

Frank Scholl, März 2010
Dokumentation für Schaltungsversion 1.2

Der TeaTimer

Ein pfiffiger Helfer, der auch vergesslichen Zeitgenossen einen richtig gebrühten Tee garantiert.



Wer kennt nicht die Situation: man macht sich eine Tasse heißes Wasser und gibt einen Teebeutel hinein. Einige Minuten warten und – bis man zurück kommt hat der Teebeutel vielleicht 20, 30 Minuten oder noch länger gebadet. Und der Tee? Manche Teesorten verlangen ja eine längere Brühzeit, aber eigentlich sollte der Beutel pünktlich raus.

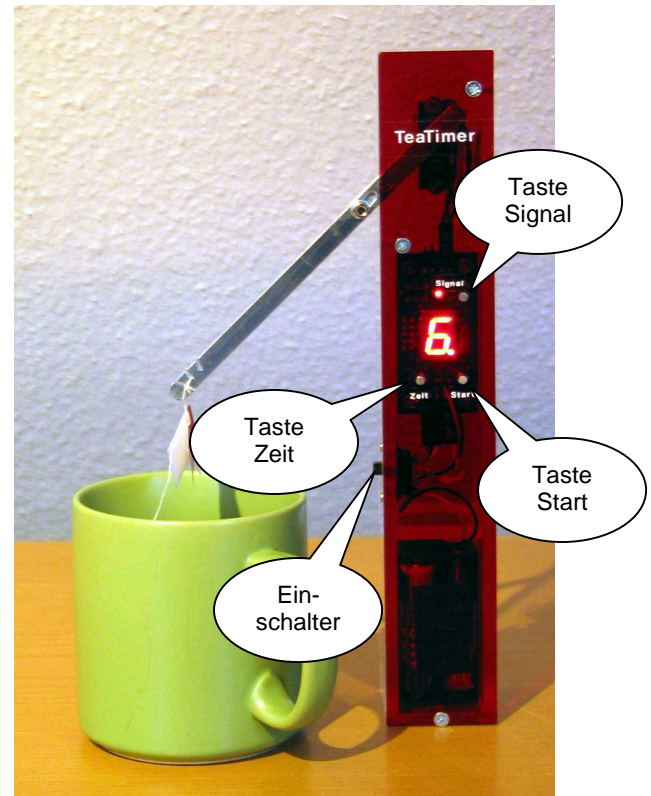
Also muss ein Timer her. Ich habe da schon mehrere Varianten ausprobiert. Leider bin ich mit keiner davon glücklich geworden. Alle Wecker, die einen Alarm auslösen wollen ja gehört werden. Meist habe ich zu diesem Zeitpunkt meinen Schreibtisch im Büro kurz verlassen und der Alarm blieb ungehört.

Also musste eine Lösung her, die sich auch ohne mein Zutun um den Tee kümmert. Einfach zu bedienen sollte sie sein, einfach zu bauen, robust und zuverlässig! Und hier ist er: der TeaTimer.

Das Prinzip ist sehr einfach: eine batteriebetriebene Elektronik steuert ein einfaches Modellbau-Servo, an dem ein Arm zum Schwenken des Teebeutels befestigt wird. Mit einem Knopf stellt man die gewünschte Brühzeit zwischen 1 und 9 Minuten ein und mit einem weiteren Knopf startet man den Vorgang. Danach kann man das Geschehen sich selbst über lassen. Rechtzeitig wird der Teebeutel aus der Tasse gezogen, egal was man inzwischen angefangen hat.

Bedienung

- Einen Teebeutel am Schwenkarm befestigen und eine Tasse mit heißem Wasser darunter stellen. Der Teebeutel sollte etwa 2cm über dem Wasserspiegel hängen.
- Einschalten.
- Der TeaTimer zeigt die zuletzt gewählte Brühzeit in Minuten an.
- Mit der Taste „Zeit“ kann die Brühzeit geändert werden (1 bis 9 Minuten).
- Mit der Taste „Signal“ kann das akustische Signal am Ende des Brühvorgangs ein- oder ausgeschaltet werden. Wenn die LED leuchtet ist das Signal aktiviert. Die Aktivierung kann jederzeit aus- oder eingeschaltet werden. Die Einstellung bleibt im Gerät auch nach Ausschalten gespeichert.
- Mit der Taste „Start“ beginnt der Brühvorgang: der Teebeutel wird in die Tasse getaucht. Nach 20 Sekunden und dann nach jeder weiteren Minute wird der Beutel geschwenkt. Durch erneutes Drücken der Taste „Start“ kann der Brühvorgang jederzeit beendet werden.
- Am Ende des Brühvorgangs wird der Teebeutel aus der Tasse gehoben und, falls „Signal“ aktiviert ist, eine Tonfolge über den Lautsprecher abgespielt. Den Teebeutel nicht ausdrücken sondern nur abtropfen lassen! Dadurch bleiben die Bitterstoffe im Beutel zurück und der Tee behält sein volles Aroma. Auf dem Display läuft eine Schlange durch. Durch Druck auf die Start-Taste wird die gewählte Brühzeit wieder angezeigt und eine neue Tasse kann gebrüht werden. Sollte man den Tee vergessen haben, ertönt jede weitere Minute das Tonsignal (falls aktiviert).



Achtung:

Bitte den TeaTimer nicht aus- und wieder einschalten, während der Teebeutel im Tee eingetaucht ist! Beim Wiedereinschalten fährt der Hebelarm sofort in die oberste Position. Bei einem schnellen Servo wird der Tee dann in die weitere Umgebung verspritzt...

Ändern des Tonsignals

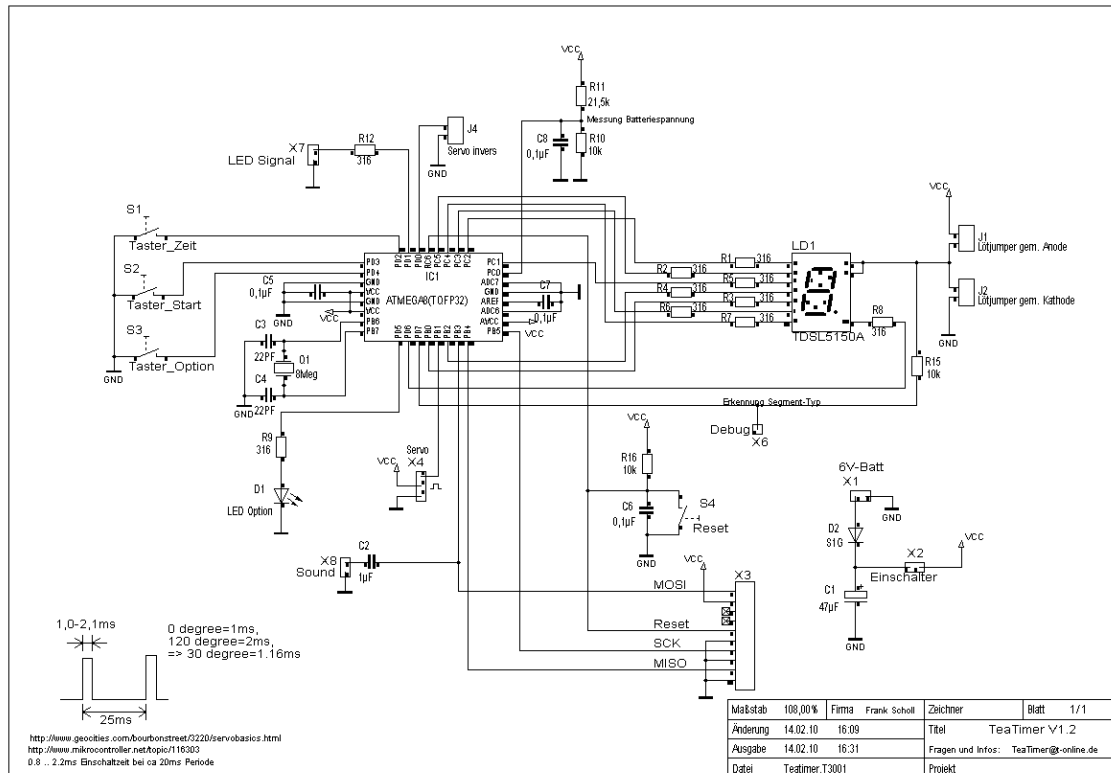
Wenn die Taste „Signal“ beim Einschalten gedrückt gehalten wird, ertönt wiederholt die Signal-Tonfolge. Alternativ kann man auch nach dem Einschalten die Taste lange gedrückt halten um in diese Signalauswahl zu gelangen.

Mit der Signaltaste kann man jetzt eine andere Tonfolge auswählen. Die Nummer der Tonfolge wird auf dem Display angezeigt. Hat man eine gefällige Tonfolge gefunden, kann man sie mit der Taste „Start“ übernehmen. Die Einstellung bleibt gespeichert.

Abfrage der Software-Version:

Hält man die Taste „Start“ beim Einschalten gedrückt, wird die Software-Version ziffernweise ausgegeben. Die Ausgabe beginnt mit „U“ für „Version“ gefolgt von einem Datumscode JJMMTT, also z.B. „U – 0 – 9 – 1 – 1 – 1 – 0“ für die Version vom 10.11.2009.

Die Schaltung



Schaltplan

Die Schaltung besteht im wesentlichen aus einem ATMEL ATmega8, der mit einem 8MHz-Quarz getaktet wird (die Zeit soll ja genau sein!), einer 7-Segment-Anzeige und 3 Tasten. Und natürlich dem Stecker für das Servo.

Es können sowohl Displays mit gemeinsamer Anode als auch mit gemeinsamer Kathode (empfohlen) verwendet werden. Über die Löt-Jumper J1 bzw. J2 wird die Anode bzw. Kathode angeschlossen. Über R15 erkennt der Controller nach dem Einschalten den gewählten Typ und steuert das Display entsprechend. Nach der Erkennung kann der Port PD7 zum debuggen auf Ausgang geschaltet werden. Damit kann man Laufzeiten messen o.ä. An X4 wird ein analoges Modellbau-Servo angeschlossen, das den Hebelarm für den Teebeutel bewegt. Das Servo erhält seine Positions-Information über den 16-Bit-PWM-Ausgang an PB1. Die Pulsweite zwischen 1 und 2 Millisekunden bestimmt die Winkelposition.

Hier sind 2 Dinge zu beachten:

- Die Polung des Steckers stimmt u.U. nicht mit der des Servos überein. Ggf. muss im Plus und Minus im Stecker getauscht werden!
- Es gibt „rechtsdrehende“ und „linksdrehende“ Servos.
Über den Löt-Jumper J4 (der beim Programmstart eingelesen wird) kann die Drehrichtung umgekehrt werden. Normalerweise ist J4 geschlossen.

Die 3 Tasten schalten einfach GND auf die Portpins, im Controller werden die Pullup-Widerstände aktiviert.

Über C2 wird ein externer Lautsprecher angeschlossen. Die Tonfrequenzen werden über PWM des Port PB3 ausgegeben.

Mit R10 und R11 ist die Messung der Batteriespannung möglich.

Über X7 kann eine externe LED angeschlossen werden. Die LED dimmt ständig auf und ab und kann für Beleuchtungseffekte verwendet werden.

Der Reset-Taster ist eigentlich nicht notwendig, hilft aber bei der Software-Entwicklung.

Über X3 wird der Controller programmiert. Wenn man die Software nur einmal reinladen will, braucht man den Stecker nicht einzubauen. Einfach während des Downloads dranhaltend geht auch und die Platine sieht dann besser aus.

Die Batterie ist ständig über D2 an C1 angeschlossen. Durch einen Schiebeschalter an X2 ist der Spannungsanstieg beim Einschalten dann sehr steil, weil der Kondensator nicht erst geladen werden muss. Bei schlechten Batterien könnte das beim Resetverhalten von Vorteil sein.

Elektrische Stückliste Teatimer_V1.2 (PCB) discoverversion								
Datum	07.03.2010							
Anzahl	Name	Wert / Bezeichnung	Gehäuseform	Bestückseite	Einzelpreis	Summe	Lieferant z.B.	Artikel-Nr.
Basisversion (ohne Lautsprecher):					<i>Kosten der Baugruppe gesamt:</i>	16,00 €		
1	C1	47µF	2817_ELKO	oben	0,21 €	0,21 €	Reichelt	SMD TAN 47/10
4	C5, C6, C7, C8	0,1µF	0805	unten	0,05 €	0,20 €	Reichelt	X7R-G0805 100N
2	C3, C4	22PF	0805	unten	0,05 €	0,10 €	Reichelt	NPO-G0805 22P
8	R1, R2, R3, R4, R5, R6, R7, R8	316R	1206	oben	0,01 €	0,08 €	Pollin	Best Nr. 220 679
3	R10, R15, R16	10k	0805	unten	0,10 €	0,31 €	Reichelt	SMD-0805 10.0K
1	R11	22k	0805	unten	0,10 €	0,10 €	Reichelt	SMD-0805 22.0K
1	IC1	ATMEGA8(TQFP32)	TQFP32(7X7)	unten	1,45 €	1,45 €	Reichelt	ATMEGA 8-16 TQ
1	D2	S1G (S1M)	DO214	unten	0,02 €	0,02 €	Pollin	Best Nr. 140 441
1	Q1	8Meg	HC49/U	oben	0,18 €	0,18 €	Reichelt	8.0000-HC49U-S
1	LD1	TDSL5150A (J2 schließen!)	TDSL5150	oben	0,54 €	0,54 €	Reichelt	SC 52-11 RT
1	S1	Taster_Zeit	TASTER_KURZHUB	oben	0,14 €	0,14 €	Reichelt	TASTER 9305
1	S2	Taster_Start	TASTER_KURZHUB	oben	0,14 €	0,14 €	Reichelt	TASTER 9305
1	X1	Batterieclip für 9-Volt-Block, T-Form			0,14 €	0,14 €	Reichelt	CLIP 9V-T
1	X2	Einschalter	1X02	oben	0,01 €	0,01 €	Reichelt	SL 1X40W 2.54
1	X4	Servo-Anschluss	1X03	oben	0,02 €	0,02 €	Reichelt	SL 1X40W 2.54
1	nicht auf Leiterplatte	Schiebeschalter-Miniatur, Lötanschluß, 2x UM			0,26 €	0,26 €	Reichelt	T 219
1	nicht auf Leiterplatte	Halter für 4 Mignonzellen (AA), Druckknopf			0,35 €	0,35 €	Reichelt	HALTER 4XUM3-QDK
1	nicht auf Leiterplatte	Leiterplatte "TeaTimer V1.1"			6,00 €	6,00 €	Leiterplattenhersteller	
1	nicht auf Leiterplatte	Servo (analog)	Pinbelegung beachten !		4,95 €	4,95 €	Conrad	z.B. model craft ES-05
4	nicht auf Leiterplatte	Batterien AA (Mignon)			0,20 €	0,80 €	ALDI	
optionale Zusätze für Lautsprecher:					<i>Kosten der Baugruppe gesamt:</i>	0,88 €		
1	C2	1µF	1206	unten	0,21 €	0,21 €	Reichelt	X7R-G1206 1/100
1	R9	316R	1206	unten	0,01 €	0,01 €	Pollin	Best Nr. 220 679
1	D1	LED Option	LED_3MM_ROT	oben	0,06 €	0,06 €	Reichelt	LED 3MM RT
1	X8	Sound	1X02	oben	0,01 €	0,01 €	Reichelt	SL 1X40W 2.54
1	S3	Taster_Option (Sound)	TASTER_KURZHUB	oben	0,16 €	0,14 €	Reichelt	TASTER 9305
1	nicht auf Leiterplatte	Miniatur-Lautsprecher			0,45 €	0,45 €	Pollin	8003 (Pollin)
optionale Zusätze für externe LED:					<i>Kosten der Baugruppe gesamt:</i>	0,02 €		
1	R12	330R	1206	unten	0,01 €	0,01 €	Pollin	Best Nr. 220 679
1	X7	Anschluss ext. LED	1X02	oben	0,01 €	0,01 €	Reichelt	SL 1X40W 2.54
optionale Zusätze für Entwicklerversion:					<i>Kosten der Baugruppe gesamt:</i>	0,25 €		
1	X3	Wannenstecker_10_liegend		oben	0,14 €	0,14 €	Reichelt	SL 2X10W 2.54
1	X6	Debugpin	1X01	oben	0,00 €	0,00 €	Reichelt	SL 1X40G 2.54
1	S4	Reset-Taste	TASTER_KURZHUB	oben	0,11 €	0,11 €	Reichelt	TASTER 3301
außerdem								
Litzen und Buchsen zum Anschluss an die Stecker X1, X2, ...								

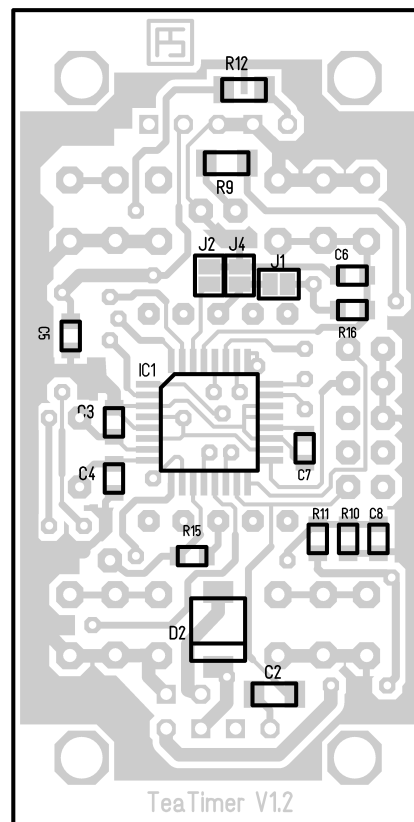
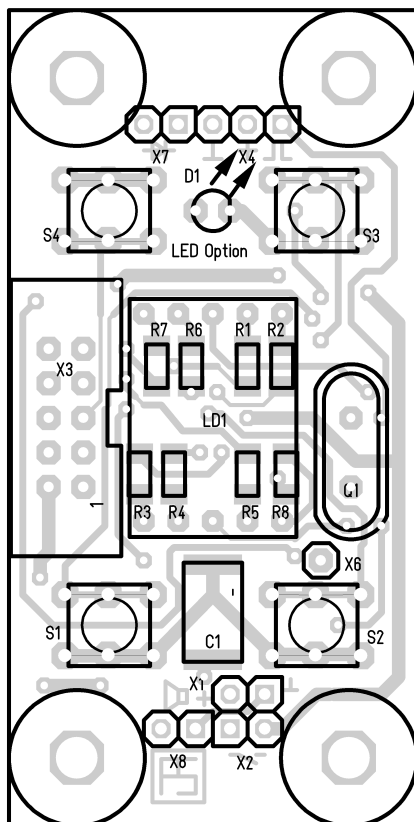
Stückliste

Die optionalen Schaltungsteile können nach Belieben ergänzt werden. I.d.R. reichen die Bereiche Basisversion und Lautsprecher.

Aufbau der Platine, Optionen

Zum Bestücken der Leiterplatte ist schon ein wenig Lötterfahrung erforderlich. SMD-Bauteile sind nichts für Anfänger. Besonderes Geschick und eine feine Lötspitze erfordert der Controller. Am besten positioniert man den Controller möglichst exakt auf dem Land und hält ihn mit einem Finger fest. Dann erwärmt man 2 diagonal liegende Pins vorsichtig (noch kein Lötzinn verwenden!) Der Controller ist dann erst mal angeheftet. Jetzt mit feinem Lötzinn nacheinander alle Pins kurz (!) benetzen.

Bei der Bestückung fängt man mit den niedrigsten Bauteilen an. Das Display sitzt direkt über den Widerständen – also das **Display zuletzt bestücken!**



Zur Inbetriebnahme lädt man die Software (Hexfile) in den Controller. Vielleicht noch mal kontrollieren ob die Jumper (Lötbrücken) J1, J2 oder J4 richtig gesetzt sind.

Wenn das Display nach dem Einschalten eine „3“ anzeigt und der Punkt im Sekunden-Rhythmus blinkt funktioniert schon mal das wichtigste.

Mit S1 sollten sich nun die Ziffern 1 bis 9 durchzählen lassen.

Mit S3 sollte die LED D1 aus- und eingeschaltet werden können.

Ein angeschlossenes Servo sollte über S2 ein Lebenszeichen von sich geben.

Bei Abbruch (S2 erneut drücken) sollte das Servo zurückfahren und (sofern die LED D1 leuchtet) ein Signalton aus dem Lautsprecher ertönen.

Download der SW

Zum Download des Hex-Files eignet sich z.B. PonyProg, das nur ein einfaches Interface benötigt [2].

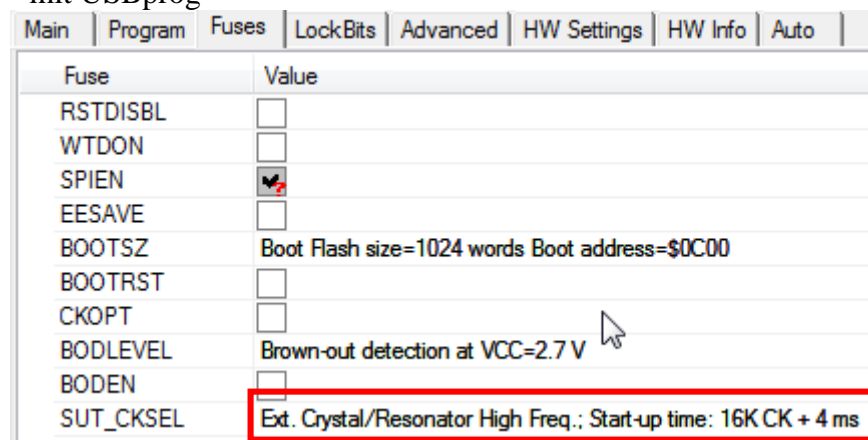
Komfortabler, dafür aber auch teurer ist z.B. USBprog [3], mit dem man direkt aus dem AVR-Studio [4] downloaden kann.

Damit die Software überhaupt läuft, müssen zunächst die Fusebits eingestellt werden. Ein fabrikneuer ATMEL-Controller läuft mit einem internen Takt von 1MHz, wir brauchen aber den externen 8MHz-Quarz.

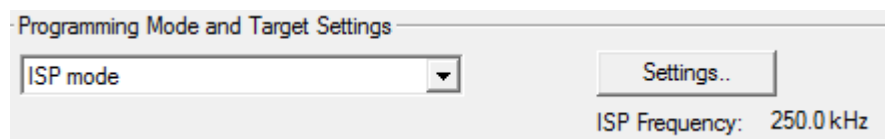
Fusebits:

- in Ponyprog alle Clksel deaktiviert = 1111 = ext. Quarz

- mit USBprog



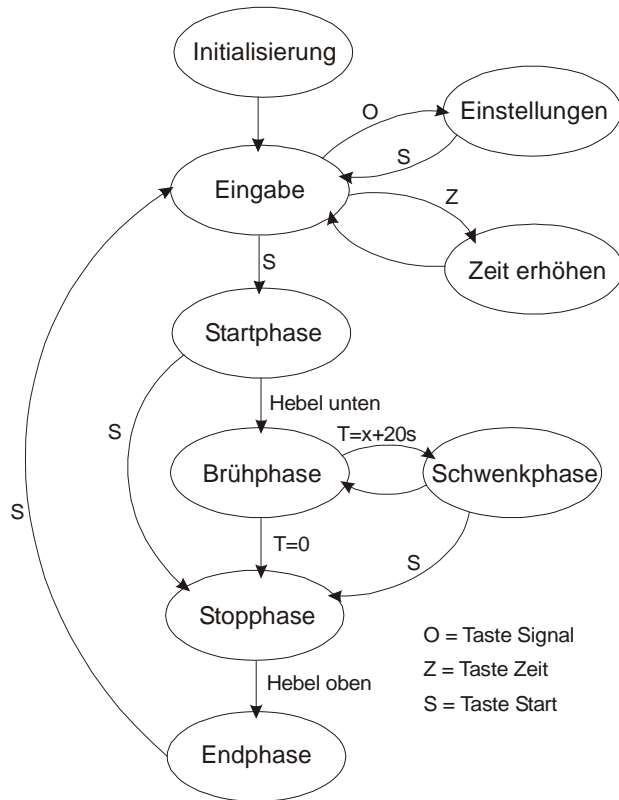
Wenn AVRISP mkII (=USBprog) keine Kontakt zum Controller aufbauen kann, liegt es meist daran, dass ein neuer (unprogrammierter) ATmega8 mit 1MHz intern läuft, die Programmierung muss aber 1/4 der Taktfrequenz sein! Also in den Optionen auf 250kHz einstellen:



Software

Auf die Details der Software möchte ich hier nicht eingehen. Wen es interessiert kann den kommentierten Quelltext einsehen. Hier nur der grobe Aufbau:

Im Hauptprogramm läuft eine State machine, deren Zustände sind im folgenden Diagramm zu sehen:



Im Hintergrund laufen 3 Timer-Interrupts.

Der wichtigste ist der „16-Bit-Timer1 Overflow Interrupt ISR“. Er wird alle ausgeführt und dient in erster Linie zur Zeitmessung, Berechnung Servoposition, Tastenabfrage und Messung Batteriespannung.

Mit Timer0 wird die LED-PWM gesteuert. Er ist nur aktiv, wenn LED-PWM aktiv ist.

```

|<----- var.us ----->|<----- (10ms-var.us) ----->|
|                                     +--> Timer0 OVR : alle Ziffern aus
+--> Timer0 OVR : alle Ziffern an
  
```

Timer1 (16bit, auf-/abwärts, 10bit prescaler, 2x output compare (output PB1, PB2), 1x input capture (PB0), CTC)

```

|<----- 20ms ----->|
|           +--> 5ms OC1B: 2.Ziffer aktivieren
+--> Zähler-Start bei 0
  
```

Timer2 steuert die Sound-Ausgabe über output compare PB3 (CTC mode).

Gehäusevorschlag

Den Ideen für den mechanischen Aufbau des TeaTimers sind natürlich keine Grenzen gesetzt. Als Anhaltspunkt rate ich lediglich, die Höhe der Servo-Achse auf ca. 22cm und die Länge des Teebeutel-Hebels auf ca. 12cm festzulegen. Damit kann man auch eine größere Tasse darunter stellen, bei der der Teebeutel vollständig eingetaucht und wieder herausgezogen wird.

Im einfachsten Fall müssen Servo und Leiterplatte lediglich an einem Holm mit einem Standfuß befestigt werden. Wirkt aber sehr asketisch...

Im folgenden beschreibe ich den Aufbau eines Plexiglasgehäuses, das m.A. sehr edel aussieht. In dieses Gehäuse passt ein Miniservo mit ca. 22,5 x 13 x 23mm Körpermaß, z.B. „modelcraft ES-05“ oder auch „robbe No.8503“. Für größere Servos müssen die Seitenteile und Mittelstege entsprechend angepasst werden.

Ich verwende glasklares Plexiglas mit 5mm Stärke für alle Teile außer der Front. Dafür habe ich 3mm transparent rot eingesetzt.

Im folgenden wird immer wieder erwähnt, die Kanten zu polieren. Hier eine kurze Beschreibung, wie ich dabei vorgehe:

Die zu polierende Kante wird zunächst mit einer Abziehklinge von dem Spuren des Bandschleifers befreit. Ein Teppichmesser tut's auch. Anschließend mit Schwingschleifer in 3 Stufen mit Körnungen 240/320 – 400 – 600 schleifen. Man sollte mindestens die letzte Stufe nass schleifen um Mini-Spannungsrisse zu vermeiden (ich hatte nach dem Kleben tatsächlich welche – sieht hässlich aus). Schließlich poliert man mit der Schwabbelscheibe und etwas Poliermittel nach: die Kante ist nun klar wie Wasser!

Grundsätzlich sollten die Spuren des vorangegangenen Arbeitsgangs nicht mehr zu sehen sein!

Zum Verkleben verwende ich „ACRIFIX® 1R 0192“. Der Kleber härtet bei Lichteinfluss glasklar aus. Beim Umgang ist darauf zu achten, dass alle Klebeflächen poliert sind. Mit unpolierten Flächen habe ich keine guten Erfahrungen gemacht. Auch bei der Dosierung ist Fingerspitzengefühl angebracht: zu viel Kleber quillt heraus und zu wenig hinterlässt sichtbare Lufteinschlüsse. Ein Tropfen daneben erzeugt Spuren, die nicht mehr zu beseitigen sind.

1. Die Teile des Gehäuses werden nach der Zeichnung „TeaTimer_Gehäuse_Servo-robbe-8503_Einzelteile.pdf“ mit der Stichsäge oder Kreissäge ausgesägt. Dabei sollte keine allzu hohe Schnittgeschwindigkeit gewählt werden, damit der Kunststoff hinter dem Sägeblatt nicht wieder zusammenschweißt. Detailliertere Verarbeitungshinweise siehe [1].
Hier sind die Maße noch etwas großzügig zu wählen, damit man später noch Fleisch für passgenaues Schleifen hat.
2. Nun die beiden Seitenteile, die Träger des Servos und der Platine, sowie der Batteriefachdeckel mit dem Bandschleifer auf Zeichnungsmaß bringen.
3. Am linken Seitenteil die Ausschnitte für den Hebelarm und den Einschalter samt Bohrungen anbringen.
4. In die Träger die Bohrungen und Gewinde M3 / M2 einbringen.
5. Nächster Schritt: zusammenkleben der Seitenteile mit den Mittelstegen.
Bei mehrteiligen Mittelstegen muss man sich zur genauen Positionierung eine Hilfsplatte erstellen, die die Stege auf Servo- und Platinenmaß zusammen hält, sonst ist das ganze eine zu wackelige Angelegenheit. Der Servospalt sollte so groß sein, dass

man das Servo auch wirklich einsetzen kann (vorher mal ausprobieren!). Da das Kabel unten am Servo herauskommt, muss man es durch die erweiterte Öffnung an der Seite schieben oder das Servo „einkippen“.

Vor dem zusammenkleben werden alle (!) Kanten der Stege poliert.

Das exakte zusammenkleben ist bei diesem Schritt besonders wichtig. Ein schiefer Grundkörper ist später nur schwer zu korrigieren.

6. Ist die Konstruktion fest, wird die Rückseite auf dem Bandschleifer ggf. nochmals plan geschliffen bis sie nicht mehr kippelt, wenn man sie auf eine ebene Fläche legt. Ansonsten nur die Kleberüberschüsse entfernen und die gesamte Rückseite polieren. Die Rückwand wird auf Maß geschliffen, und ggf. die Schallöffnungsbohrungen für den Lautsprecher angebracht. Die Seiten der Rückwand werden poliert und auf die Mittelkonstruktion geklebt.
7. Nach Aushärten des Klebers Ober- und Unterseite des Grundkörpers auf dem Bandschleifer bearbeiten bis alles senkrecht steht und nicht mehr kippelt. Danach die Kanten polieren. Deckel (mit Einschnitt für Hebel) und Boden auf Maß schleifen, alle Kanten polieren und aufkleben.
8. Den Lautsprecher (falls verwendet) mit bereits angelötetem Kabel durch 4 Klebstoffpunkte an der Rückwand befestigen.
9. Frontplatte auf Maß schleifen und Kanten polieren.
Die Frontplatte auf die Mittelkonstruktion legen und die 3 Löcher zur Verschraubung mit 2,5mm durch die Frontplatte in die Seiten des Grundkörpers bohren. Die Löcher der Frontplatte werden auf 3mm aufgebohrt und gesenkt, in den Grundkörper werden Gewinde M3 geschnitten.
10. Jetzt kann man die bestückte Platine mit Abstandshaltern anschrauben und durch die transparente Frontplatte die Bohrungen für die Taster anzeichnen. Ich bohre die Löcher 3mm und öffne sie dann mit einer Schlüsselfeile soweit, dass abgesägte Nieten, die als Verlängerung zu den Tastern dienen, mit wenig Spiel reinfallen. Die Verlängerung hat den Vorteil, dass auch nicht hundertprozentig positionierte Löcher nicht zu klemmenden Tasten führen.
11. Nun noch den Hebel nach Zeichnung anfertigen.
Zum besseren Transport kann der 2-teilige Hebel abgewinkelt werden.
Ein einteiliger Hebel ist dafür leichter herzustellen.
12. Die Beschriftung der Frontplatte klappt sehr gut mit weißen Rubbelbuchstaben. Zum Schutz klebt man einen Streifen Tesa darüber.

Nun noch alles zusammenbauen und erst mal eine Tasse Tee genießen... 😊

Quellen:

[1] Verarbeitungshinweise für Plexiglas

http://www.plexiglas-shop.com/DE/de/man_info/verarbeitungshinweise-fnbc65zznu.htm

<http://www.selbst.de> (→ suche „Plexiglas“)

[2] PonyProg serial device programmer

<http://www.lancos.com/prog.html>

[3] USBprog

<http://www.embedded-projects.net/usbprog>

[4] AVR Studio 4

http://www.atmel.com/dyn/Products/tools_card.asp?tool_id=2725