[År]

King Mangal

[Firmanavn]

[Dato]

Mikrokontroller



Innhold

[Arduino 1](#_Toc7354416)

[Digitale pinkoder 1](#_Toc7354417)

[analoge pinkoder 2](#_Toc7354418)

[Pin 0 og 1 på digital side av arduino 2](#_Toc7354419)

[Generelt 2](#_Toc7354420)

[Analog pinkoder 0 til 5 2](#_Toc7354421)

[andre pinner 2](#_Toc7354422)

[Reset 3](#_Toc7354423)

[GND 3](#_Toc7354424)

[3](#_Toc7354425)

[// 3](#_Toc7354426)

[Fro eksempel 3](#_Toc7354427)

[Setup 3](#_Toc7354428)

[Loop 4](#_Toc7354429)

[voice recognition module 5](#_Toc7354430)

[Serial.begin 5](#_Toc7354431)

[pinMode 5](#_Toc7354432)

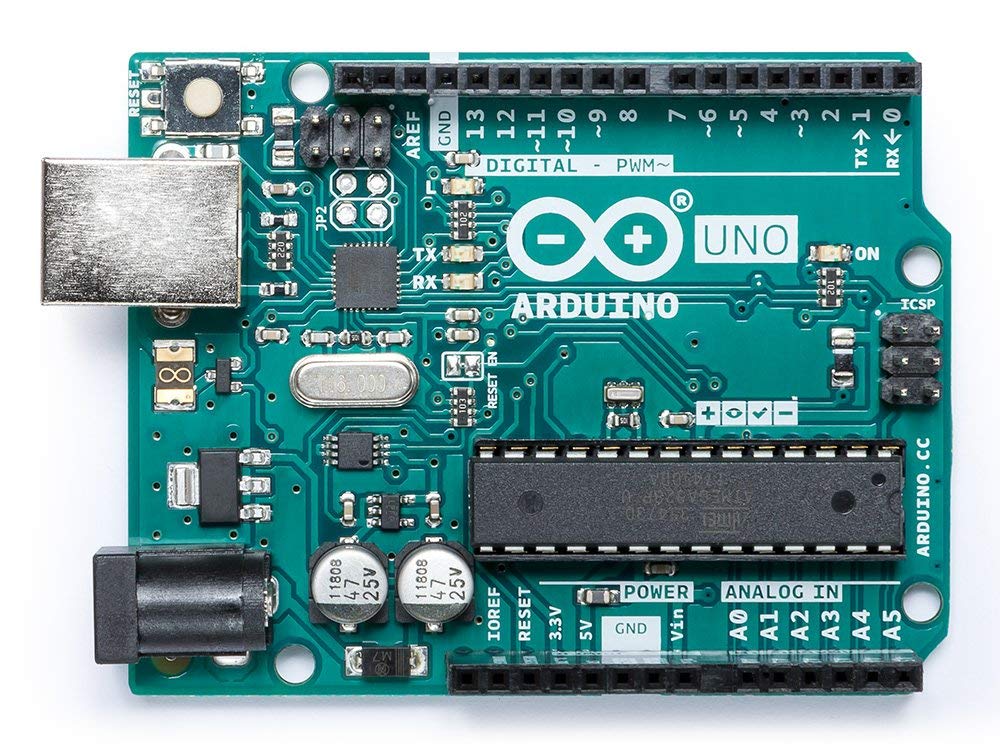
[Delay 5](#_Toc7354433)

[Serial Write 5](#_Toc7354434)

[Analog Write 6](#_Toc7354435)

[Oppkobling 6](#_Toc7354436)

# Arduino

arduino er åpen kildekode maskinvare, noe som betyr at alle designfiler for dette kjeder har blitt offentliggjort. så noen kan gå og lage kloner av Arduino og mange mennesker har gått og laget kloner til Arduino og solgt dem. mange kloner er veldig gode, og det er på en måte hele ideen med åpen kildekode maskinvare, at du kan åpne ting slik at folk kan dele.

Figur 1

# Image result for digital pin headersDigitale pinkoder

Figur 2 digital



Digitale pinkoder er plastikk og det er en mengde tall ved siden av den (vist på figur 2), og de er nummerert fra 0 til 13 fra høyre til venstre. Det er faktisk 14 små hull der som er nummerert. De gir deg tilgang på de analoge pinnene på motsatt side av Arduino.

# Image result for arduinoanaloge pinkoder

Figur 3 analog

Disse analoge pinkodene kan brukes enten som innganger, slik at de for eksempel kan lese nivåer av spenning fra noe. Eller kan de brukes som utganger, hvor de kan sende ut en spenning på 5 eller en spenning på 0 volt.

Digitale pinkoder som 3, 5, 6, 9, 10 og 11 kan bruke pulsemodulasjon, som i utgangspunktet betyr at de kan justere spenningen de bruker mellom 0 og 5 volt. I disse hullene kan du stikke motstander eller du kan stikke lysdioder. Det igjen fører til elektrisk kontakt mellom de elektriske komponentene.

# Pin 0 og 1 på digital side av arduino

Hvis du ser nøye på en av bildene over, kan du se at pin 0 har en RX og at pin 1 har TX. Det står for å motta og overføring. Så når du snakker med datamaskinen eller en annen enhet med arduino, er det disse pinnene som blir utnyttet.

# Generelt

Hvis du skal gjøre mye kommunikasjon eller seriell kommunikasjon med arduino på et bestemt prosjekt, er det best å ikke benytte seg av disse pinnene for mye. fordi det kan påvirke hvordan programmet fungerer.

# Analog pinkoder 0 til 5

Figur 4

De analoge pinnene o-5 gir deg tilgang til noe som kalles analog til digital omformer og den er på denne mikrokontrolleren. Hva analog til digital omformer lar deg gjøre, er å ta analoge signaler og konvertere dem til digitale signaler. Så disse hullene gir deg mulighet til å konvertere signaler.

# andre pinner

Figur 5

på figur 5 er en del andre hull som jeg skal beskrive. Det første vi skal se på er V, så på bilde kan vi blant annet se 3v3 som står for 3,3 volt og det 5v som står for 5 volt. Så hvis du har din arduino koblet til enten en strømforsyning som et batteri eller koblet til datamaskinen din med en USB-port, med en USB-kabel, kan du få enten 5 volt eller 3.3 volt fra disse pinnene. Det betyr at hvis du kobler en leder til en 5 volt hullet og måler slutten av ledningen, så vil multimeteren vise 5 volt. Det masse med 3.3 volt.

# Reset

Når du presser reset knappen så vil Arduino starte fra starten av programmet. Det betyr at du ikke sletter noe som helts, dersom du trykker på knappen, den bare omstarter hele programmet og det gjør den super raskt.

# GND

Det er tre jordpinner. Det er to på bunnen ved siden 3,3 og 5 volt og det er en på toppen, og det er ved siden av digital pin 13. jordpinner gir deg tilgang til de laveste spenningene på brettet.

Det er også en (power) på LED. Lysdioden lyser når du tilfører strøm til arduino bord enten via en USB til en datamaskin eller en ekstern batterikapasitet.

# 

Figur kode

# //

Her er en kode og det første jeg skal snakke om er skråstrekene (//). Når det ser slike linjer er det alltid en komment bak. Denne kommenten forteller kompilatoren om å ignorere denne linjen fra resten av koden.

I teksten finner vi også en skråstrek med stjerne på toppen (/\*) også masse tekst og avslutter det igjen med en skråstrek med stjerne. Hva betyr alt dette? Jo, kommentarer gjør det letter for deg å gjenkjenne en fremtidig versjon av deg selv. altså kommentarer gjør det lettere for deg å huske hva du gjorde akkurat da du skrev et stykke kode.

# Fro eksempel



Figur eksempel

Her har vi først en tekst og noen variabler under. Nå skal jeg forklare nærmere viktigheten med denne teksten som vi snakket om litt tidligere. Hvis vi ikke hadde hatt denne kommentaren, så hadde vi lurt på hva (pushButton) betyr. Men siden vi har denne kommentaren som sier at (digital pin 2 har en trykknapp festet til den, gi det et navn, skjønner vi at pushbutton er et navn. Int står for (integer)

Legg merke til at det er semikolon etter 2 tallet. Semikolon er for å markere hva en periode er. Så på slutten av alle setninger av kode skal du ha et semikolon, som lar kompilatoren vite at du er ferdig med kodens setning. Dersom du ikke har med semikolon, vil det vises (error)

Hvis du ser igjen på figur 6, er det også en del andre tegn som () og []. Før jeg forklarer hva disse symbolene står for, skal jeg forklare hva funksjonene (setup) og (loop) gjør.

# Setup

Disse er to svært vanlige funksjoner som du vil bruke i nesten alle arduino programmer som du skriver. En funksjon innkapsler et ekstremt nyttig stykke kode og reduserer det til et enkelt søkeord, noe som gjør at du raskt kan implementere det stykke kode. Setup funksjon setter opp programmet. Når programmet kjøres, går det bare en gang i setup funksjonen, det betyr at alle ting i oppsettfunksjonen blir kjørt en gang og da er du ferdig med det.

# Loop

Loop funksjonen er en veldig viktig del av koden. Loopen går om og igjen og om og igjen. Loop brukes altså til å gjenta en ting.

Så nå kan vi snakke om symbolene. Det er alltid åpen og lukket parentes etter hver funksjon (). Disse representerer start og slutt på noe. Se figur 6.

Og hvis du lurer på hva void foran setup betyr, så betyr det at denne funksjonen ikke returnerer noe tilbake til programmet. Men alt du trenger å vite er at det er void foran (setup) og det er void foran (loop).

Så la oss nevne hva disse [] betyr. Så det disse tegnene gjør er at de omslutter ytterligere instruksjoner utført av funksjonen. Det er alltid en åpen brakett og det er alltid en lukket brakett.

# voice recognition module

Figur 8 accesPort

Det første du må gjøre er å laste ned Acessport. Etter at vi lastet ned appen er det første vi må gjøre er å ta opp stemmeinstruksjoner / kommandoer. Hver stemme instruksjon har maksimal lengde på 1300ms (1.3sec), som sikrer at de fleste ord kan spilles inn. Når du begynner å spille inn, kan du ikke stoppe opptaksprosessen til du har fullført alle 5 instrukser / kommandoer for en gruppe. Når du begynner med opptak, blir også det forrige innholdet i den gruppen slettet.

Vi vil registrere gruppen 1 med 5 kommandoer: Hvit, Rød, Blå, Grønn og AV. Åpne AccessPort-appen og start seriekommunikasjonen ved 9600 bps. (klikk på bildet). Klikk nå på hex og skriv AA 11 og klikk på send-knappen. Dette vil starte opptaksprosedyren for den første gruppen.

I dette trinnet er tiden nøkkelen ... Du vil bli bedt om å registrere hver kommando (hvit, rød etc) tre ganger.

# Serial.begin

Dette betyr ikke noe mer enn hvor mye bits Arduino overfører per sekund.

# pinMode

Denne funksjonen brukes til å konfigurere en bestem pin fo å oppføre seg enten som en inngang eller som en utgang.

# Delay

Måten denne funksjonen fungerer på er ganske rett frem. F.eks. når du forsinker (1000), stopper Arduino på den linjen i 1 sekund. Forsinkelsen er en blokkeringsfunksjon. Blokkeringsfunksjon forhindrer et program fra å gjøre noe annet før den bestemte oppgaven er fullført.

# Serial Write

Denne er der for å snakke binært, en byte om gangen og sender byte til serieporten.

# Analog Write

Analog Write brukes til å oppdatere statusen til analoge pinner og brukes til å adressere PWM-pinnene på brettet.

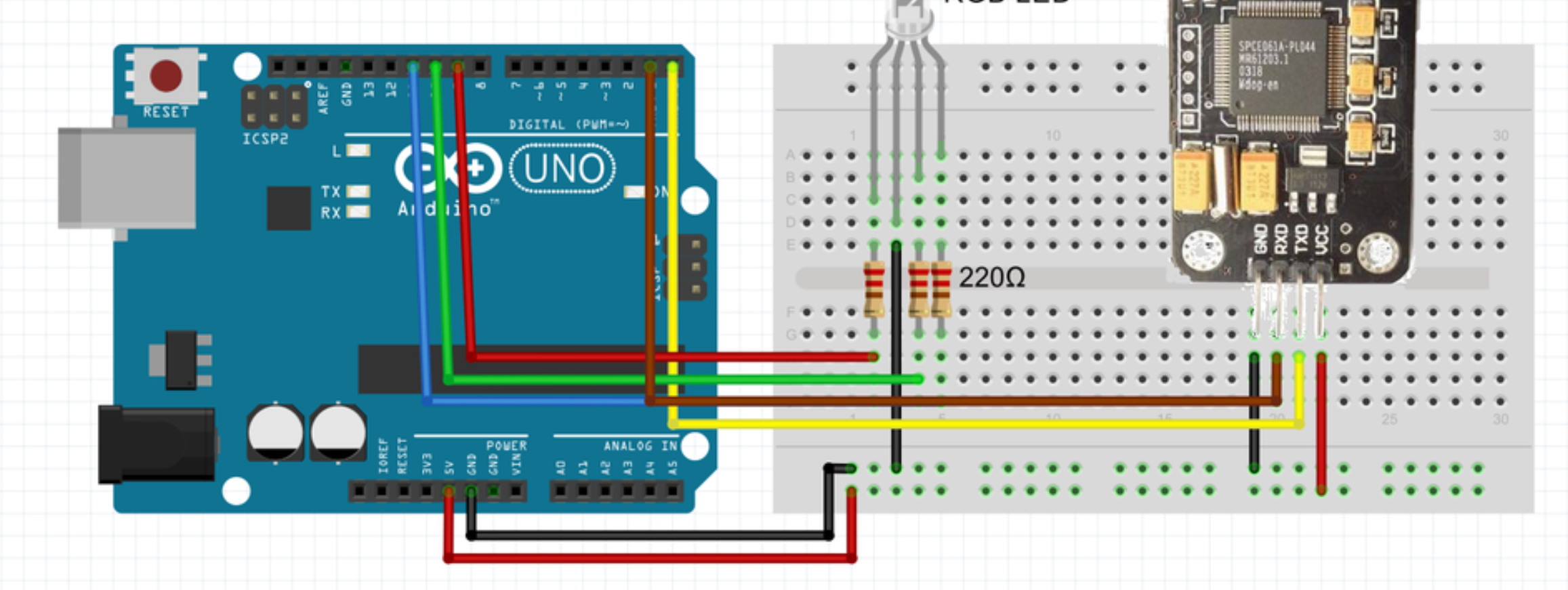
PWM-pinnene er 8-biters pinner som betyr at du kan sette arbeidssyklusen mellom 0-255.

Når du er ferdig med det må du koble opp og det gjør du slik

# Oppkobling

Hva du trenger:

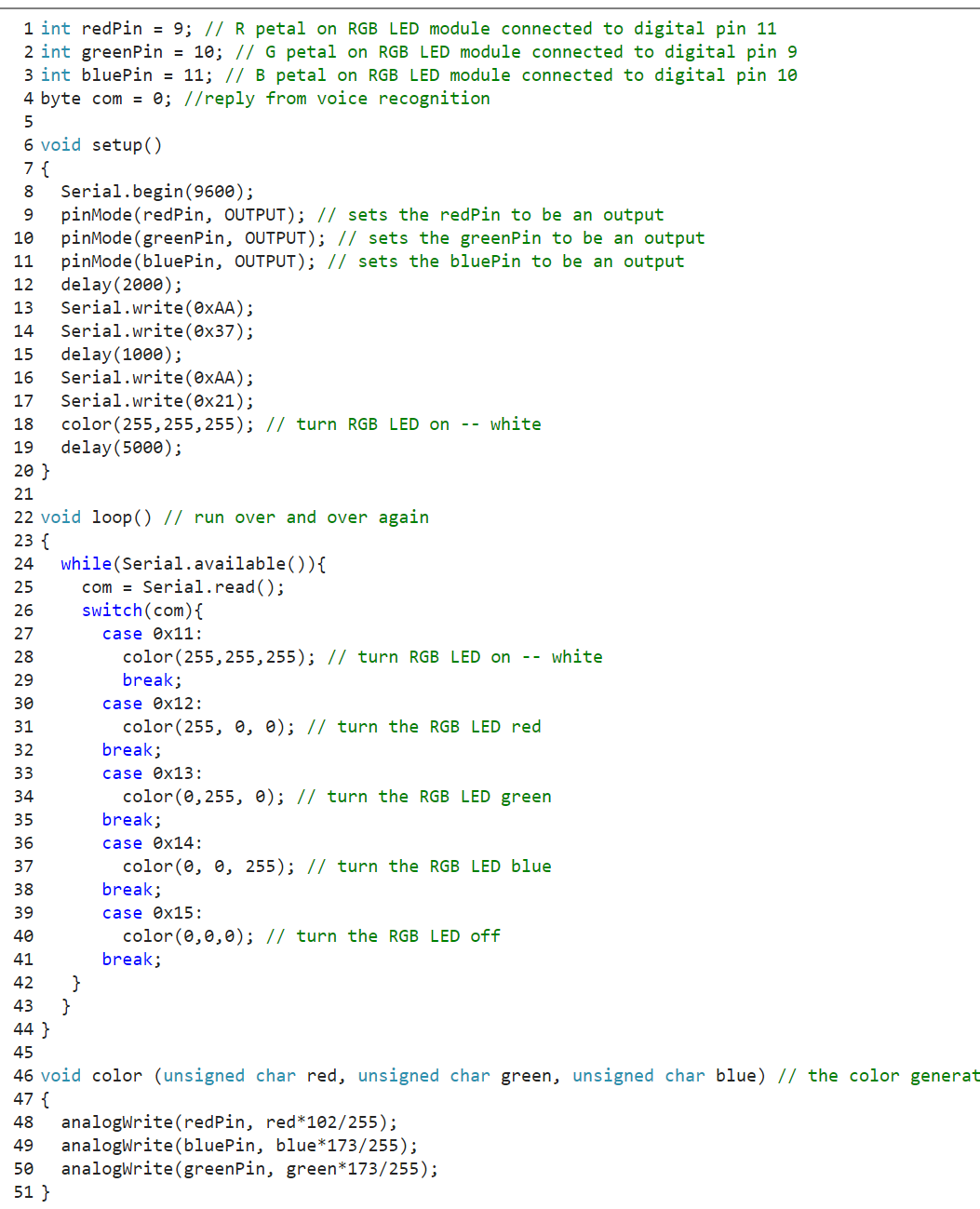
* Arduino uno
* Breadboard
* Voice Recognition Mdule (serial)
* Michrphone
* RGB LED
* 3X220 Ohm resistors



Figur 9 skjema

som du ser er det ikke så vanskelig å koble opp.

Og her kommer koden.



Figur 10 koden

Når du har alt dette på plass og har sendt koden til arduino. Så vil det skje en del kule ting. Nå kan vi snakke til mikrofonen også vil de lysene som vi har kodet lyse og du kan slå dem av og på ved å snakke til mikrofonen.