Online-Lehrgang statt Lehrbuch

Praxiskurs AVR-Programmierung in C

eLearning – eine neue Lehr- und Lernform hält nun auch in der Reihe „Lernpaket“ des Franzis-Verlages Einzug. Anhand eines Freeduino-AVR-Boards wird in einem Online-Lehrgang mit zahlreichen Praxisversuchen Schritt für Schritt das Wissen zur Programmierung eines AVR-Mikrocontrollers bis hin zu umfangreicheren Mess-, Steuer- und Regelanwendungen vermittelt.

Programmieren lernen

Will man heute in der Elektronik „mitreden“, gehört die Beschäftigung mit Mikrocontrollern zwingend dazu, denn kaum eine moderne elektronische Schaltung kommt ohne diese praktischen Bausteine aus. Sie fassen zahlreiche Funktionsgruppen, die früher als umfangreiche IC-Sammlung die Elektronikplatinen belegten, auf einem komplexen Chip zusammen. Die Verknüpfung dieser Funktionsgruppen zu einem funktionierenden Ablauf erfolgt über ein in den Controllerspeicher zu ladendes Programm.

Das zu schreiben, war früher etwas für Spezialisten, die bücherfüllende, stark hardwarebezogene Assemblerprogramme entwickelten, die von anderen kaum nachvollziehbar waren. Im Lauf der Zeit entwickelten sich immer mehr Hochsprachen, die zwar nicht so maschinennah und auch nicht immer so schnell arbeiten, aber einfacher zu programmieren sind, da sie den Programmierer von viel Routinearbeit entlasten. Eine auch für Hobby-Programmierer, die sich nicht mit der Assemblerprogrammierung befassen wollen, einfach beherrschbare Programmiersprache ist BASIC. Ein übersichtlicher Befehlssatz, schnell begreifbare Befehlssequenzen und eine einfache Struktur machten BASIC in den 1980er Jahren zur weit verbreiteten Standard-Programmiersprache unter Hobbyisten.

Damals ging es um das Programmieren kompletter Mikrorechner auf der Basis von Mikroprozessoren wie 68xxx oder Z80.

Heute haben derlei Rechner nur noch historische Bedeutung, ihre ursprünglichen Aufgaben übernehmen heute im Wesentlichen Personalcomputer.

Den Part „Steuern – Messen – Regeln“ hingegen, der früher ebenfalls von kleineren Mikroprozessorsystemen übernommen wurde, haben heute Ein-Chip-Mikrocontroller übernommen, die direkt in die Anwendungssysteme einbezogen sind, neudeutsch „embedded“.

Am weitesten verbreitet sind die PIC-Prozessoren von Microchip und die AVR-Controller von Atmel. Schritt für Schritt kommen heute ARM-Prozessoren hinzu, die schon in die Leistungsbereiche der „großen“ Mikroprozessoren hineinreichen. Insbesondere die AVR-Controller erfreuen sich auch bei Hobbyelek-

Bild 1: Die Startseite des Online-Kurses, links das übersichtliche Inhaltsverzeichnis

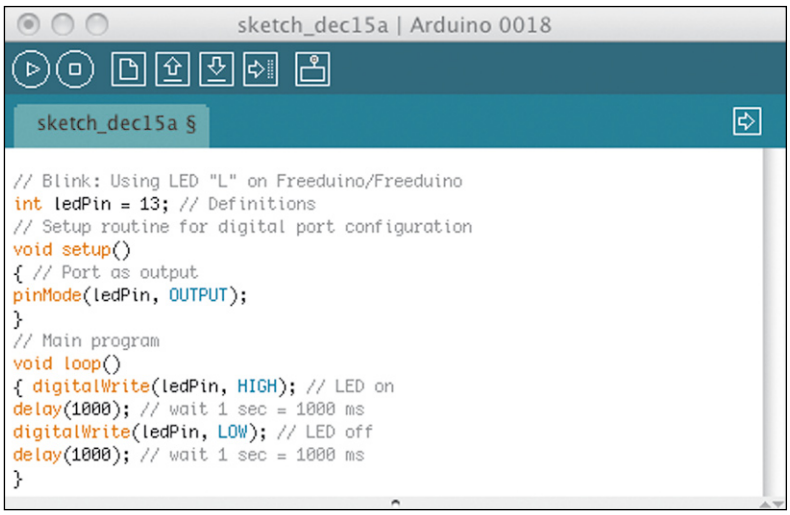

Bild 2: Die im Kurs verwendete Entwicklungsplattform von Arduino

Inhalt des Online-Lehrgangs

12 Lerneinheiten mit individueller Erfolgskontrolle:

1. Einstieg in die Hardware:
   Der AVR-Mikrocontroller und das Freeduino-Board
2. Physical Computing:
   Elektronische Bauelemente im Überblick
3. Die USB-Schnittstelle als Kommunikationsmedium zwischen PC und Controller
4. Die Programmier- und Entwicklungsumgebung
5. Einführung in die Programmierung mit „C“ und „Processing“
6. Programmierpraxis: Befehle, Variablen, Felder und mehr
7. Schnittstellenpraxis: Kommunikation mit Laptop und PC
8. Der Mikrocontroller in seinem Element:
   Messen, Steuern und Regeln
9. Richtig auf Unerwartetes reagieren: Interrupts
10. Wenn präzise Zeitmessung gefragt ist: Timer und Counter
11. Ganz ohne Peripherie geht es nicht:
    Schallwandler, Taster und Displays
12. Von der Applikation zum mikrocontrollerbasierten Gerät

tronikern großer Beliebtheit. Sie sind in einer großen Vielfalt am Markt, vom kleinen Mini-Controller für einfache Aufgaben bis hin zum hoch spezialisierten AVR, etwa mit integrierter LCD-Ansteuerung.

Auch diese Controller werden heute in den allermeisten Fällen in Hochsprachen programmiert. Dabei kommen meist die Programmiersprachen C und das ähnlich einfach wie BASIC strukturierte BASCOM zur Anwendung. Aus C entwickelten sich zahlreiche Derivate, so auch das hier in der Folge betrachtete „Processing“, das wir ja bereits bei der Vorstellung des Arduino-Systems in [1] ausführlich beschrieben haben.

Nun ist es trotz einfacher, begreifbarer Strukturen von Hochsprachen dennoch nicht einfach, eine Programmiersprache zu erlernen, zumal für Menschen, die sich bisher nicht mit dem Programmieren beschäftigt haben. In aller Regel kauft man sich ein entsprechendes Buch oder recherchiert im Internet nach Programmieranleitungen. Auch das ist nicht jedermanns Sache, denn bei Weitem nicht immer stellt sich hier anhand des „trocken“ dargebotenen Stoffs ein Lernerfolg ein. Mit der enormen Verbreitung der Mikrocontroller fällt es jedoch den Autoren heute leichter, ihr Buch mit einer käuflichen Hardwareplattform sowie einer Daten-CD bzw. einem Internet-Download zu unterstützen, so dass im Buch vorgestellte Lerninhalte sofort praktisch umsetzbar sind. Besonders verbreitet ist hier seit einigen Jahren die Arduino-Plattform.

Auf solchen Kombinationen von Lehrbuch und Hardwareplattform basieren die seit Jahren erfolgreichen Franzis-Lernpakete, die eine enorme Verbreitung erfahren haben.

Lesen, lernen, nachbauen

Nun gehen Franzis und ELV noch einen Schritt weiter und machen das Erlernen einer Programmiersprache noch einfacher. Der interaktive Online-Kurs „AVR-Mikrocontroller in C programmieren“ (ELV-Best.-Nr. JM-10 20 44) basiert wieder auf einem Lernpaket, das neben einem Arduino-Derivat, dem „Freeduino“-Board, zahlreiche Elektronikteile sowie ein kleines Steckboard zum lötfreien Aufbau der Peripherie zum Mikrocontroller enthält. Als Spannungsversorgung genügt zunächst allein der USB-PC-Anschluss. Somit erscheint das Lernpaket auf den ersten Blick wie eines der vielen anderen Lernpakete des Verlages. Nur – statt des sonst beiliegenden dicken Buches werden dem Käufer ein Link und ein Zugangscode mitgeteilt, mit denen er auf die Startseite des zugehörigen Online-Kurses (Bild 1) gelangt. Lässt man Cookies zu, muss man nicht bei jedem Neustart des Lehrgangs die Zugangsdaten erneut eingeben.

Hier beginnt die Reise durch die Welt der Mikrocontroller. In einem Flash-Player, der in allen gängigen Browsern läuft, bewegt man sich Schritt für Schritt durch einen für Einsteiger ausgelegten Kurs, der ständig von praktischen Beispielen flankiert ist, die man mit der mitgelieferten Hardware unmittelbar nachvollzieht. Programmiert und trainiert wird auf der zum Arduino gehörenden Entwicklungsplattform (Bild 2), die auf „Processing“ [2] basiert, einem C-Derivat, das sich hervorragend für den Einstieg in C einsetzen lässt. Zunächst wird nur das per USB an den PC oder

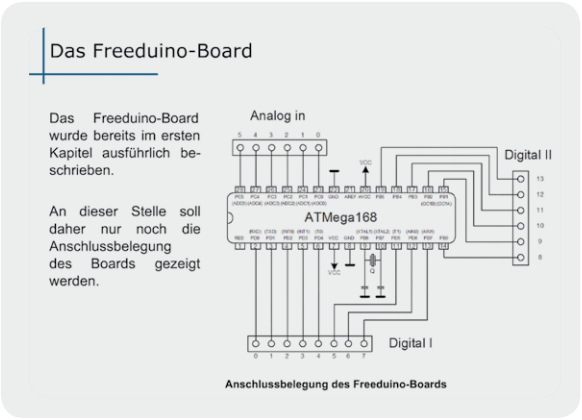

Bild 3: Alles detailliert erklärt – hier die zur Verfügung stehenden AVR-Ports

Mac anzuschließende Freeduino-Board benutzt, später kommt als Peripherie das Steckbrett mit den mitgelieferten Bauteilen zum Einsatz.

Jedes Kapitel des Lehrgangs wird, neben dem praktischen Übungsteil, der unmittelbar mit dem Einstieg in die Hardware beginnt, auch von einer interaktiven Überprüfung des Gelernten in Quizform begleitet. Wer dies aber überspringen will, kann dies tun und sofort zum nächsten Kapitel gehen. Man wird also nicht zur dennoch empfehlenswerten Klausur genötigt.

Zwölf Kapitel für den Erfolg

Dies ist der erste Online-Lehrgang des Verlages, weitere sind zum Ausbau der Programmierkenntnisse angekündigt. Gliederung und Aufbau sind klar und logisch, lediglich die Zugriffsgeschwindigkeit ist bei weniger schneller Hardware wie Netbooks oder Tablet-PCs nicht sehr hoch.

Über 400 Seiten lang kann man den Lehrgang Stück für Stück abarbeiten, jederzeit beliebig nachschlagen wie in einem Buch, unmittelbar Themen anspringen – eine neue Qualität des Lernens. Beendet man das Programm zwischenzeitlich, kann man beim nächsten Start unmittelbar wieder dort anfangen, wo man zuvor aufgehört hat – sehr praktisch.

Der gesamte Lehrgang ist in zwölf übersichtliche Kapitel aufgeteilt, ein ebenso übersichtliches (ausblendbares) Inhaltsverzeichnis macht die direkte Auswahl eines Themas einfach. Für die direkte Suche ist eine Suchfunktion vorhanden. Wird es notwendig, weitere Informationen oder z. B. die Entwicklungsumgebung aus dem Internet zu beziehen, sind die Texte durch Direkt-Links unterstützt.

Der Kurs wird in enger Kooperation zwischen den beiden Verlagshäusern ELV und Franzis angeboten, weshalb es auch im ELV-Forum einen speziell hierfür angelegten Diskussionsbereich gibt, bei dem es um das Thema Mikrocontroller-Programmierung gehen soll.

Die Programmcodes können unmittelbar aus den Programmfenstern, aber auch direkt aus einer Zusammenfassung auf den eigenen Computer geladen und in der Entwicklungsumgebung gestartet werden – man muss nichts mühsam abtippen und kann die Programme unmittelbar nach eigenem Gusto modifizieren, in eigene Programme einbinden usw.

Absolute Beginner

Genau dieser englische Begriff fällt dem Autor ein, wenn man den Kurs beginnt: er ist selbst für absolute Elektronik-Einsteiger geeignet. Nach einer kurzen Einführung in die Welt der Mikrocontroller und Mikroprozessoren wird das Freeduino-Board ausgepackt und Schritt für Schritt vorgestellt. Immer wieder wird mit interaktiven Buttons gearbeitet, so dass die Seiten übersichtlich bleiben und ein sofortiger Lerneffekt eintritt. Anschluss (Bild 3) und die verschiedenen Möglichkeiten der Spannungsversorgung werden detailliert erklärt und reichlich bebildert. Den Abschluss der Vorstellung des Freeduino-Boards und der internen Komponenten des verwendeten AVR-Controllers bildet der Hinweis auf das bei der Arbeit mit dem Controller unvermeidliche Datenblatt, das man immer auf dem Rechner parat haben sollte.

Anschließend geht es an den Inhalt des mitgelieferten Bauteilepakets. Schritt für Schritt werden, ohne zu viel theoretischen Ballast, alle Bauelemente vorgestellt und praktische Tipps zur Arbeit mit dem jeweiligen Bauteil gegeben.

Schließlich wird der USB-Port nebst einigen Grundlagen, auch zur Spannungsversorgung via USB, besprochen und der USB-Chip auf dem Freeduino-Port konfiguriert. Der benötigte USB-Treiber und weitere Programme können direkt aus dem Anhänge-Menü heruntergeladen werden.

Die Entwicklungsumgebung

Mit dem Herunterladen der Entwicklungsumgebung mit „Processing“ von der Arduino-Webseite (ebenfalls ohne langes Suchen per Direkt-Link) geht es nun an die ersten Schritte zur Programmierung auf dem eigenen Rechner. Der Empfehlung, nicht die neueste Version zu laden, sondern die Version 018, sollte gefolgt werden, damit es zu einem einwandfreien Ablauf der Beispielprogramme kommt.

Es geht los!

Nach der ausführlichen Einführung in die Entwicklungsumgebung geht es im nächsten Kapitel los mit der Einführung in die C-, d. h. „Processing“-Programmierung. Anhand eines praktischen Programmbeispiels werden

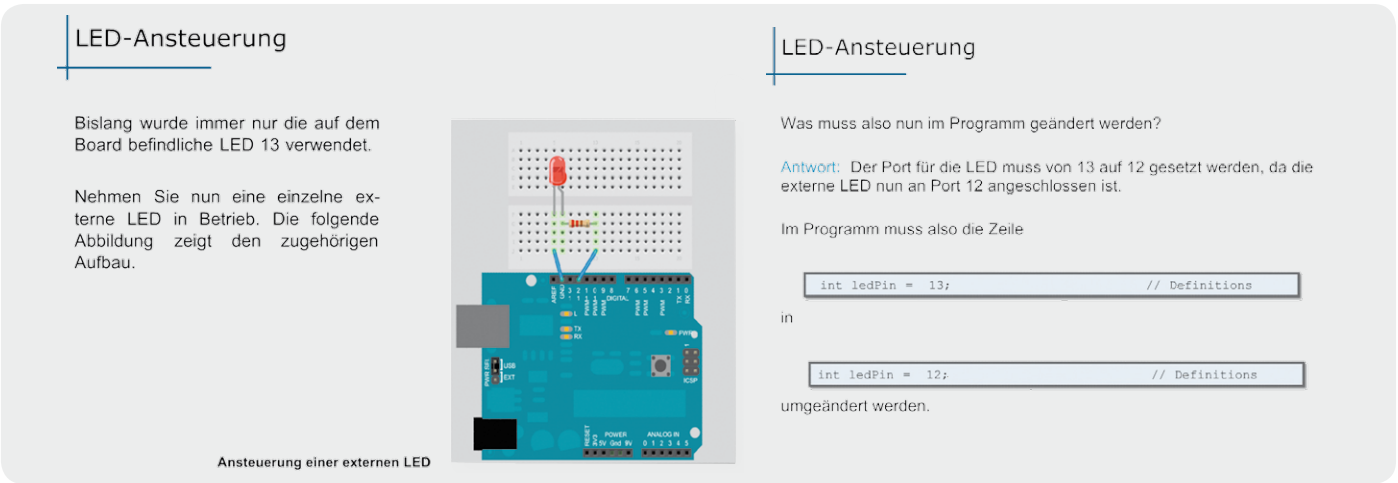

Bild 4: Immer mit praktischer Unterstützung – hier die erste einfache Übung zur Portauswahl für die Ausgabe

die ersten Schritte gemacht, ein Programm Zeile für Zeile, Befehl für Befehl analysiert und dabei Befehle und Strukturen eingehend behandelt. Danach folgt sofort das praktische Beispiel, indem das vorhandene Programm modifiziert und ein Experiment auf dem Steckbrett aufgebaut wird (Bild 4). Stets werden die praktischen Beispiele unterstützt mit Tipps, Praxishinweisen, ja sogar anschaulichen Videosequenzen.

Nach den ersten Erfolgen widmet sich der Kurs dann ausführlich und in verständlicher Form der Programmiersprache selbst. Durch die gute Gliederung kann man später immer wieder sofort auf einen bestimmten Begriff oder ein Syntaxelement zugreifen und nachlesen.

Auch hier geht es anschließend sofort an praktische Übungen – als erstes Projekt wird ein Multifunktions-LED-Fahrradlicht aufgebaut und eine Softwarelösung dazu entwickelt.

Das nächste Kapitel widmet sich ganz der Kommunikation zwischen Mikrocontroller und PC bis hin zur Fernsteuerung des AVRs durch den PC.

Ab Kapitel 8 wird es dann richtig interessant. Hier folgt das Kapitel „Messen, Steuern, Regeln“ in der Praxis. Von der Messwerterfassung bis hin zur Visualisierung und Steuerung auf dem PC-Bildschirm (Bild 5), alles wird Schritt für Schritt erklärt und gleichzeitig praktisch ausgeführt. Faszinierend, wie man z. B. mit ganz einfachen Mitteln und wenigen Programmzeilen einen hüpfenden Ball auf dem PC-Bildschirm per einfacher Handbewegung berührungslos dirigiert.

Den größten Teil des Kapitels 9 haben die Autoren des Kurses dem Thema „Interrupt“ gewidmet. Nicht ohne Grund, denn die Interrupt-Behandlung ist nicht ganz trivial und wird in vielen anderen Veröffentlichungen viel zu stiefmütterlich behandelt. Im Kapitel 10 schließlich geht es weiter mit Timern, PWM, Portbefehlen und der Ausgabe von Analogwerten.

Das Kapitel 11 ist Praxis pur. Hier entstehen schon komplette kleine Lösungen, von der elektronischen Grußkarte mit Melodie bis zur Nutzung des mitgelieferten Punktmatrix-Displays für Spiele, Zähler und sonstige Anzeigen (Bild 6).

Kapitel 12 schließlich rundet den Kurs ab – hier wird nach einem Exkurs zur USB-unabhängigen Spannungsversorgung des Freeduino eine Lösung zum Bau eines Digitalvoltmeters vorgestellt, nicht ohne dabei den ADC des Mikrocontrollers genau zu erläutern. Schließlich folgt die Loslösung des AVRs aus dem Entwicklungssystem. Es wird gezeigt, wie dieser innerhalb einer autonom arbeitenden Schaltung eingesetzt, mit Spannung und Takt versorgt und programmiert wird (Bild 7). Abschließend entsteht – allerdings wieder mit dem Freeduino-Board, damit es immer noch komplett ohne Löten abgeht – ein autarkes, komplettes Gerät, ein batteriebetriebenes Digitalthermometer im eigenen Gehäuse (Bild 8).

Der den Kurs abschließende Ausblick bringt es auf den Punkt: „Wenn Sie die 12 Kapitel dieses Kurses durchgearbeitet haben, dann sind Sie auf dem Gebiet der Mikrocontrollertechnik kein Anfänger mehr.“

Dem ist, außer dem Ausblick auf weitere Aufbaukurse zum Thema, nichts hinzuzufügen: ein Kurs, der Spaß macht, echten Lern- und Erfolgswillen weckt, perfekt praktisch begleitet ist – was will man mehr?

Gegenüber vergleichbaren Kursen anderer Anbieter stimmt hier auch der Preis – wo bekommt man für unter 100 Euro einen solchen Kurs samt Hardware? ELV

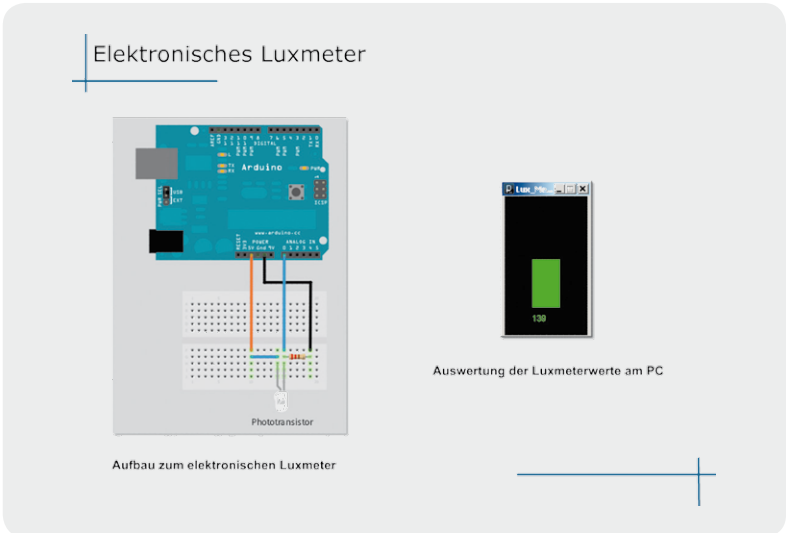

Bild 5: Der AVR erfasst, der PC visualisiert – das Experiment zum Luxmeter.

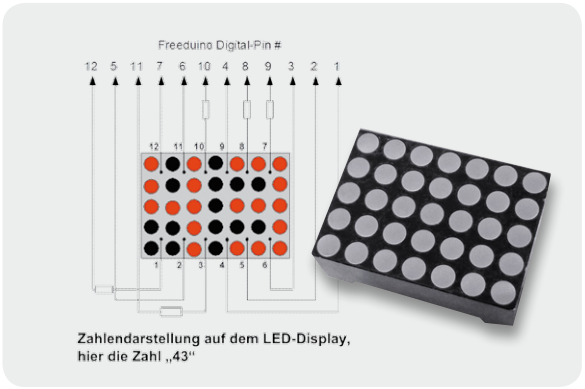

Bild 6: Vielseitig verwendbar – die mitgelieferte LED-Matrixanzeige

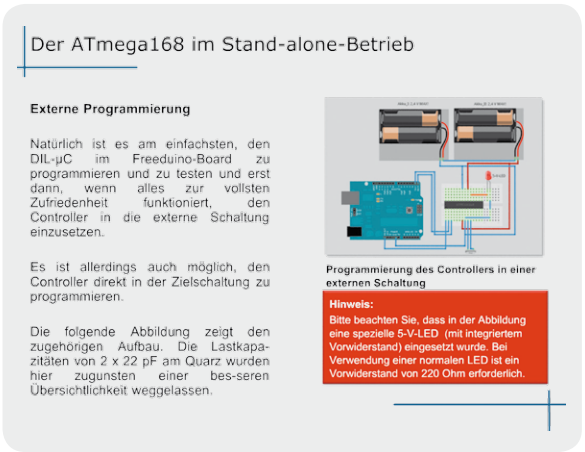

Bild 7: Losgelöst – die autarke Mikrocontrollerapplikation entsteht.

Bild 8: Höhe- und Schlusspunkt des Kurses – das erste komplette Gerät

Weitere Infos:

[1] Das Arduino-Projekt,
ELVjournal Dezember/Januar 2011, S. 88 ff.
[2] www.arduino.cc/playground/Interfacing/Processing