Stundenplan 3000

Der **Stundenplan 3000** ist eine Mikrocontroller Applikation und erfühlt die Aufgabe eines elektronischen Stundenplanes. Dieser Stundenplan kann für sechs Wochentage bis zu 14 Unterrichtsstunden pro Tag speichern und anzeigen. Alle Eingaben werden in dem Mikrocontroller dauerhaft gespeichert und gehen auch bei einem Batteriewechsel nicht verloren. Die Unterrichtsfächer können in einer Wochenübersicht oder in einer Tagesreihenfolge abgerufen werden. Zudem ist der Stundenplan mit einer automaischen Abschaltfunktion versehen, die aktiviert wird wenn er für ca. 3min nicht benutzt wird.

Konzept:

Dieses Elektronikprojekt ist für Schüler, Azubis und Elektronikeinsteiger konzipiert worden.

Es wurde als ein einfache und leicht verständliche Mikrocontroller Applikation entwickelt, die als Grundlage für folgende Themen dienen soll:

* Bauteile
  + Transitoren
  + Widerstände
  + LEDs
  + Tasten
  + Mikrocontroller
* Bestückung von Platinen
* Löten
* Einführung BASCOM AVR IDE
* Mikrocontroller Programmtechnik
  + Initialisierung
  + Interrupts
  + Multitasking
  + Multiplex Anzeige
  + Tastatur
  + Sleepmodus / Reset
  + EEprom

-Relevanter Anwendungsbereich

Bedienung:

Der Stundenplan wird über die Taste „**EIN**“ aktiviert. Es stehen zwei Anzeigemodi zu Verfügung. Der erste ist die Wochenübersicht. Hierbei werden die die Tage und die entsprechenden Fächer im Rotationverfahren („Mo“ bis „Sa“) für eine Sekunde angezeigt. Es spielt dabei keine Rolle ob ein Tag mit Unterrichtsfächern programmiert ist. Der Wochenmodus ist nach einem Tastendruck der Taste „EIN“ aktiviert.

Der zweite Anzeigemodus ist die Tagesansicht. Sie wird durch eine direkt Anwahl eines Tages **(„Mo“-„Sa“**) aktiviert. In der Tagesansicht wird nur der angewählte Tag und die entsprechenden Fächer angezeigt. Die Besonderheit ist jedoch, dass die Unterrichtstunden nun rotierend in der Reihenfolge in ihrer Programmierung angezeigt werden. Somit kann bei der Programmierung der Fächer auch die chronologische Reihenfolge der Unterrichtsstunden festgelegt werden.

Um die Unterrichtsstunden für einen Tag zu programmieren, muss der Stundenplan sich in der Tagesansicht befinden und der gewünschte Tag angewählt sein.

Über die Taste „**Prog./Speichern**“ wird der Programmiermodus aktiviert. Mit der Taste „**Fach**“ wird das gewünschte Fach ausgewählt. Die LED des entsprechenden Faches blinkt nun auf. Dabei spielt die Blinkweise der LED eine wichtige Rolle. Blink die LED nur kurz auf, ist dieses Fach nicht für die Tag gespeichert. Blinkt die LED lange auf mit einer kleinen Pause, ist dieses Fach für diesen Tag gespeichert. Um ein Fach zu speichern oder wieder zu löschen, wird die Taste „**Setzen/Löschen**“ benutzt.

Der Mikrocontroller merkt sich bei der Programmierung, in welcher Reihenfolge die Unterrichtsstunden angewählt werden und gibt diese Reihenfolge später bei der Tagesansicht wieder. Ein erneuter Tastendruck „**Prog./Speichern**“ schließt die Programmierung ab.

Wird die Programmierung ohne eine Anwahl abgeschlossen, wird der entsprechende Tag gelöscht.

Der Stundenplan verfügt über eine „Auto off“ – Funktion. Das heißt, er schaltet sich ca. 3min nach dem letzten Tastendruck aus.

Schaltungsbeschreibung:

Das Herz der Schaltung ist der Mikrocontroller IC1. Er steuert alle Funktionen, wie LED –Anzeige, Tastaturabfrage und die Speicherverwaltung.

Die Anzeige der Unterrichtsstunden, die Anzeige der Wochentage und die Tastaturabfrage sind in drei Blöcke geteilt und werden im Multiplexverfahren verarbeitet. Das heißt, es teilen sich mehre

LEDs und Taster ein Port vom Mikrocontroller. Damit der Mikrocontroller zwischen den LEDs und Tastern unterscheiden kann, benötigt er die Transistoren T1, T2 und T3. Diese Transistoren arbeiten wie ein Schalter. Wenn der Mikrocontroller z.B. den Transistor T2 aktiviert, werden die LEDs für die Wochentage (D15 - D20) mit der Masse verbunden. Bekommen nun die LED ein EIN-Signal vom Mikrocontroller, leuchten diese auf. Deaktiviert der Mikrocontroller wiederum den Transistor wird die Verbindung zum Masse unterbrochen, worauf die LEDs nicht mehr leuchten können.

Dabei wird für alle 10ms immer nur eine LED Gruppe eingeschaltet, entweder die Unterrichtsstunden (D1-14) oder die Wochentage (D15-D20). Dadurch werden die LEDs mit einer Frequenz von 50Hz geschaltet. Also wird jede leuchtende LED 50mal in der Sekunde ein- und ausgeschaltet. Da unsere Augen sehr träge sind, können wir dieses Ein- und Ausschalten nicht mehr erkennen und sehen nur noch ein dauerhaftes Leuchten der LEDs.

Bei den Tastern ist es etwas anders. Mit dem Transistor T1 aktiviert der Mikrocontroller die Tasten. Wird eine Taste gedrückt, wird der entsprechende Port des Mikrocontrollers gegen Masse gezogen. Dieses Low-Signal detektiert der Mikrocontroller als Tastendruck.

Damit der Mikrocontroller alle Funktionen und Aufgaben bewältigen kann, ist das Programm als Multitasking Funktion aufgebaut. Zwar ist dieses Multitasking aufgrund der Übersichtlichkeit sehr einfach gehalten worden, aber es zeigt die Grundzüge einer Multitasking Programmstruktur auf.

Es laufen also mehrere Programmprozesse gleichzeitig unabhängig voneinander. Wie zum Beispiel die Tastaturabfrage, LED Anzeige oder Zeitfunktionen.

Die Widerstände R1 –R14 sind Vorwiderstände für die LEDs. Sie tragen dafür Sorge, dass die LEDs nicht zu viel Spannung bekommen und ihr Betriebsstrom begrenzt wird. Da gerade bei einer batteriebetriebenen Schaltung das Strommanagement eine wichtige Rolle spielt, sind die Dioden als LOW-Current Dioden ausgelegt. Diese Sorte Leuchtdioden benötigen nur 2mA Betriebsstrom.

Ein weiterer Trick ist um bei den Leuchtdioden noch mehr Strom zu sparen, ist sie gepulst (hier mit 50Hz) zu betreiben.

Die Formel für ein LED Vorwiderstand ist recht einfach.

Eine wichtige Rolle hat der Taster „EIN“ (S10). Der Stundenplan ist eigentlich nie aus, er ist immer an und der Taster „EIN“ ist eigentlich auch kein richtiger EIN –Schalter. Mit dem Taster S10 wird an dem Mikrocontroller ein Reset ausgelöst. Das heißt, egal was der Mikrocontroller auch macht, bei einem RESET fängt der Mikrocontroller immer wieder mit allem neu an.

Zuerst sortiert sich der Mikrocontroller neu. Er bringt alle Ports in einem definierten Zustand, löscht Speicher und Register. Da fängt er mit dem ersten Programmschritt an.

Bei einem Auto Off wird der Stundenplan wie bereits angedeutet nicht richtig ausgeschaltet. Zuerst schaltet der Mikrocontroller alle LEDs und die Tasten ab. Dann versetzt sich der Mikrocontroller in einem Sleep –Modus. In diesem Sleep –Modus verbraucht der Mikrocontroller (also auch der Stundenplan) nur 3µA Strom, also 0.000003 Ampere. Es würde Jahre dauern bis in diesem Modus die Batterien aufgebraucht wären. Ist der Stundenplan aktiv, beträgt die Stromausnahmen auch nur maximal 15mA.

Durch einen RESET mit der Taste „EIN“ (S10) wird der Mikrocontroller „aufgeweckt“ und somit der Stundenplan aktiviert.

Bauteile:

**Menge Artikel Pos**

20 Stk. Low Current LED 5mm D1 – D20

3 Stk. BC547 T1, T2, T3

14 Stk. 470Ω, 1/4W R1 – R14

4 Stk. 100kΩ, 1/4W R15, R17, R19, R21

3 Stk. 10kΩ, 1/4W R16, R18, R20

1 Stk. ATMEGA8 IC1

1 Stk. 100nF C1

1 Stk. Stiftleiste 1x6 K1 + K2

10 Stk. Miniaturtaster S1 – S10

2 Stk. Mignon AA 1,5V Bat 1, Bat 2

