Fredrik Kartevoll

Innhold

[Introduksjon 2](#_Toc498523339)

[Arduino nano istedenfor teensy 3](#_Toc498523340)

[Bruk av terminalen i arduino 3](#_Toc498523341)

[IMU mot arduino 3](#_Toc498523342)

[I2C kommunikasjon 3](#_Toc498523343)

[Lese av verdier på IMU 3](#_Toc498523344)

[G-kraft sensor 3](#_Toc498523345)

[Konklusjon 4](#_Toc498523346)

[Kilder 5](#_Toc498523347)

# Introduksjon

I del 2 av prosjektet skal vi bruke IMU’en MPU 6050 GY-521

# Innføring i teensy 3.6

I denne oppgaven skal vi ta i bruk en Teensy 3.6. Teensy 3.6 er en USB-basert mikrokontroller som er Arduino-kompatibel. For å programmere Arduinoprogrammer på Teensyen laster vi ned tilleggsbiblioteket Teensyduino. Man må også sørge for at board type i arduino IDE er satt til Teensy 3.6.

## Oppkobling av teensy 3.6

Teensy 3.6 kobles mot PC med mini-usb kabel. Denne kabelen tilfører Teensy 3.6 med drivspenning og den kan både sende og motta data.

For å teste forbindelsen mellom Teeny og PC, lastet vi opp eksempelprogrammet «tutorial 1 – blink». Programmet ble lastet opp til Teensey uten feil og lysdioden på Teensey blinket hvert andre sekund.

## Bruk av terminalen i Arduino IDE

Arduione IDE inneholder terminal funksjonen. Denne kan være et viktig verktøy for feilsøking ved arduino programmering. Man kan bruke terminalen for å sjekke verdier på variable, inndata, utdata og mye mer.

For å skrive til terminalen må man først kjøre Serial.begin(sett inn baud rate her); Det er da viktig at man bruker samme baud rate i overvåkningterminal i Arduino IDE som man printer med Serial.begin funksjonen. Når man har kjørt Serial.begin funksjonen, kan man bruke Serial.print(det man vil printe her) for å printe informasjon til terminalen.

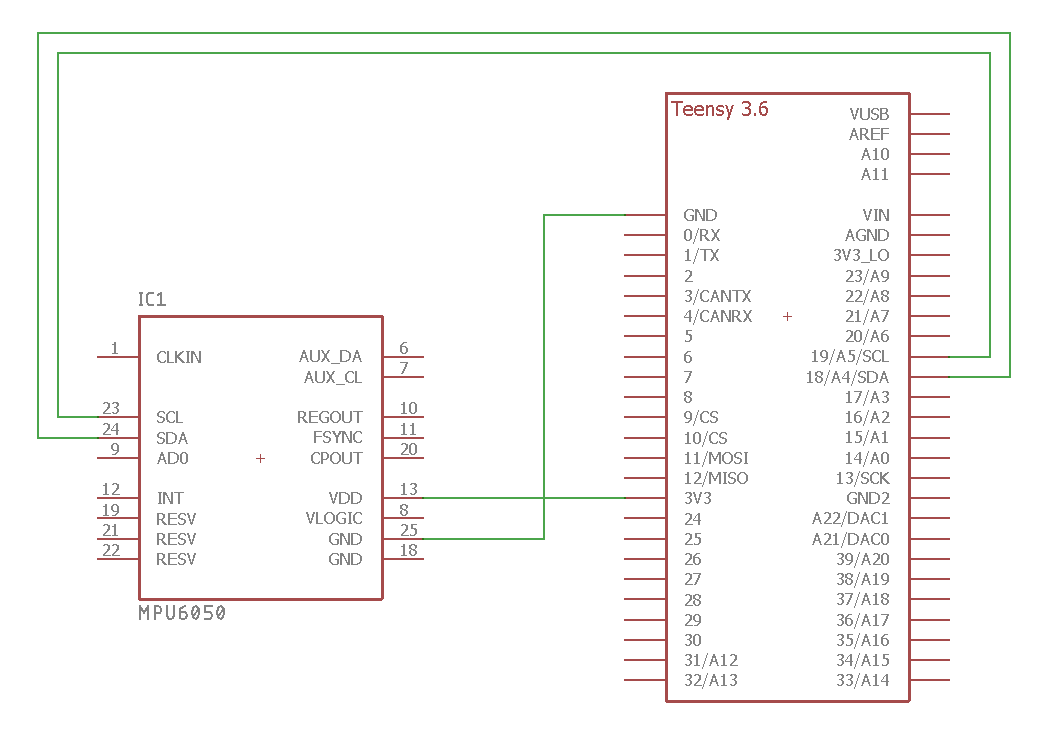
For å teste om det fungerte å skrive til terminalen, lagde vi et program som printet meldingen «Hello serial monitor» kontinuerlig med en forsinkelse på 1 sekund.

# IMU mot Teensy 3.6

Vi skal nå koble en IMU (Inertial Measurement Unit) mot vår Teensy 3.6. IMU’en vi bruker er en MPU 6050 GY-521. Denne har 3 gyroskop som kan måle vinkelhastighet, 3 akselerometer som kan måle akselerasjon og en temperatursensor.

I databladet til IMU’en leste vi at den skulle ha en drivspenning i området 2,375V til 3,46V. Vi valgte derfor å bruke Teensy’ens utgangsspenning på 3,3V til drifte IMU’en.

Mellom Teensy og IMU blir det brukt I2C kommunikasjon. Denne kobles slik at SCL og SDA pinne på IMU går til SCL og SDA pinne på Teensy (se figur NR). AD0 kan bli brukt for å endre I2C adressen til IMU’en. Hvis AD0 er høy(3,3V) vil IMU’en har I2C adresse 0x69. Vår AD0 er lav vil da gi IMU’en I2C adresse 0x68. Hvis man vil bruke flere IMU’er kan man da variere I2C adressene syklisk slik at man kan aksessere IMU’ene en etter en. Med en høy syklusfrekvens kan man da ha flere IMU’er på samme I2C bus-system.



## Lese av verdier på IMU

For å lese verdier fra IMU’en bruker vi wire funksjonen til å printe verdiene til terminalen i Arduino IDE.

Vi sjekket akselerasjonsverdien i Z retning forårsaket av tyngdeakselerasjonen. Når IMU’en lagg stille målte vi verdien til ca -18000. Basert på denne målingen lagde vi et program der vi fikk meldinger til terminalen ut ifra målte verdier. Hvis verdien var over 10000 fikk vi melding om at brikken var opp ned. Hvis verdien var mindre enn -10000 fikk vi melding om at brikken var i normal stilling.

# g-kraft sensor

Vi skal nå lage et program som skal måle endringer i g-krefter. Hvis man får en større endring enn 1g på ett sekund skal lysdioden på Teensy’en toggles.

Programmet vi har laget er laget med en for løkke som kjører 10 ganger, eller frem til led toggles. I løkken vil akselerasjonsverdien i Z retning bli lest av IMU’en. Løkken vil kjøre hvert 100ms, hvor den leser inn

Skalering mot g. kan være +- 2, 4, 6, 8 eller 16g leser i datablad. Skalering mot g er 32768/maks g.

programbeskrivelse

# Konklusjon

# Kilder

Nano pinout <https://forum.arduino.cc/index.php?topic=147582.0>

MPU 6050 datasheet <http://www.hotmcu.com/gy521-mpu6050-3axis-acceleration-gyroscope-6dof-module-p-83.html>

Wirelib info <https://www.arduino.cc/en/Reference/Wire>

Teensy 3.6 pinout: <https://www.pjrc.com/teensy/pinout.html>

Teensyduino software: <https://www.pjrc.com/teensy/teensyduino.html>

Arduino mot MPU: https://playground.arduino.cc/Main/MPU-6050