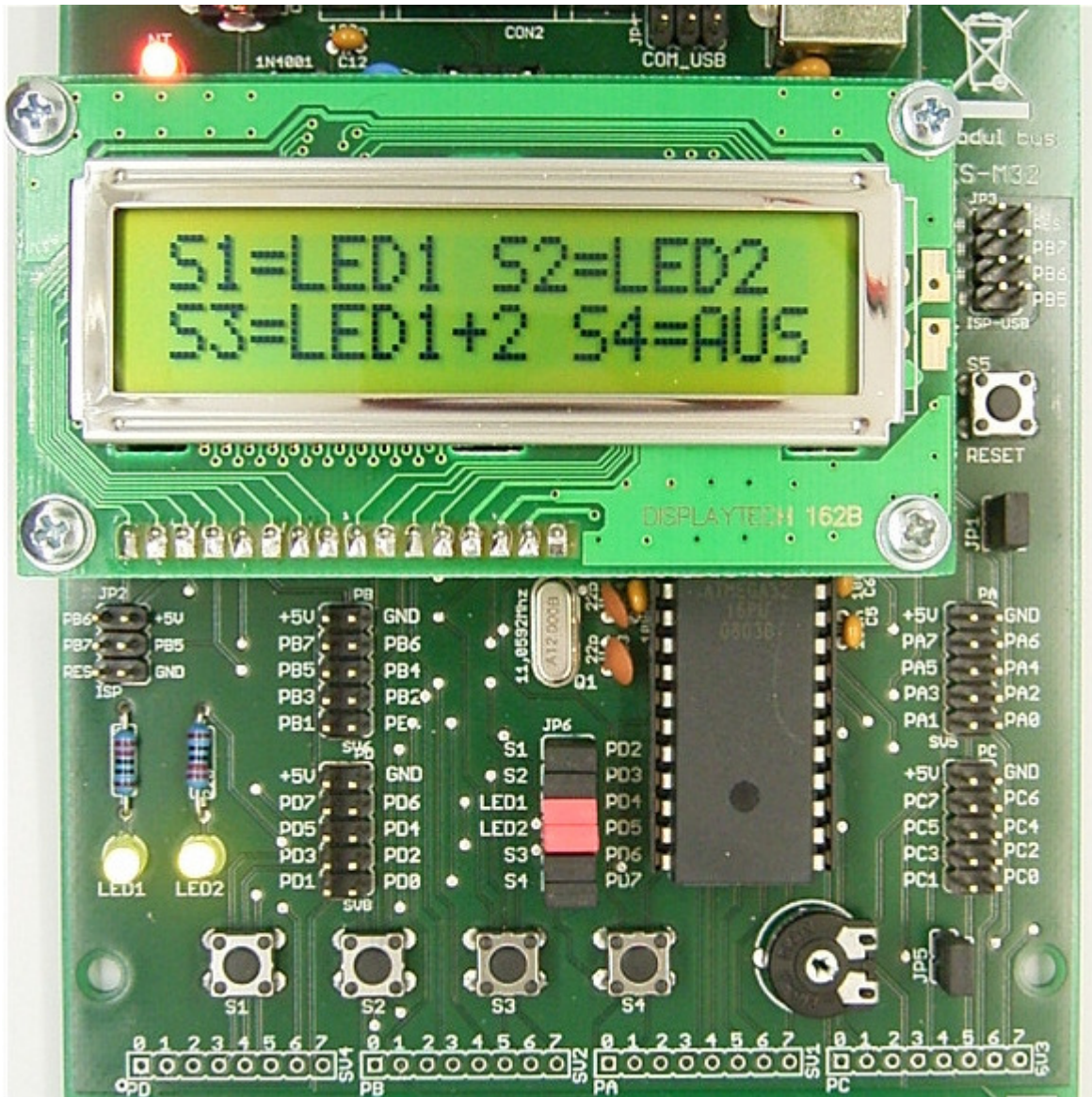
Über die vier Taster S1 bis S4 können die einzelnen Funktionen gestartet werden. Alle sechs Jumper an JP6 müssen gesteckt sein, weil sie die vier Taster und zwei LEDs mit den Ports des Prozessors verbinden. Jede gestartete Testfunktion läuft in einer Endlosschleife. Für den nächsten Test muss daher jeweils Reset gedrückt werden um die Software neu zu starten.



Der erste Test verwendet das RAM des Mega32 als Datenspeicher. Es werden 200 aufsteigende Zahlenwerte gespeichert und dann im Display angezeigt.



Test 2 verwendet den AD-Wandler. Damit man etwas messen kann muss der AREF-Jumper (JP1) gesetzt sein. Der AD-Wandler im Mega32 erhält damit eine Referenzspannung von 5 V. Damit reicht der Messbereich ebenfalls bis 5 V. Wenn zusätzlich der ADC0-Jumper (JP5) gesetzt ist, hat man eine Verbindung zum Poti auf der Platine. Mit einem kleinen Schraubendreher kann nun eine Messspannung eingestellt werden. Diese wird in der oberen Zeile als Ganzzahl bis 1023 (Auflösung 10 Bit) dargestellt, in der unteren Zeile als Spannung in Volt.



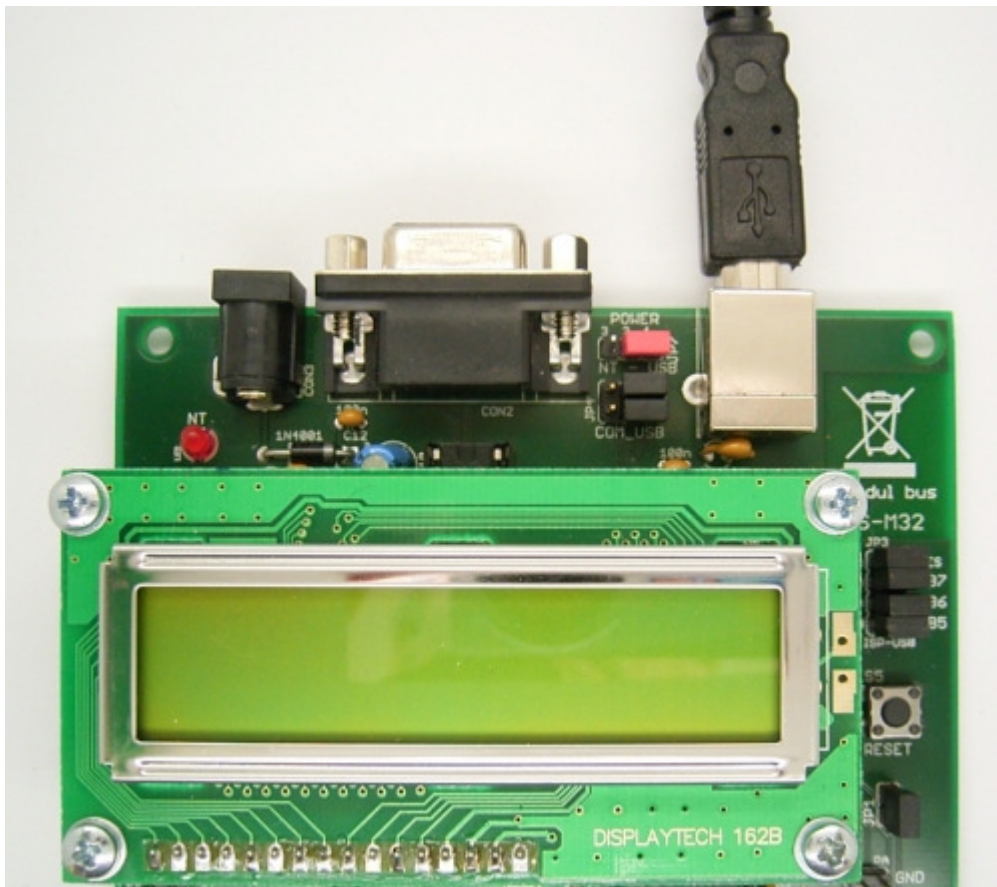
Test 3 demonstriert die Funktion der Tasten und der beiden LEDs. Mit den Tasten 1 bis 4 lassen sich die LEDs einzeln oder zusammen einschalten und auch gemeinsam ausschalten.



Der vierte Test schließlich steuert die Helligkeit der beiden LEDs über die PWM-Ausgänge des Mega32. Mit den Tasten 1 und 3 wird die Helligkeit erhöht, mit den Tasten 2 und 4 verringert. Der Maximalwert ist entsprechend der Auflösung von 10 Bit jeweils 1023.

ES-M32

Programmierung über USB

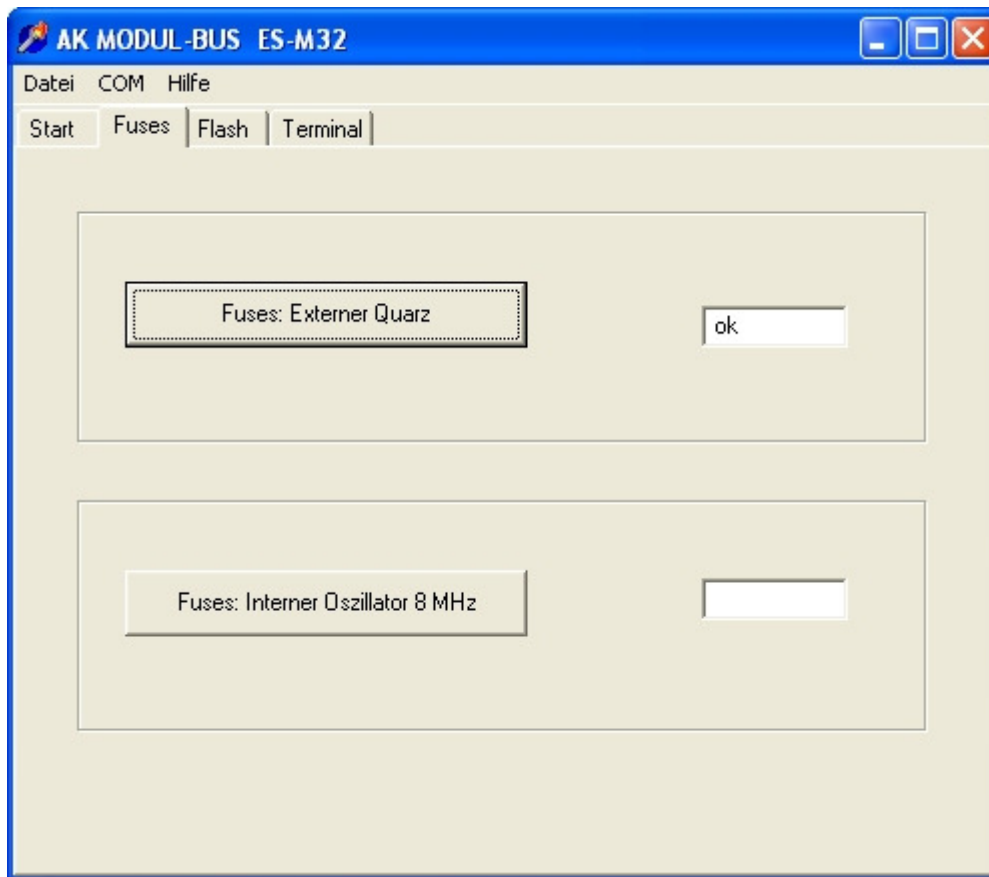


Das Entwicklungssystem ES-M32 kann komplett über den USB betrieben werden, man braucht also nur ein Kabel für alles: Stromversorgung, Programmieren, serielle Kommunikation. Das Programmier-Tool ESM32.exe erledigt alle Aufgaben. Es steuert den FT232R auf der Platine im Programmier-Modus direkt an, im Terminal-Modus über den virtuellen COM-Treiber. Während der Programmierung darf kein zweiter USB-Wandler von FTDI angeschlossen sein.

Auf der Registerkarte „Start“ sieht man ein Foto mit allen wichtigen Jumper-Stellungen. Betriebsspannung und die Leitungen RXD/TXD sind am USB angeschlossen. Außerdem sind die vier ISP-Leitungen an JP3 verbunden.

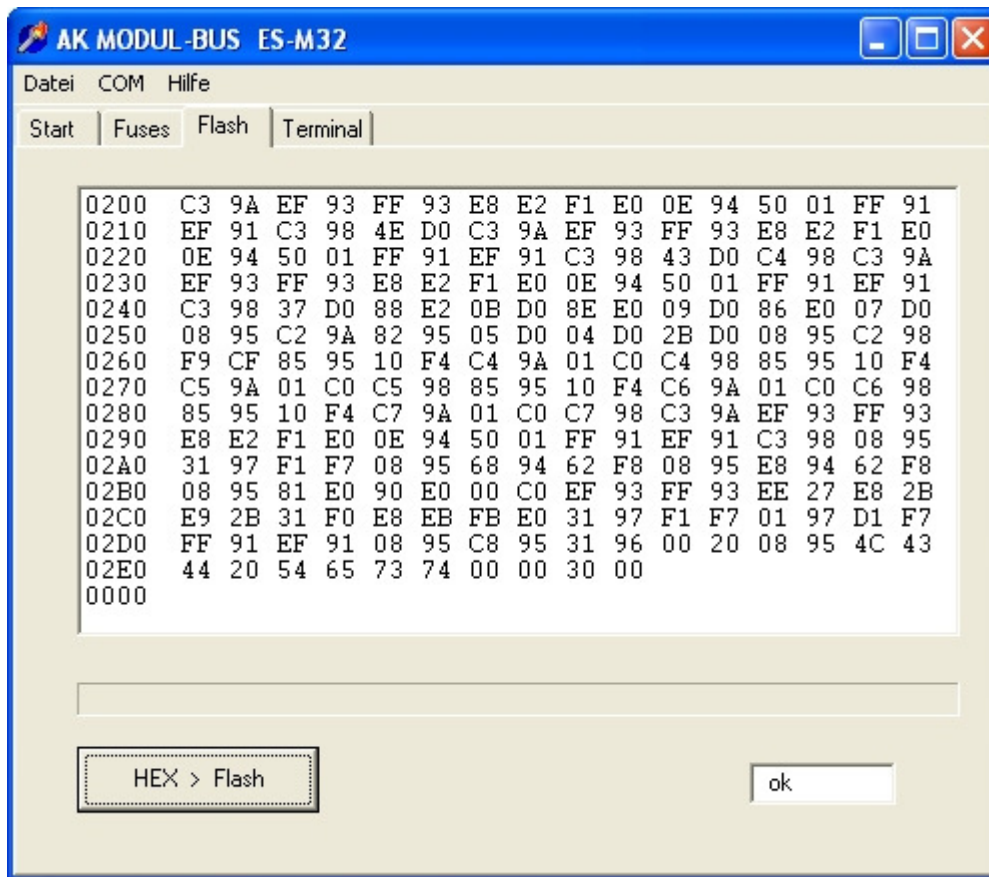


Ein fabrikneuer ATmega32 ist auf den internen 1-MHz-Oszillator eingestellt. Wenn ein neuer Mega32 verwendet werden soll müssen zuerst die Fuses programmiert werden. Diese Programmierung muss für jeden Controller nur einmal ausgeführt werden. Mit der Einstellung „Externer Quarz“ wird der 11,0592 MHz-Quarz verwendet. Alternativ kann auch der interne Oszillator mit 8 MHz ausgewählt werden. Es wurde bewusst darauf verzichtet alle Fuses individuell einzustellbar zu machen, um mögliche Fehler auszuschließen. Auch wenn das ES-M32 als Programmiergerät eingesetzt werden soll, reichen diese beiden Grundeinstellungen aus, wobei man wählen kann ob mit oder ohne Quarz gearbeitet werden soll.



Ein Programm lädt man mit der Flash-Funktion. Ein Klick auf „HEX-Flash“ öffnet ein Datei-Menü. Nach der Auswahl einer Hex-Datei wird ihr Inhalt in das Textfenster geladen. Ein Balken zeigt den Verlauf des Programmiervorgangs. Am Ende erhält man eine ok-Meldung.

Programme werden wie die Fuses über die ISP-Schnittstelle des Mega32 gebrannt. Die Software verwendet dabei den FT232R auf der Platine im Bit-Bang-Modus wie eine parallele Schnittstelle. Die COM-Einstellung ist dabei ohne Bedeutung. Voraussetzung ist aber, dass kein zweiter FT232 am USB angeschlossen ist. Nach der Programmierung beendet das Programm den Bit-Bang-Modus. Die verwendeten Leitungen RI, DCD, DSR und CTS des FT232R sind danach wieder Eingänge. Die vier ISP-Jumper dürfen daher auch im Terminal-Betrieb gesetzt bleiben und stören die Kommunikation nicht.



Wenn ein Programm die serielle Schnittstelle verwendet lässt es sich mit der Terminal-Funktion direkt testen. Die virtuelle COM-Schnittstelle (z.B. COM2) wird nur einmal eingestellt und ist dann beim nächsten Start bekannt. Einstellbar sind Schnittstellen bis COM9. Welche Schnittstelle der Treiber erzeugt kann im Geräte-Manager nachgesehen und geändert werden. Alternativ zur virtuellen seriellen Schnittstelle über den FT232R kann auch eine reale Schnittstelle über den MAX232 verwendet werden, wenn die Jumper entsprechend gesteckt sind.

Auf der Registerkarte „Terminal“ befindet sich auch ein Auswahlfenster für die Übertragungsgeschwindigkeit bis 115200 Baud. Nach der Einstellung der Baudrate können Einzelbytes oder Textzeichen gesendet und empfangen werden.

