

Blinkande discogolv



Lunds Tekniska Högskola, LTH
Elektro- och informationsteknik

Grupp 8

Emma Trulsson, I11

Elin Varga, I11

Anna Bökberg, I11

EITF11 Digitala Projekt
VT 2014

Handledare: Bertil Lindvall

Inlämningsdag: 19 maj 2014

Abstract

The course Digitala Projekt , EITF11, focuses on the task to develop and construct a prototype of a chosen product. Hence the purpose of this paper is to describe the different stages of the project and its final result. The main function of the product is chosen to be the one of a disco floor. Firstly a specification of the requirements is specified. Thereafter the adequate hardware can be chosen and its components connected. Furthermore a source code is produced, which describes the products different functions, and creates the connection between the technical application and reality. The last stage consists of testing the prototype integrated with the source code.

Innehållsförteckning

Abstract	1
1 Inledning	4
1.1 Problemformulering.....	4
1.2 Syfte	4
1.3 Prototypbeskrivning.....	4
1.4 Avgränsningar	4
2 Teori	5
2.1 Musik.....	5
2.2 Komponenter	5
2.2.1 Filter	5
2.2.2 Operationsförstärkare	7
2.2.3 Resetknapp	8
2.2.4 Kristall	9
2.2.5 Analog-digital-omvandlare.....	9
3 Metod.....	10
3.1 Konstruktion.....	10
3.2 Programmering	11
4 Beräkningar	12
4.1 Lågpasfilter	12
4.2 Högpasfilter	12
4.3 Bandpassfilter	12
4.3.1 Bandpassfiltrets lågpasfilter	12
4.3.2 Bandpassfiltrets högpasfilter.....	12
4.4 Operationsförstärkaren	12
4.5 Programmering – timer.....	12
5 Resultat	14
6 Diskussion	15
6.1 Konstruktion.....	15
6.2 Musikalitet	15
6.3 Andra felkällor.....	16
6.4 Vidareutveckling	16

7 Slutsats	18
8 Referenslista.....	19
8.1 Elektroniska.....	19
8.2 Muntliga	19
9 Bilagor	20
9.1 Kravspecifikation.....	20
9.2 Kopplingsschema	21
9.3 Källkod.....	22

1 Inledning

1.1 Problemformulering

Det är vanligt att primärt fokus i projektarbeten hamnar på slutprodukten och att det inte reflekteras över vägen dit. För att komma till en slutprodukt som tar samtliga uppsatta krav i beaktande och som dessutom har väl genomtänkta lösningar behöver utvecklingsprocessen vara utformad på ett sätt som genererar detta, där inga komponenter glöms bort på vägen. Detta arbete ska därför skapa en förståelse för hur ett framgångsrikt utvecklingsarbete kan utformas.

1.2 Syfte

Syftet med detta arbete är att illustrera industriellt utvecklingsarbete. Detta görs genom att studenterna ska planera, konstruera och bygga en modell som efter noggranna justeringar ska generera i en prototyp för vidareutveckling med nödvändig dokumentation. Detta ska dessutom ge studenterna en god förståelse för hur utvecklingsarbete kan se ut.

1.3 Prototypbeskrivning

I detta projekt har en prototyp över ett blinkande discogolv konstruerats. I prototypen finns en mikrofon som tar upp musik utifrån. Sedan analyseras frekvenserna som finns representerade i musiken och delas upp i intervall som är kopplade till en lysdiod för respektive intervall. Dioden ska alltså blinka när frekvenser i tillhörande frekvensintervall finns representerade i det inkommande ljudet. Blinkandet kommer dessutom att ske i takt med musiken. Slutprodukten bör ha fler lysdioder än prototypen för att kunna täcka upp ett helt discogolv.

1.4 Avgränsningar

I detta arbete kommer färdiga komponenter såsom kondensatorer, resistorer och lysdioder användas. De kommer att sammanlänkas och monteras ihop av studenterna, som också själva ska konstruera kopplingsschema över modellen. Det kommer endast finnas tre lysdioder och musiken kommer att delas in i två intervall. Eftersom det finns begränsat utbud av komponenter kommer vissa uträknade värden för komponenter att i verkligheten uppskattas med närmsta tillgängliga värde.

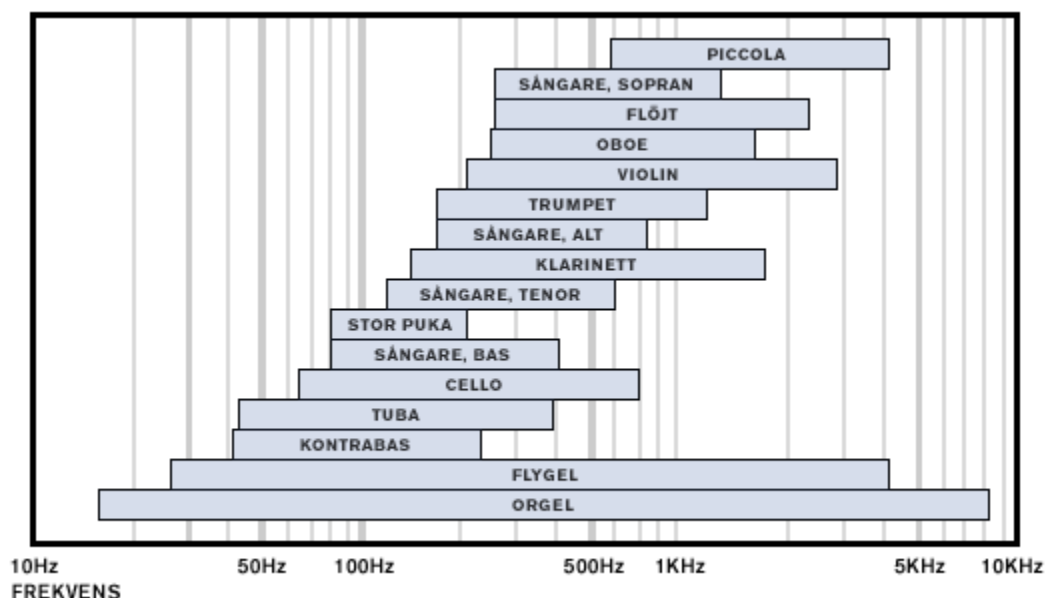
Dessutom finns en begränsning i tid då projektet ska utföras under en läsperiod. Detta gör att prototypen endast kommer att vara en småskalig variant av ett discogolv.

2 Teori

2.1 Musik

Ljud definieras vanligen som trycksvängningar som breder ut sig som vågor. Traditionellt menas svängningar i luften med en frekvens mellan 20 och 20 000 Hertz vilka kan uppfattas av det mänskliga örat¹.

Olika instrument kan generera toner som spänner över olika frekvensintervall. Nedan visas en bild där grundtonerna och dess frekvensområde för ett urval av instrument redovisas. Med grundtoner menas de toner som är möjliga att frambringa på respektive instrument².



Figur 1. Frekvensintervall som olika instrument kan generera toner i, <http://www.component.se/docs.php?id=3>.

2.2 Komponenter

2.2.1 Filter

Informationsbärande signaler behandlas av ett filter, vilket definieras som en elektrisk krets eller algoritm som överför insignalen till en utsignal på ett förutbestämt sätt. Ett filters funktion är således att släppa igenom signalen med valda frekvenser, samtidigt som oönskade frekvenser dämpas³.

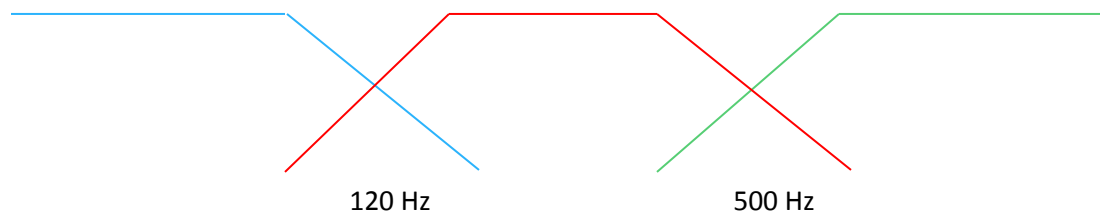
I detta projekt kommer passiva filter användas. Dessa konstrueras av resistorer samt kondensatorer. För att dela upp de inkommande signalerna i tre frekvensintervall, är det önskvärt att använda ett låg-, hög- samt bandpassfilter. Då filter kopplas med hjälp av en resistor och en kondensator blir tröskelfrekvens

¹ http://www.ne.se/lang/ljud/243395?i_h_word=frekvens (2014-04-10)

² <http://www.component.se/docs.php?id=3> (2014-04-10)

³ <http://www.ne.se/lang/filter/169537> (2014-04-10)

inte en skarp gräns utan en avtagande/stigande gräns. Detta innebär att filtret kommer att släppa igenom frekvenser som ligger utanför tröskelfrekvensen, se *Figur 2*⁴.



Figur 2. Illustration av filtrena. Blått - lågpasfilter, rött - bandpassfilter och grönt - högpasfilter.

För att räkna ut de olika komponenterna för filtrena i projektet har följande formel använts:

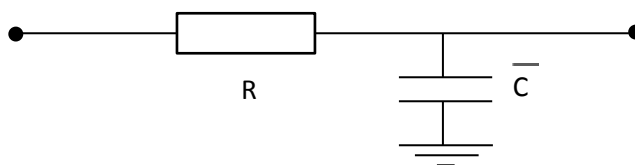
$$f = \frac{1}{2\pi RC} \quad \text{där } f \text{ är tröskelfrekvensen, } R \text{ beskriver resistorn samt } C \text{ kondensatorn.}$$

Tröskelfrekvensen varierar beroende på vilket filter som ska användas⁵.

2.2.1.1 Lågpasfilter

Ett lågpasfilter släpper igenom frekvenser upp till en viss nivå och dämpar frekvenser över denna. I praktiken är det svårt att uppnå ett idealt lågpasfilter, varför en approximation ofta används⁶.

Lågpasfiltret kopplas enligt bild nedan⁷.



Figur 3. Illustration av lågpasfiltret.

2.2.1.2 Högpasfilter

Ett högpasfilter släpper igenom frekvenser över en viss nivå och dämpar frekvenser som ligger under denna⁸.

Högpasfiltret kopplas enligt *Figur 4*⁹.

⁴ Bertil Larsson Universitetsadjunkt CI Lunds Tekniska Högskola, samtal den 7 april 2014

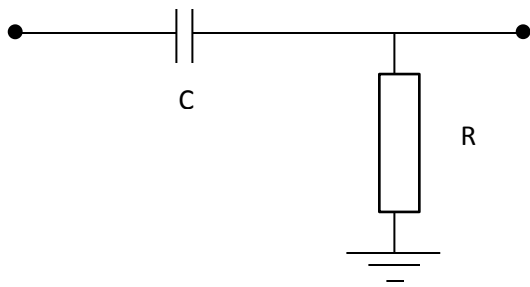
⁵ <http://www.eit.lth.se/fileadmin/eit/courses/ess010/Laborationer/handledning2013.pdf> (2014-05-07)

⁶ <http://www.ne.se/lang/filter/169537> (2014-04-10)

⁷ Larsson

⁸ <http://www.ne.se/lang/h%C3%B6gpasfilter> (2014-04-10)

⁹ Larsson

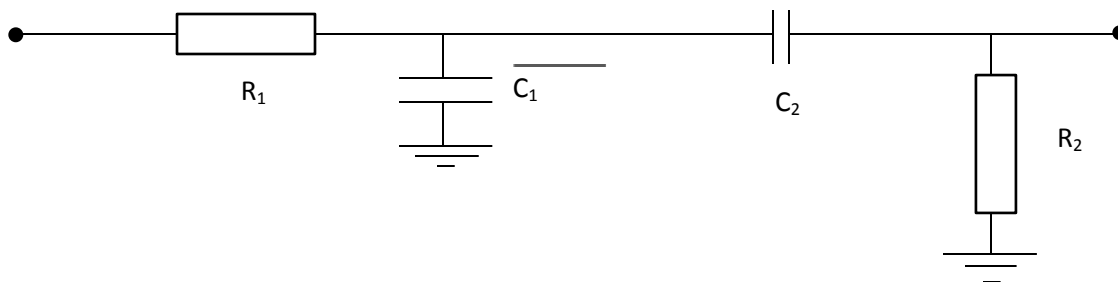


Figur 4. Illustration av högpasfilteret.

2.2.1.3 Bandpassfilter

Ett bandpassfilter låter signaler mellan två bestämda frekvenser passera. Signaler som antingen är över eller under detta intervall dämpas. För att uppnå denna effekt konstrueras bandpassfiltret med hjälp av först ett lågpasfilter och sedan ett högpasfilter. Om tröskelfrekvensen för lågpasfiltret sätts till bandpassfiltrets takfrekvens kommer alla signaler under denna passera medan signaler med frekvenser högre än detta kommer att dämpas. Högpasfiltrets tröskelfrekvens sätts således till bandpassfiltrets lägsta tillåtna frekvens. Alla signaler som har en frekvens högre än denna kommer då passera. Resultatet blir att signaler mellan bandpassfiltrets lägsta och högsta tillåtna frekvens kommer att passera medan alla andra dämpas, ett bandpassfilter har skapats¹⁰.

Bandpassfiltret kopplas enligt *Figur 5*¹¹.



Figur 5. Illustration av bandpassfiltret.

2.2.2 Operationsförstärkare

En operationsförstärkare, OP-förstärkare, är en elektronisk förstärkare vars huvuduppgift är att höja eller sänka amplituden. OP-förstärkaren har två ingångar, vilket illustreras i *Figur 6*. Insignalen från kretsen kopplas med OP-förstärkarens pluspol medan minuspolen kopplas med insignalen från återkoppling¹².

OP-förstärkaren kännetecknas av en hög förstärkning av spänningsskillnaden mellan förstärkarens ingångar samtidigt som en låg förstärkning av den gemensamma spänningen för de båda ingångarna. En

¹⁰ <http://www.ne.se/bandpassfilter> (2014-04-10)

¹¹ Larsson

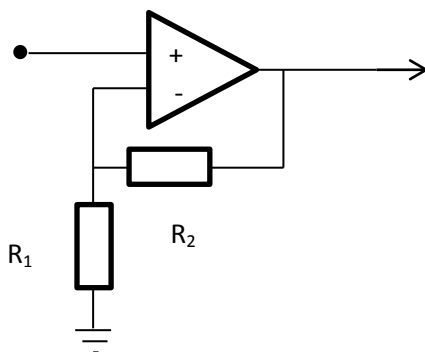
¹² Ibid

OP-förstärkare fungerar som en komparator, det vill säga kan endast jämföra två spänningar med varandra¹³.

Återkoppling krävs för att försäkra att OP-förstärkaren får funktion som exempelvis förstärkare. Detta löses genom att en liten andel av spänningen på utgången kopplas tillbaka till ingången, på detta vis utgörs insignalen till OP-förstärkaren av skillnaden mellan denna lilla andel och den önskade signalen. Återkopplingen används också för att minimera självsvängningar från insignalen¹⁴.

$A = 1 + \frac{R_2}{R_1}$ där A är förstärkning och R_1 och R_2 är resistorerna¹⁵.

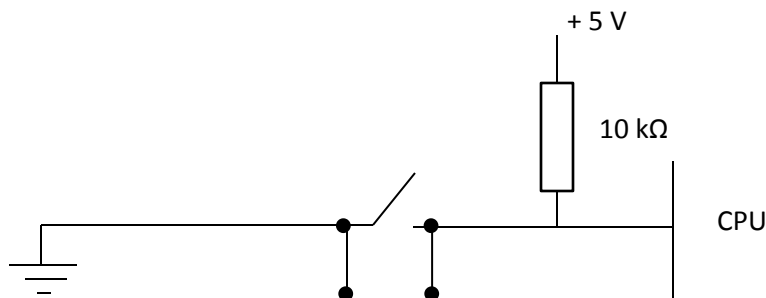
OP-förstärkaren kopplas enligt *Figur 6*.



Figur 6. Illustration av OP-förstärkaren.

2.2.3 Resetknapp

En resetknapp, återställningsknapp, återställer ett system till sitt ursprungsläge. Då knappen trycks ned forcerar den en systemåterställning. Ett av benen kopplas till processorn. Det andra benet kopplas till GND. De resterande två kopplas parallellt ihop med det ben som kopplats till processorn respektive det som är kopplats till GND, se *Figur 7* för illustration¹⁶.



Figur 7. Illustration över resetknappen.

¹³ <http://www.ne.se/lang/operationsf%C3%B6rst%C3%A4rkare> (2014-04-10)

¹⁴ <http://www.ne.se/lang/operationsf%C3%B6rst%C3%A4rkare> (2014-04-10)

¹⁵ Larsson

¹⁶ <http://ww1.microchip.com/downloads/en/DeviceDoc/21615c.pdf> (2014-04-10)

2.2.4 Kristall

Inom mätteknik får inte frekvensvariationer större än ett fåtal miljondelar. För att uppnå detta styrs oscilloskopet av ett tillståndselement vilken binder frekvensen till ett förutbestämt värde.

Kvartskristallen är ett sådant element¹⁷.

2.2.5 Analog-digital-omvandlare

Analog-digital-omvandlare, A/D-omvandlare, är en komponent som omvandlar analoga signaler till digitala. Det minsta värdet som A/D-omvandlaren kan konvertera är spänningen från GND, det vill säga noll volt. Det största värdet representeras av spänningen från AREF och väljs av operatören¹⁸.

¹⁷ <http://www.aef.se/Elektronikutveckling/Artiklar/Kvartsosc/Kvartsoscillatorer.htm> (2014-04-10)

¹⁸ <http://www.atmel.com/Images/doc2466.pdf> (2014-04-10)

3 Metod

3.1 Konstruktion

För att få discogolvet att blinka i tre olika färger måste de inkommande signalerna delas upp i tre olika intervall. Utifrån *Figur 1* har de tre intervallen valts till 0 – 120 Hertz för lågpasfiltret, 120 – 500 Hertz för bandpassfiltret samt 500 – ∞ Hertz för högpasfiltret. Detta eftersom det ur *Figur 1* går att avläsa att det finns grundtoner för musikinstrument representerade i samtliga intervall.

Arbetet inleds med att göra en kravspecifikation för den tänkta produkten. Detta ska göras för att klargöra vilka funktioner discogolvet ska ha. Den färdiga kravspecifikationen fungerar sedan som en mall för det vidare arbetet, se *9.1 Kravspecifikation*. Därefter ska ett dokument över alla nödvändiga komponenter göras vilka sammankopplas i ett kopplingsschema. I detta visualiseras hur de olika komponenterna förhåller sig till varandra, se *9.2 Kopplingsschema*. Ett delkopplingsschema från mikrofonen till processorn, CPU:n, även innehållande filtrena, erhålles från extern handledare då detta inte faller inom ramarna för kursen. Denna del är analog.

När dessa två delar avklarats är grunden för projektet lagd och själva konstruktionsarbetet börjar. De komponenter som ska kopplas ihop illustreras i *9.2 Kopplingsschema*. Låg-, band- och högpasfiltrena används för att få rätt frekvensintervall in till A/D-omvandlaren. Resetknappen används för att kunna nollställa kretsen i de fall då det kan behövas. Kristallen förstärker samt stabiliserar de inkommande frekvenserna och kopplas in för att kunna generera tidsavbrott. Vidare utnyttjas OP-förstärkaren för att förstärka amplituden på de analoga signalerna samt minska brus och självsvängningar. Alla kontakter på VCC- och GND-ledningen kommer lödas fast medan övriga kommer viras. Ett undantag från detta är resetknappen som kommer lödas fast eftersom denna har för korta ben för virning.

Eftersom signalerna från mikrofonen in till CPU:n är analoga kommer dessa delar först testas på en sockel innan komponenterna monteras på kretskortet. Under testningen kan storleken på olika komponenter fastställas utifrån givna värden, uträkningar och kopplingsschemat. Detta kommer göras för både kopplingen från mikrofonen in till filtrena, filtrenas komponenter samt OP-förstärkaren. Följden kommer succesivt monteras på sockeln och kommer att testas med hjälp av ett oscilloskop, där en signal från en tongenerator skickas in med en frekvens inom respektive filters område. Önskat resultat är en signal utan självsvängningar och med önskad spänning. När detta resultat erhålles monteras följden på kretskortet.

För att bestämma hur resetknappen ska kopplas används en multimeter i syfte att avgöra mellan vilka ben koppling ska ske. De två ben som resulterar i noll Ohm på multimetern då resetknappen är nertryckt, är knappens switch. Switchen illustreras i *Figur 7* som den diagonala linjen. När detta är bestämt, kan resetknappen kopplas enligt *Figur 7*.

Då konstruktionen är klar kommer prototypen att testas i programmet Atmel Studio 6.0. Prototypen kopplas ihop med programmet med hjälp av komponenten JTAG och USB-sladd. I Atmel testas först de olika ingångarna och utgångarna. I detta fall kommer tre olika lysdioder samt låg-, band- och högpasfiltrena som konverteras till digitala signaler i A/D-omvandlaren att testas. Lysdioderna kommer

att testas genom att se om de lyser då rätt port är markerad samt A/D-omvandlaren kommer att testas genom att se om omvandlaren genererar rätt värde av den analoga insignalen. Det sistnämnda görs genom en kortare källkod samt att två av portarna kopplas till GND (noll Volt) respektive VCC (fem Volt).

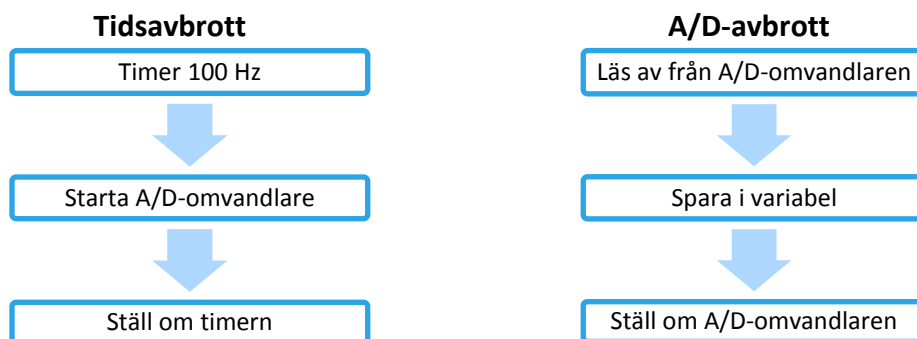
3.2 Programmering

Sista steget består av att programmera, se 9.3 Källkod, för att säkerställa att lysdioderna blinkar vid rätt frekvens och i lämplig takt. Källkoden kommer även att testas med hjälp av JTAG och Atmel Studio 6.0 för att uppnå önskat resultat. Vid testningen kommer olika toner att spelas upp i olika intervall med en tongenerator för att testa om rätt lysdiod blinkar vid rätt frekvensintervallintervall.

Källkoden till prototypen består av två avbrott (interrupt) och olika algoritmer för att beräkna hur lysdioderna ska blinka i takt till musiken. Först kommer ett tidsavbrott att göras var tionde millisekund. I detta tidsavbrott startas A/D-omvandlaren, antingen via ingången för låg-, band eller för högpassfiltret. Därefter kommer ett A/D-avbrott att ske där värdet från A/D-omvandlaren läses in och sparas som en variabel. Dessa två avbrott sker kontinuerlig då prototypen är kopplad till en strömkälla. För att A/D-omvandlaren inte ska kunna läsa in signaler oftare än tidsavbrottet sker bör prescalern hos A/D-omvandlaren väljas till 16.

Samtidigt som de två avbrotten sker, ska de olika algoritmerna köras. När insignalen från respektive filter överstiger en viss bestämd gräns, "limit" (olika för respektive filter), ökar en räknare med ett. Samtidigt finns en räknare som ökar med ett för varje tidsavbrott. För att sedan beräkna antal gånger som varje lysdiod ska lysa per sekund beräknas den procentuella tid som insignalen översteg limit. Detta görs genom att registrera antal frekvenstoppar över limit under två sekunder. Därefter divideras antal frekvenstoppar över limit med en faktor som är beräknat i förhållande till två sekunder. Denna faktor väljs till 200 eftersom antal frekvenstoppar över limit ska divideras med den totala tiden som denna beräkning gjorts (två sekunder). Därefter multipliceras denna procentsats med tio och divideras sedan med tre för att lysdioderna inte ska blinka för snabbt. Denna uppskattning är gjord för att försöka replikera takten i musiken.

För att sedan få lysdioderna att blinka delas tiden upp i olika intervall. Under ett visst intervall ska lysdioden vara tänd, medan den nästa intervall ska vara släckt. En gul lysdiod kopplas till lågpassfiltret, en röd till högpassfiltret samt en grön lysdiod till bandpassfiltret.



Figur 8. Illustration över avbrotten.

4 Beräkningar

4.1 Lågpasfilter

Tröskelfrekvensen $f = 120$ Hz

Resistorn sätts till $R = 10$ k Ω

Ger kondensatorn: $C = \frac{1}{2\pi Rf} = \frac{1}{2\pi \cdot 10 \cdot 10^3 \cdot 120} \approx 0,13 \mu F$

4.2 Högpasfilter

Tröskelfrekvensen $f = 500$ Hz

Resistorn sätts till $R = 10$ k Ω

Ger kondensatorn: $C = \frac{1}{2\pi Rf} = \frac{1}{2\pi \cdot 10 \cdot 10^3 \cdot 500} \approx 0,03 \mu F$

4.3 Bandpassfilter

Bandpassfiltret består av ett låg- och ett högpasfilter. Detta betyder att två kondensatorer måste bestämmas för detta filter.

4.3.1 Bandpassfiltrets lågpasfilter

Tröskelfrekvensen $f = 500$ Hz

Resistorn sätts till $R = 10$ k Ω

Ger kondensatorn: $C = \frac{1}{2\pi Rf} = \frac{1}{2\pi \cdot 10 \cdot 10^3 \cdot 500} \approx 0,03 \mu F$

4.3.2 Bandpassfiltrets högpasfilter

Tröskelfrekvensen $f = 120$ Hz

Resistorn sätts till $R = 100$ k Ω

Ger kondensatorn: $C = \frac{1}{2\pi Rf} = \frac{1}{2\pi \cdot 100 \cdot 10^3 \cdot 120} \approx 0,013 \mu F$

4.4 Operationsförstärkaren

$$A = 1 + \frac{R_2}{R_1} \quad \rightarrow \quad 11 = 1 + \frac{100\,000}{10\,000}$$

Önskad förstärkning, A, är 11. R_1 sätts därför till 10 k Ω och R_2 till 100 k Ω .

4.5 Programmering – timer

Vilken timer som ska användas till tidsavbrottet bestäms utifrån beräkningar. Tidsavbrottet är bestämt till tio millisekunder vilket är detsamma som 100 Hertz. Kristallen, vilken agerar som en extern klocka, är på $16 \cdot 10^6$ Hertz och timern måste kunna skala ner denna frekvens till 100 Hertz. Den högsta prescaler som timern kan ha är 1024.

$$100 = \frac{16 \cdot 10^6}{1024 \cdot x} \quad \rightarrow \quad x = 156,25$$

För att kunna avgöra vilken timer som behövs, jämförs en åtta-bit timer med en 16-bit timer. En åtta-bit timer ger det decimala talet 255, medan en 16-bit timer ger talet 65535. Utifrån detta ses att en åtta-bit timer är fullt tillräcklig eftersom $156,25 < 255$. Detta resulterar i att TIMERO kommer att användas vid programmeringen.

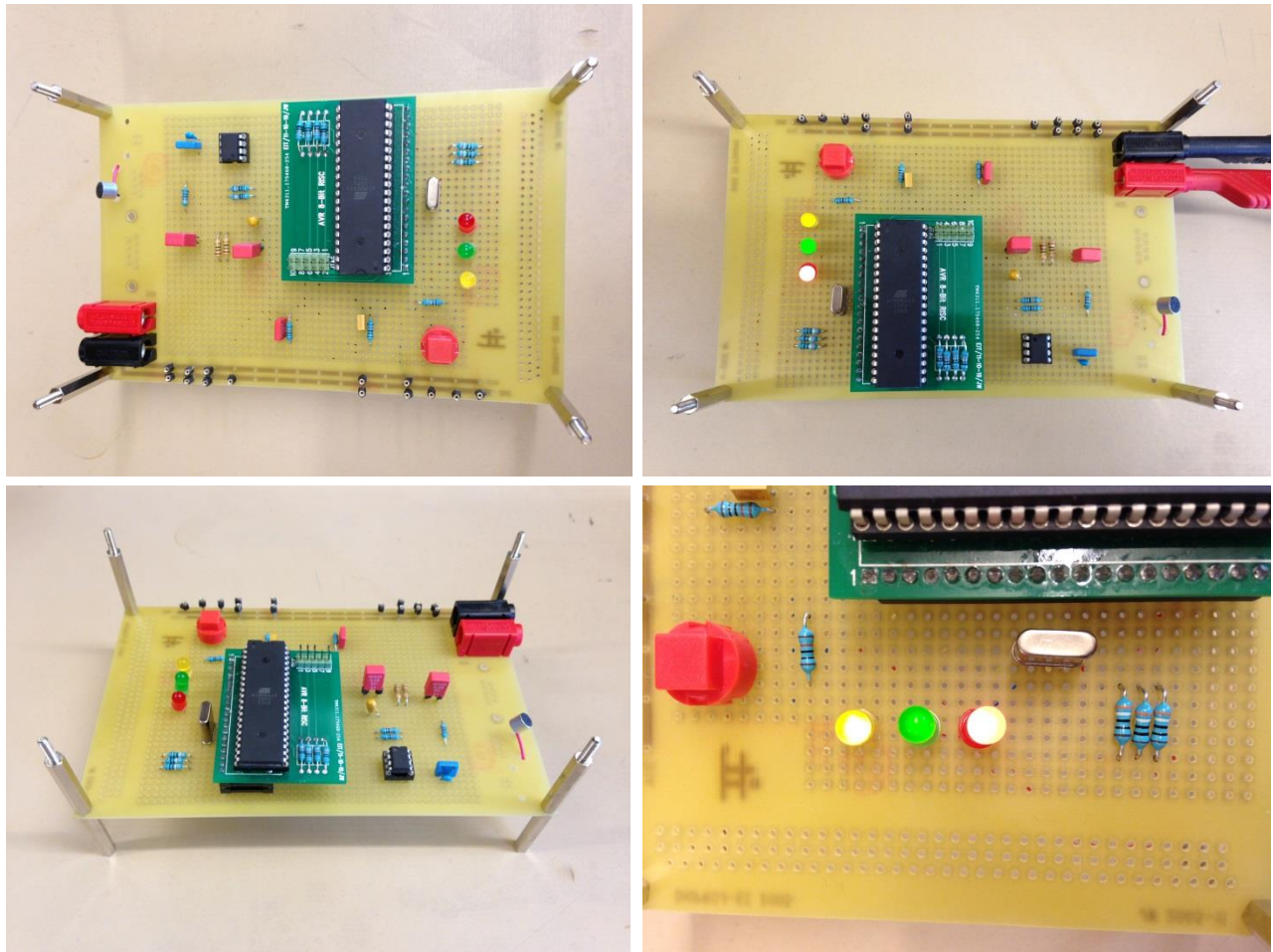
För att tidsavbrott ska ske med lämplig periodicitet, det vill säga var tionde millisekund, sätts ett initieringsvärde vilket gör att timern istället för att börja räkna från noll startar på initieringsvärdet. När timern når upp till det decimala talet 255 sker tidsavbrottet. Därefter nollställs timern samt påbörjar ny uppräknings från initieringsvärdet. Initieringsvärdet beräknas enligt (värdet 156,25 används för att uppnå periodiciteten 10 millisekunder, se beräkning ovan):

$$255 - 156,25 = 98,75$$

5 Resultat

Under projektets gång gjordes ett antal revideringar. Den största innefattade bandpassfiltret. Under de analoga testen med oscilloskopet upptäcktes att intervallgränserna var flackare än väntat. Detta resulterade i att de olika intervallen överlappade till för stor del. Det blev alltså ingen tydlig skillnad mellan de olika intervallen, då samtliga toner togs upp av bandpassfiltret men också av ett av de andra intervallen, beroende på ton. Detta löstes genom att bandpassfiltret togs bort. Därmed uppkom två mer distinkta intervall vilka, trots förändringen, överlappade varandra till viss del. För att fortfarande ha tre olikfärgade lysdioder i drift konstruerades en algoritm i källkoden som lät den tredje dioden vara en funktion av de två andra.

Det slutgiltiga resultatet är en prototyp av ett blinkande discogolv, enligt planeringen, där ljus och blinkfrekvens beror av inkommande ljud. Se prototyp i bild nedan.



6 Diskussion

6.1 Konstruktion

OP-förstärkaren kopplades in mellan mikrofon och filtren för att minimera brus och minska självsvängningen. Den förstärkning som OP-förstärkaren genererar sattes till elva genom att först testa och kontrollera in- och utsignal med oscilloskopet. Initialt sattes samma storlek på resistorerna R_1 och R_2 , vilket resulterade i en för låg förstärkning och oönskade självsvängningar hos utsignalen. För att minimera självsvängningarna krävdes en högre förstärkning. Detta löstes genom att tiodubbla R_2 , vilket resulterade i en stabilare utsignal och en fullt tillräcklig förstärkning.

Fördelen med att använda passiva filter är att de är enkla att dimensionera. Dock är det svårt att uppnå skarpa tröskelvärden. För att framställa ett discogolv som blinkar i fler färger och med fler frekvensintervall kan det därför vara önskvärt att använda sig av en annan filtermodell som genererar skarpare tröskelvärden för att förhindra överlappning. Trots denna svårighet lyckades ett tillfredställande resultat uppbringas.

Lösningen med ett programmerat bandpassfilter valdes på grund av att dioderna skulle blinka på ett visuellt tilltalande sätt vilket inte skett om den ena lysdioden hade lyst konstant. Dessutom gav detta möjligheten att låta lysdioden som representerar bandpassfiltret lysa då de andra var släckta, vilket skapar en bra effekt, speciellt i lugnare låtar där de resterande dioderna hade varit släckta den största delen av tiden.

Nackdelen med ett programmerat filter är att det blir mer tidskrävande att programmera lysdiodens blinkningar, då den inte är kopplad till en separat insignal utan är helt beroende av de två andra lysdiodernas insignaler. Detta gör att de simulerade blinkningarna måste ta hänsyn till hur de andra två lysdioderna kommer blinka och hitta ett gemensamt mönster hos dessa. Dock är detta bättre än att ha ett bandpassfilter inkopplat som hade genererat överlappande blinkningar.

6.2 Musikalitet

Ett krav var att lysdioderna skulle blinka i takt till musiken. Detta är valt att göras med analys av hur mycket ljud som kommer in i de olika intervallen och studera när ljudet överstiger en viss gräns. Det som är en avgörande faktor i denna analys är den minsta gräns ljudet måste överstiga för att det ska tas i beaktande som en frekvenstopp och således tas med i beräkningen av takten i musiken. Om gränsen är för låg kommer för mycket ljud att tas upp och vice versa om gränsen är för hög. Denna gräns är därför optimal för blinkfrekvensen på olika nivåer beroende på vilket typ av ljud som kommer in. Detta resulterar i att denna gräns är anpassad för att fungera bra för så många ljud som möjligt, men är således inte optimal för alla.

Dessutom är det i programmeringen bestämt att toppar endast ska läsas av vid avbrott var tionde millisekund. Hade avbrott gjorts oftare hade en större del av det inkommande ljudet uppfattats och det hade utgjort ett mer heltäckande underlag till frekvensanalysen. Därför utgör den valda avbrottsfrekvensen en begränsning för prototypen. Dock anses 100 avbrott per sekund generera ett tillräckligt stort underlag för att uppnå produktens syfte.

En annan sak som påverkar huruvida lysdioderna blinkar i takt med musiken eller inte är algoritmen som beräknar hur många gånger per två sekunder som lysdioderna ska blinka. Eftersom tiden då insignalen överstiger den valda gränsen, alltså tiden med toppar, delas med ett förvalt värde, kommer blinkfrekvensen påverkas av detta värde. Därmed är det inte givet att den kalkylerade siffran kommer reflektera den verkliga takten i musiken (med till exempel 120 slag/minut), eftersom sång och melodi ofta inte enbart ligger på taktslaget och är tyst däremellan. Vanligare är att melodin är kontinuerlig och flyter över taktslagen.

Dock är det inte säkert att lamporna kommer blinka i takt även om till exempel en metronom används, som endast genererar pulsslag i ett förutbestämt tempo. Detta eftersom den förutbestämde faktorn i så fall måste spegla hur lång tid det kommer ut ljud ifrån metronomen i förhållande till hur länge det är tyst emellan ljudintervallen. Detta eftersom blinkningarna kommer bero på den procentuella tiden som det kommer in ljud i form av en insignal och inte på hur signalerna sammanfaller. Om det kommer in tio pulsslag på två sekunder och det motsvarar 50 procent av tiden, kommer inte prototypen veta om det är två eller 20 slag som kommit in, utan endast att de kommit in under 50 procent av tiden och beräknar således en takt utifrån detta.

6.3 Andra felkällor

En möjlig felkälla är att de olika komponenterna inte nödvändigtvis har önskad precision. Eftersom projektet i första hand går ut på att få djupare förståelse för processen och inte för resultatet i den slutgiltiga prototypen kan det antas att de ingående komponenterna inte är branschledande. Dock anses de ha tillräckligt bra precision för att skapa en fungerande prototyp och för att uppfylla projektets syfte.

Under projektets gång har det till exempel upptäckts att ljudkällan måste hållas tätt intill mikrofonen för att denna ska uppfatta ljudet. Därför finns det möjlighet att lysdioderna inte blinkar korrekt om ljudkällan hålls för långt ifrån mikrofonen alternativt att den skrapar emot mikrofonen och i så fall ger upphov till brus.

Slutligen är det trots noggrannhet, möjligt att lödningar och virningar gjorda under projektets gång, inte blivit korrekt utförda eller att de släppt marginellt under testning och hantering av prototypen. Allvarliga fel borde dock ha upptäckts under testnings- och programmeringsfasen och då detta inte gjorts anses eventuella brister vara försumbara.

6.4 Vidareutveckling

Eftersom mikrofonen tar in allt ljud utifrån, även till exempel prat och buller, hade ytterligare ett filter med uppgift att filtrera bort och dämpa detta kunnat tilläggas. På så sätt hade lysdioderna endast blinkat i relation till musiken och inte påverkats av diverse störningar. Att även dämpa sången och endast ta hänsyn till melodin i musiken ses som realistiskt då sångens frekvenser sammanfaller med flera instruments, se *Figur 1*, vars frekvensintervall önskas tas in i processen.

Om projektet hade löpt över en längre tid hade prototypen kunnat utvecklas till att också innefatta en transparent box runt dioderna för att på så sätt representera en miniatyr av en golvplatta. Detta hade sedan även kunnat utvecklas till en platta i verklig storlek med fler lysdioder. Dessa hade då kunnat

omfatta fler frekvensintervall och fler färger hade kunnat representeras i plattan. Dessutom hade fler färger på lysdioder kunnat användas för att skapa häftiga blinkmönster som beror av hur de andra lamporna blinkar som i sin tur beror på vilka frekvenser som tas in av mikrofonen.

Ytterligare ett utvecklingsområde är att skapa en databas som kan spara både infångat ljud och även dess motsvarande blinkfrekvens. Integreras detta med en högtalare hade det varit möjligt att återspela tidigare spelat ljud och sedan låta lamporna blinka efter detta.

7 Slutsats

Under projektets gång har djupare insikt kring utvecklingsarbetets komplexitet uppnåtts. Det som har varit den tydligaste lärdomen är att mycket kan gå fel och att effektiv felsökning därför är viktig. I detta projekt har felsökningen både involverat studier av hela kretsen med ett oscilloskop för att kontrollera att rätt signaler färdades på rätt ställen men också med JTAG i programkoden.

En annan viktig insikt som gjorts är vikten av dokumentation. Under processens gång har en loggbok förts där det dagliga arbetet har dokumenterats. Denna information har varit användbar då något behövde kollas upp från tidigare arbetstillfällen men fungerade också bra för att veta vad som behövdes göras under nästa arbetstillfälle.

Slutligen har en insikt kring att det alltid är bäst att börja småskaligt och se att det grundläggande fungerat för att sedan vidareutveckla och arbeta fram en mer avancerad produkt. Studenterna kommer att ta med sig insikter från detta projekt, rörande tidskomplexitet och det mångfacetterade arbetet som ligger bakom en färdig produkt, till framtida projekt och arbetssituationer.

8 Referenslista

8.1 Elektroniska

<http://www.eit.lth.se/fileadmin/eit/courses/ess010/Laborationer/handledning2013.pdf> (2014-04-07)

http://www.ne.se/lang/ljud/243395?ih_word=frekvens (2014-04-10)

<http://www.component.se/docs.php?id=3> (2014-04-10)

<http://www.ne.se/lang/filter/169537> (2014-04-10)

<http://www.ne.se/lang/filter/169537> (2014-04-10)

<http://www.ne.se/lang/h%C3%B6gpassfilter> (2014-04-10)

<http://www.ne.se/bandpassfilter> (2014-04-10)

<http://www.ne.se/lang/operationsf%C3%B6rst%C4%A4rkare> (2014-04-10)

<http://ww1.microchip.com/downloads/en/DeviceDoc/21615c.pdf> (2014-04-10)

<http://www.aef.se/Elektronikutveckling/Artiklar/Kvartsosc/Kvartsoscillatorer.htm> (2014-04-10)

<http://www.atmel.com/Images/doc2466.pdf> (2014-04-10)

8.2 Muntliga

Larsson, Bertil, Universitetsadjunkt CI Lunds Tekniska Högskola, samtal den 7 april 2014

9 Bilagor

9.1 Kravspecifikation

Kravspecifikation - Blinkande discogolv

Produktbeskrivning

Slutprodukten i projektet kommer vara en prototyp av ett blinkande discogolv. Detta discogolv blinkar i takt med musiken och ändrar färg beroende på frekvensintervallet i musiken.

Krav och funktion

Krav på discogolvet:

- Discogolvet ska kunna ta upp inkommande ljud.
- Det inkommande ljudet ska kunna delas upp i olika frekvensintervall.
- Olika lysdioder ska blinka beroende på vilka frekvenser som finns representerade i det inkommande ljudet.
- Lysdioderna ska kunna blinka i takt till musiken.

För att kunna uppfylla dessa krav krävs:

- En mikrofon som kan ta emot ljud (olika frekvenser).
- Tre stycken lysdioder i olika färger (grön, röd, gul).
- Tre olika filter; ett låg-, ett band- och ett högpasfilter som kan dimensioneras utifrån önskade intervall.
- Tempot definieras av tiden mellan två vågtoppar i det inkommande ljudet. Således kommer dioderna lysa hälften av tiden mellan två toppar och vara släckt resterande tid.
- En resetknapp används för att kunna nollställa systemet.

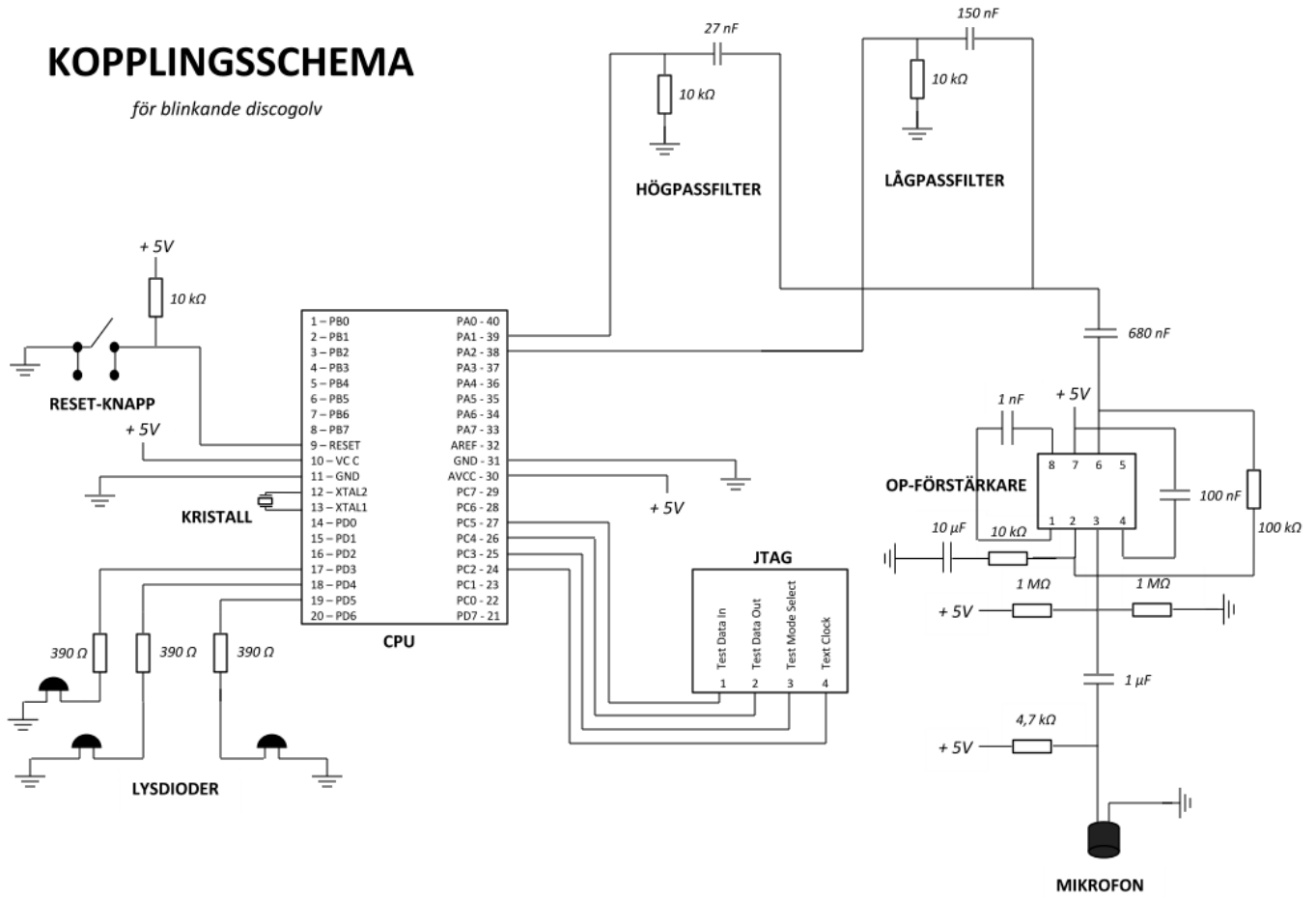
I mån av tid

- Montera en transparent platta över dioderna för att göra en liten prototyp av en golvplatta i det riktiga discogolvet.
- Skapa en databas som lagrar en låts ljusspel och som kan repeteras vid ett senare tillfälle.
- Installera en högtalare och inspelningsanordning som är kopplad till databasen. Denna ska kunna spela in och återspela musiken samt dess ljusspel.

9.2 Kopplingschema

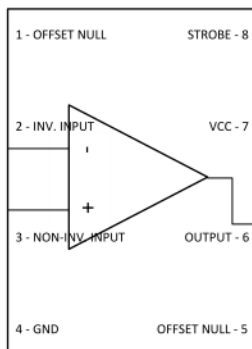
KOPPLINGSSCHEMA

för blinkande discogolv



OP-FÖRSTÄRKARE

inzoomad



9.3 Källkod

```
#include <avr/io.h>
#include <avr/interrupt.h>

char lowValue;           //värde från AD-omvandlaren hämtar från låg
char highValue;         //värde från AD-omvandlaren hämtar från hög
char limitLow = 0b00000111; //gräns för inkommande frekvenstoppar - låg
char limitHigh = 0b01100000; //gräns för inkommande frekvenstoppar - hög
int flag;               //siwtshar mellan hög och låg
int countLow;          //antal toppar för låg som är över limit
int countHigh;         //antal toppar för hög som är över limit
double countTime;     //antal 10 ms för att mäta antal toppar över ett intervall
double countTimeLow;  //antal 10 ms gul lysdiod ska lysa/vara släckt
double countTimeHigh; //antal 10 ms röd lysdiod ska lysa/vara släckt
double tempoLow;      //tempot gul lysdiod ska lysa i
double tempoHigh;     //tempot röd lysdiod ska lysa i

void setup() {

    DDRA = 0b00000000; //port a alla ingångar
    DDRD = 0b00111000; //port d utgångar där dioderna sitter

    TCCR0 = 0b00000101; //intern klocka med frekvens 1 MHz/1024
    TCNT0 = 98,75;      //initieringsvärdet, där timern börjar räkna ifrån
    TIMSK = 0b00000001; //starta klockan

    flag = 0;          //initiera lågpass

    countLow = 0;
    countHigh = 0;
    countTime = 0;
    countTimeLow = 0;
    countTimeHigh = 0;

    tempoLow = 0;
    tempoHigh = 0;
}

ISR (TIMER0_OVF_vect) { //tidsavbrott

    TCNT0 = 98,75;      //rensar initieringsvärdet
    TIFR = 0b00000001; //startar nytt avbrott

    if (flag == 0) { //vid lågpass

        ADMUX = 0b00100010; //sätter ingång till lågpassfilter
        SFIOR = 0b00000000; //vill inte ha någon autotrigger
        ADCSRA = 0b11011100; //startar AD-omvandlaren

    }

    else if (flag == 1) { //vid högpass

        ADMUX = 0b00100001; //sätter ingång till högpassfilter
        SFIOR = 0b00000000; //vill inte ha någon autotrigger
    }
}
```

```

        ADCSRA = 0b11011100;           //startar AD-omvandlaren
    }

    countTime++;
    countTimeLow++;
    countTimeHigh++;
}

ISR (ADC_vect) {                       //A/D-avbrott

    if (ADMUX == 0b00100010) {        //lyssnar på AD-omvandlaren

        lowValue = ADCH;               //sparar värdet från lågpasfiltret
        ADMUX = 0b00100001;           //slår om till högpasfilter
        ADCSRA = 0b10011100;          //nollställer flaggan
        flag = 1;                      //ställ om till högpas
    }

    else if (ADMUX == 0b00100001) {    //lyssnar på AD-omvandlaren

        highValue = ADCH;              //sparar värdet från högpasfiltret
        ADMUX = 0b00100010;           //slår om till lågpasfilter
        ADCSRA = 0b10011100;          //nollställer flaggan
        flag = 0;                      //ställ om till lågpas
    }

}

void main(void) {

    setup();                            //initierar portarna
    sei();                               //startar avbrotten

    while(1) {

        if (lowValue > limitLow) {     //då värdet från lågpas är högre än limit

            countLow++;

        }

        if (highValue > limitHigh) {   //då värdet från högpas är högre än limit

            countHigh++;

        }

        if (countTime >= 200) {        //efter två sekunder ska tempot räknas om

            tempoLow = countLow/60;     //tempot för lågpas
            tempoHigh = countHigh/60;   //tempot för högpas
            countTime = 0;              //nollställer countTime
            countLow = 0;               //nollställer topparna för lågpas
            countHigh = 0;              //nollställer topparna för högpas
        }
    }
}

```



```

}
if (countTimeLow < (1000/tempoLow)) {
    PORTD = 0b00001000; //gul lysdiod ska lysa under detta intervall
}
if (countTimeLow >= (1000/tempoLow) && countTimeLow < (2500/tempoLow)) {
    PORTD = 0b00010000; //gul släckt och grön lysa under detta intervall
}
if (countTimeLow >= (2500/tempoLow)) {
    countTimeLow = 0; //nollställer tiden för intervallet
}
if (countTimeHigh < (100/tempoHigh)) {
    PORTD = 0b00100000; //röd lysdiod ska lysa under detta intervall
}
if (countTimeHigh >= (100/tempoHigh) && countTimeHigh < (600/tempoHigh)) {
    PORTD = 0b00010000; //röd släckt och grön lysa under detta intervall
}
if (countTimeHigh >= (600/tempoHigh)) {
    countTimeHigh = 0; //nollställer tiden för intervallet
}
}
}

```