

PULSDETEKTOR BT2

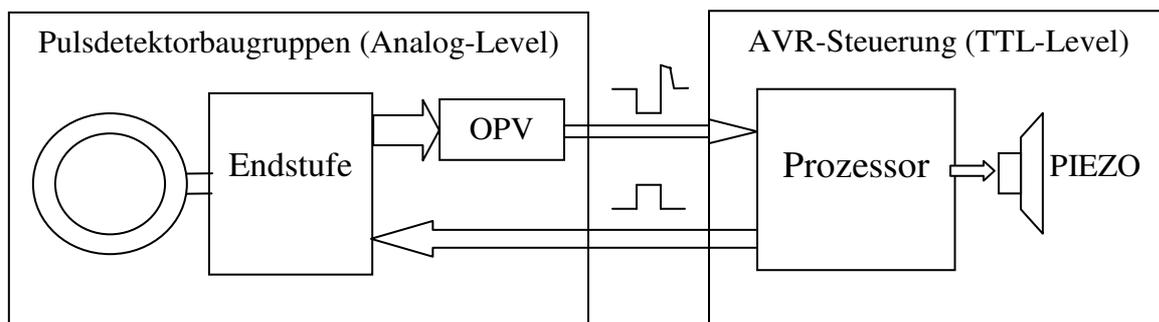


(Revision 1)

BT2

Angeregt durch die Entwicklung des BT1 ist die Idee entstanden einen Bausatz für Einsteiger zu schaffen der als interessanter Start in die Programmierung von AVR Mikroprozessoren dienen kann. Dafür wurde die Entwicklungsumgebung SiSy für die Programmierung von AVR Mikrocontrollern angepasst und eine spezielle Test- und Lernplattform entwickelt die eine einfaches ISP (In System Programming) von AVR Systemen ermöglicht. Die Detektoren der BT-Reihe sind bzw. werden mit der SiSy-ISP-Schnittstelle ausgerüstet. Damit ist die Entwicklung in Assembler, C und C++ möglich. Zusätzlich wird der Softwareentwurf zum Beispiel durch Programmablaufpläne, Struktogramme und UML-Klassendiagramme unterstützt. Der BT2 hat die Zielstellung einen Pulsdetektor mit Mikroprozessor bei geringst möglichem Bauteilaufwand zu realisieren. BT2 ist nicht für den realen Einsatz gedacht sondern als Experimental-, Lehr- bzw. Lernmuster für das Arbeitsprinzip und die Programmierung eines mikrocontrollergesteuerten Pulsdetektor. Die Zielgruppe sind vor allem Schüler, Studenten, Hobbyelektroniker und Einsteiger. Er kann auch als Einstiegsmodell für die eigene Eigenentwicklung von prozessorgesteuerten Metalldetektoren dienen. Der einfache Aufbau ermöglicht ein leichtes Verständnis der Baugruppen und einfaches nachvollziehen des Signalverlaufes. Für den BT2 werden Softwarevarianten entwickelt welche die Programmierung des Pulsdetektors in Assembler, C und C++ Schritt für Schritt transparent machen.

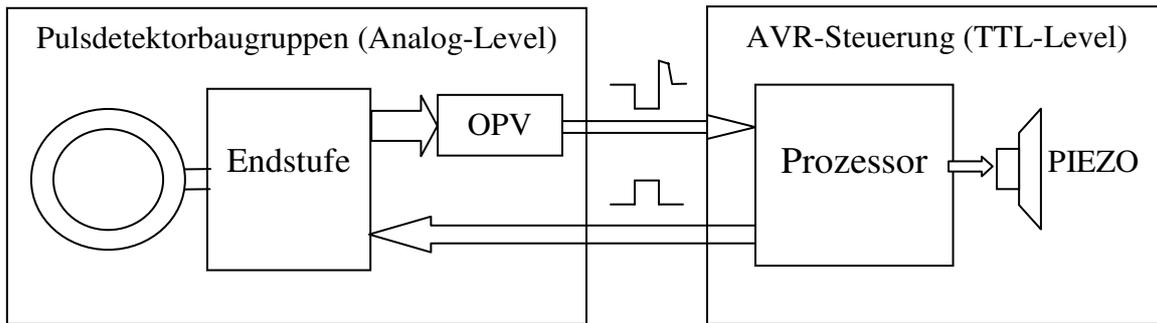
An dieser Stelle möchte ich ausdrücklich auf www.pulsdetektor.de verweisen, Die gesamte Entwicklungsgeschichte des BT2 ist dort im Forum nachzulesen. Des Weiteren möchte ich Chris und Norbert für ihre Unterstützung danken. Nur so konnte ich den BT1 optimieren und BT2 fertig stellen. Die in diesem Dokument bestimmt noch enthalten Fehler, bitte ich im Forum von pulsdetektor.de zu diskutieren.



Technische Daten

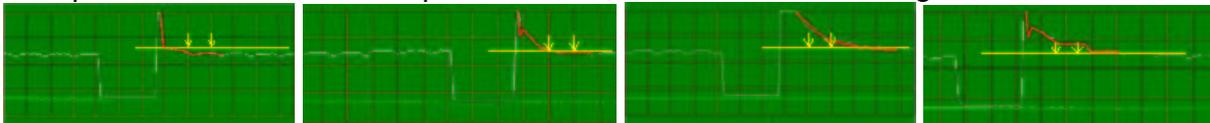
Betriebsspannung :	9 – 15V / DC
Betriebsstrom :	60 - 120 mA (Software/Spannungsabhängig)
Frequenz :	300-500 Hz (Softwareabhängig)
Pulsdauer :	10 – 200 yS (Softwareeinstellung)
Akku / Batterie :	6-12 Nc/Ni-MH, 9Volt Block, 5-10 Alcaline
Reichweite:	bis 30cm (Objekt- und Spulenabhängig)

Funktionsweise



Im Pulsdetektor BT2 wird die Steuerung von einem ATtiny12 Mikrocontroller der Firma Atmel übernommen. Das gesamte Timing kann im Programm eingestellt werden (PulsDauer, Zyklus, SampleDelay). Der Prozessor generiert in einem Abstand von ca. 1,7 – 3,5 mS (Zyklus+PulsDauer+SampleDelay) ein Pulssignal von 10 bis 200 μ S Breite (PulsDauer=1..20). Das Pulssignal (LO-Aktiv) liegt an PB3 (PIN2 ATtiny12) an und direkt an den Leistungstransistor IRF9640 gegeben. Eine Treiberstufe wie beim BT1 ist nicht notwendig da die der Leistungstransistor ebenfalls auf dem 5V Potential arbeitet. Der Leistungstransistor schaltet den Primärimpuls der aus einem 4700 μ F Elko gespeist wird. Die Spule erzeugt damit einen elektromagnetischen Impuls. Das nach dem Abschalten des Impulses auftretende Sekundärsignal (bei Metall in Spulennähe verändert sich der Verlauf der abfallenden Flanke) wird vom Operationsverstärker (LF357) verstärkt und an den Comperatoreingang PB1 (PIN6 ATtiny12) des Mikrocontroller gegeben. Der Comperator arbeitet kontinuierlich und vergleicht die Spannung zwischen PB0 und PB1 (PIN5/6 ATtiny12). Die Referenzspannung wird über den Poti mit Schalter (Spannungsteiler über Masse und +5V, Abgreifpunkt an PIN5) an PB0 geführt. Die softwareseitige Auswertung des Comperators wird nach einer Pause (SampleDelay) durchgeführt. Liegt zum Austastzeitpunkt das Signal an PB1 über der Referenzspannung an PB0, wird PortB 2 (PB2) auf LO gesetzt. Der PiezoSpeaker wird somit direkt vom Prozessor getrieben.

Signalverlauf ohne Metall im Suchbereich und bei einem Euro im Suchbereich. Gelb die Referenzspannung vom Poti und die beiden Austastzeitpunkte des Comperators. Rot das vom Operationsverstärker aufbereitete Signal.



Signalverlauf bei einem kleinen Schraubendreher (Eisen) und bei einem größeren Messinggegenstand.

Handhabung

Bei Inbetriebnahme darf kein Metall im Suchbereich der Spule sein. Am Poti mit Schalter einschalten und den Poti voll aufdrehen. Danach den Poti langsam zurückregeln bis der Speaker gerade aus ist (hier liegt die Kunst den empfindlichsten Bereich zu treffen). Betriebsbereit!

Kalibrierung

BT2 am Poti mit Schalter einschalten und ca. 30 Sekunden warm laufen lassen. Poti in Mittelstellung drehen. Am Trim-Poti des OPV dessen Arbeitspunkt einstellen. Trim-Poti nach links drehen bis der Speaker einen Dauerton abgibt. Langsam nach rechts regeln bis der Speaker gerade ausgeht. Kalibriert!

Stückliste

Widerstände:

1 1R0
1 100
3 470
1 560 2Watt
1 1M
1 100K Poti mit Schalter
1 220K Trimpoti stehend

Kondensatoren:

1 Elko 4700yF 16V
1 Elko 47yF 16V
1 KERKO 100nF

Dioden

1 1N5817
1 ZD 6,2

IC's

1 ATTiny12
1 IRF9640
1 LF357
1 7805

Sonstiges

Leiterplatte

PiezoSpeaker, selbst erregt, 3-12V

Spule

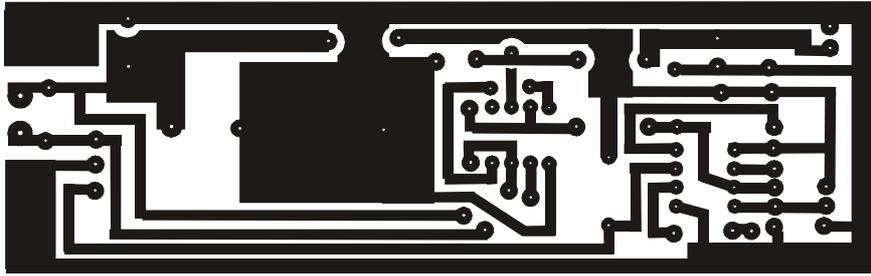
25m Schalllitze 0,2mm² PVC isoliert

Epoxydharz

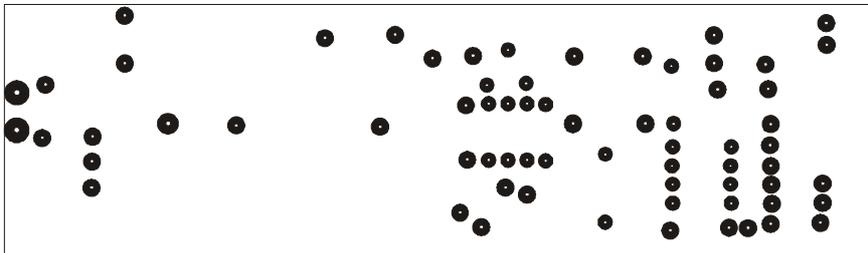
IC-Sockel (min. 1 für den ATtiny12)

Konnektoren

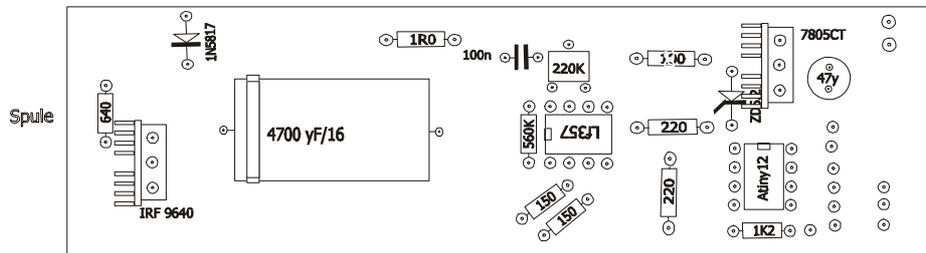
Layout



Bohrplan



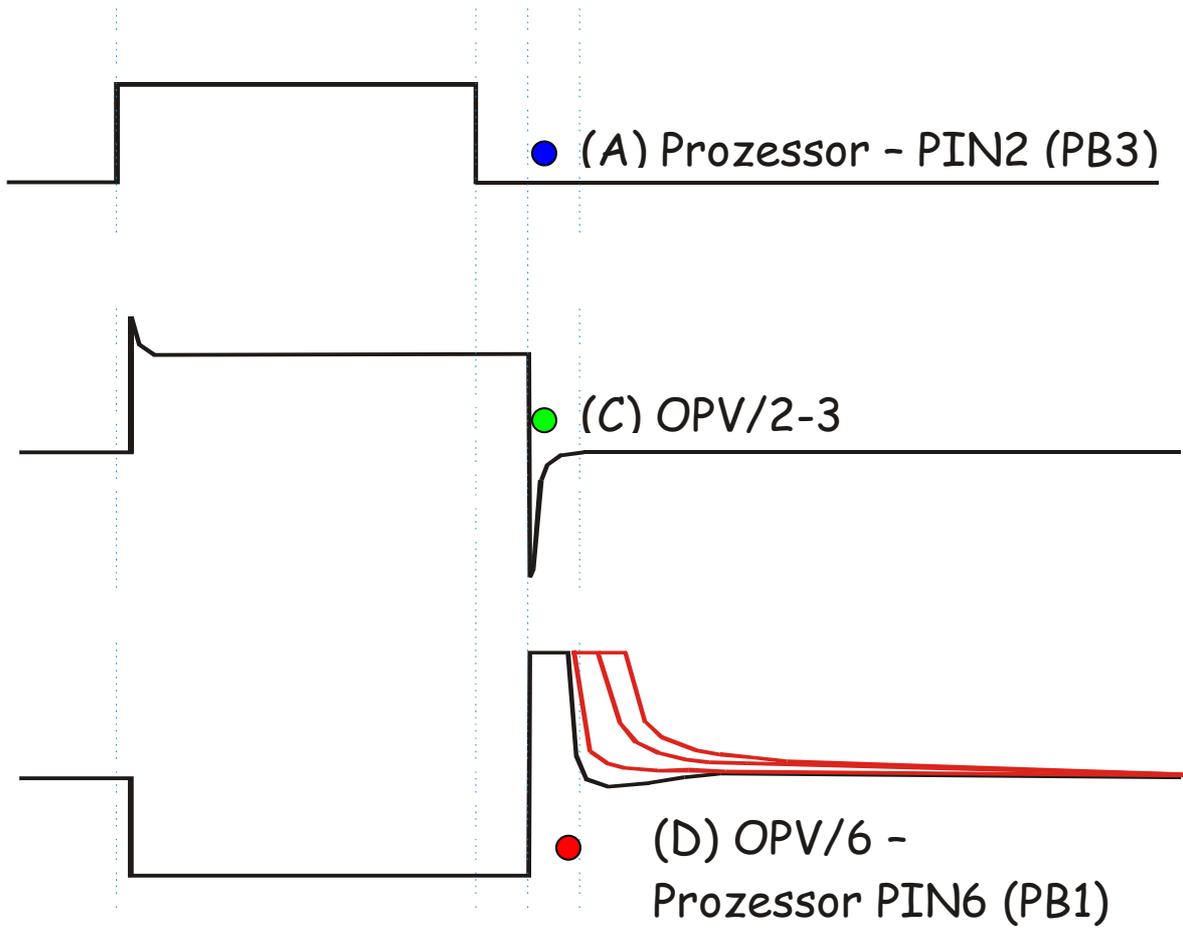
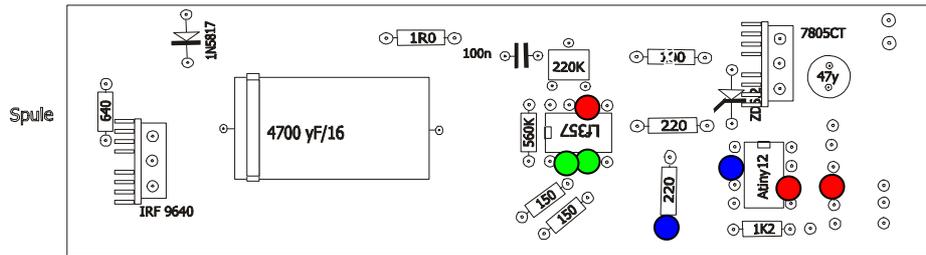
Bestückungsplan



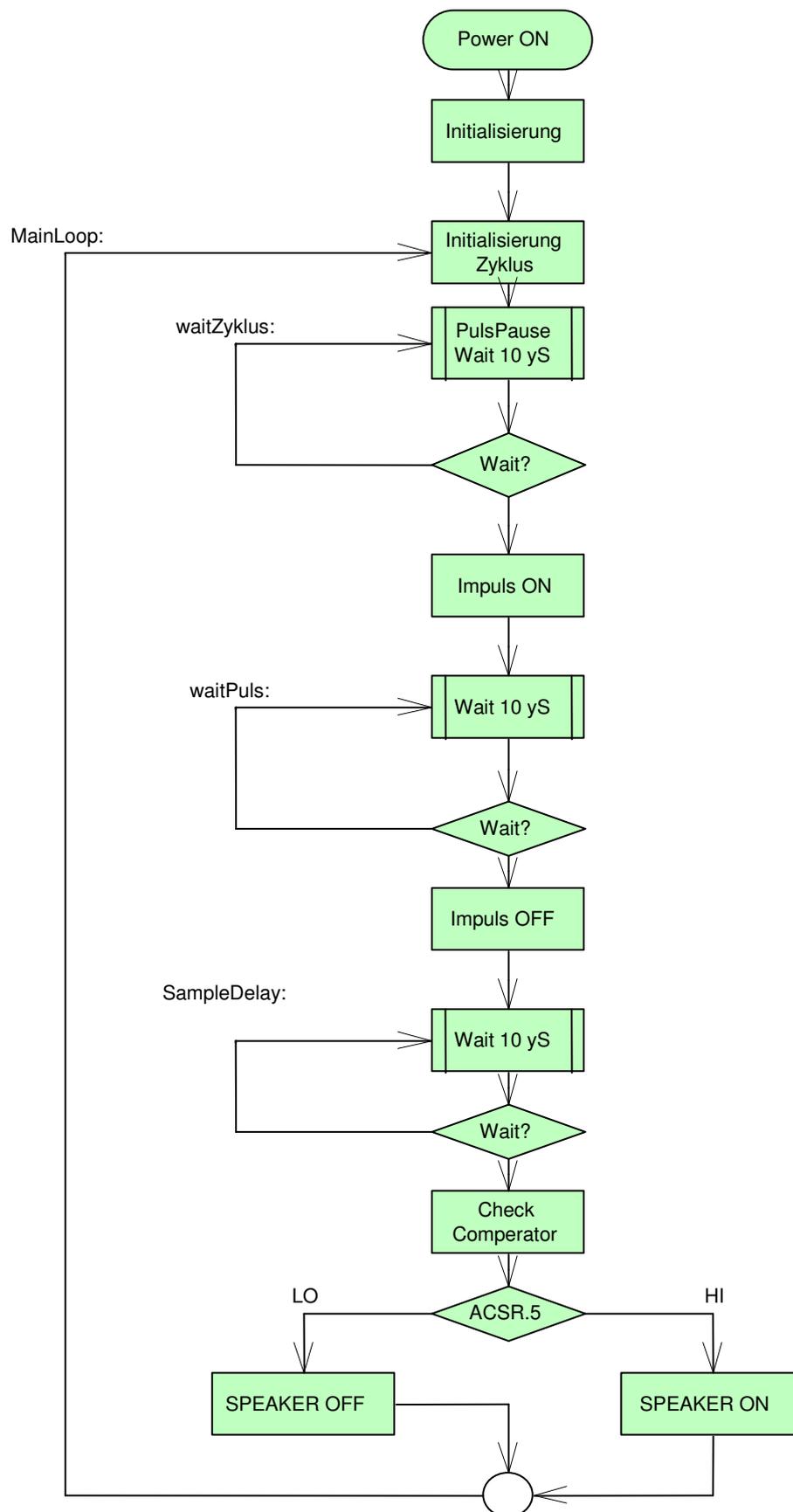
Leiterplatte

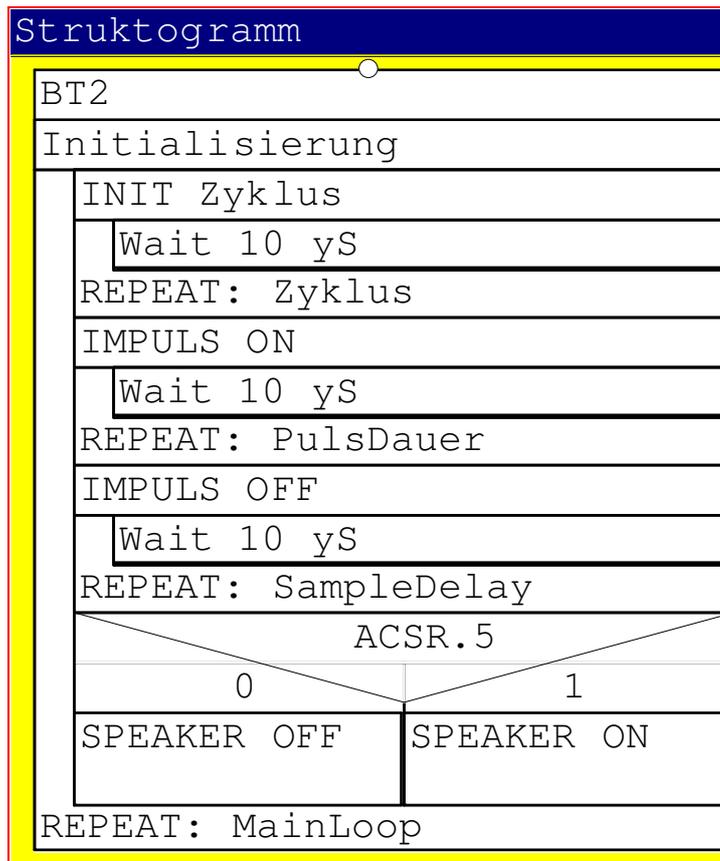


Messpunkte



SiSy-Programmablaufplan



SiSy-Struktogramm

Quellcode

```

;-----
;* Titel           :Steuerung für Pulsdetektor BT2
;* Prozessor      :AT-Tiny12 1,2 MHz intern
;* Sprache        :Assembler
;* Datum         :29.0.2004
;* Version       :1.3
;* Autor         :böses tier
;* Hardware      :für BT2 Layot-Version 1.9
;*              :PB0 an Poti,
;*              :PB1 an R220->LF357/6,
;*              :PB3 an IRF9640,
;*              :PB2 an SPK,
;*              :PB5 = RESET auf HI
;* Werkzeuge     :SiSy 2.16 mit GNU Assembler/Linker 3.x, AVRDUDE 3.x
;-----
.equ             ACSR,0x08
.equ             PINB,0x16
.equ             DDRB,0x17
.equ             PORTB,0x18
.equ             WDTCR,0x21
;-----
.equ             PulsDauer,10 ;1..20 ca 10..200 yS
.equ             Zyklus,200 ;Gesamtzyklus ca 1,7 mS 2 mal 80+ yS
.equ             Delay,3 ;SampleDelay
;-----
                rjmp    main ;$000 Power On Reset handler
                reti
                reti
                reti
                reti
                reti
;-----
; Start, Power ON, Reset, I/O init bei ATiny12 kein SRAM und STACK
main:
                nop
                ldi     R24,0x1F
                out     WDTCR,R24
                ldi     R24,0x17
                out     WDTCR,R24
                ; ----- watchdog is disable -----
                cbi     DDRB,0 ; PB0 IN OPV comperator
                cbi     DDRB,1 ; PB1 IN poti comperator
                sbi     DDRB,2 ; PB2 OUT SPEAKER
                sbi     DDRB,3 ; PB3 OUT Endstufe
;-----
                ldi     r17,0b001000
                ldi     r16,0b001100
mainloop:
                nop
                out     PORTB,r16 ;IMPULS OFF
                ;-----
                ldi     r18,Zyklus
waitZyklus:
                rcall   wait10
                subi    r18,1
                brcc   waitZyklus ;warteschleife
                ;-----
                eor     r16,r17
                out     PORTB,r16 ;IMPULS ON
                ldi     r18,PulsDauer ;wait 150 yS IMPULS
waitPuls:
                rcall   wait10
                subi    r18,1 ;warteschleife
                brcc   waitPuls
                eor     r16,r17
                out     PORTB,r16 ;IMPULS OFF
                ;-----
                ldi     r18,Delay ;wait ca 20-40 yS
sampleDelay:
                rcall   wait10
                subi    r18,1
                brcc   sampleDelay ;warteschleife
                in     r24,ACSR ;check COMPERATOR
                ldi     r16,0b0001000 ;alles aus oder ...
                sbrc   r24,5 ;wenn schwellwert skip next
                ldi     r16,0b0001100 ;PB2=1 Speakerimpulse Ton1
skip:
                rjmp    mainloop
;-----
wait10:
                ldi     r25,1 ;warte 10 yS
w10:
                subi    r25,1
                brcc   w10
                ret
;-----

```

Hexdump

```
:1000000005C01895189518951895189500008FE15A  
:1000100081BD87E181BDB898B998BA9ABB9ABC9A5C  
:1000200040E118E00CE0000008BB28EC16D021509D  
:10003000E8F7012708BB2AE010D02150E8F7012794  
:1000400008BB23E00AD02150E8F7042708BB88B199  
:10005000042708E085FD0CE0E6CF91E0915008957B  
:00000001FF
```