

Emnekode: PG5501
Emnenavn: Embedded systems
Innleveringsdato: 19.03.2023
Arbeidskrav



Høyskolen Kristiania
Semester Vår 2023

Denne besvarelsen er gjennomført som en del av utdannelsen ved Høyskolen Kristiania.
Høyskolen er ikke ansvarlig for oppgavens metoder, resultater, konklusjoner eller anbefalinger.

Innhold:

Liste over komponenter

Oppkobling av prosjekt

Fremgangsmåte

Analyse

Planlegging

Problemstillinger underveis

Oppkobling og kildekode

Funksjonalitet som ikke ble inkludert

Vedlegg, referanser

Kilder

Liste over komponenter:

Arduino UNO	1 stk.
Brødkort full+	1 stk.
Brødkort mini	1 stk.
240x135 Color TFT Display	1 stk.
Clock ZS-042	1 stk.
RFID RC522	1 stk.
Infrared I/O sensor HW-477	1 stk.
Motion sensor HW-416-B	1 stk.
Sound sensor SY-M213	1 stk.
Jumper wire	32 stk.

Oppkobling av prosjekt kan sjekkes opp i vedlegg 1.

Framgangsmåte**Analyse:**

I denne arbeidskravet ble det ikke gjort en større analyse grunnet naturen i problemstillingen.

Vi skal lage et alarm system som skal kunne vise tidspunktet noen «brøt» inn på.

Uten noen spesifikke moduler eller drivere å undersøke i dybde på gikk framgangsmåten over direkte til planlegging.

Planlegging:

Med å sammenligne hva et ekte alarmsystem har konkluderte man med følgende funksjonaliteter:

- Senorer som trigger alarmsystemet.
- Slå av og på systemet.
- Aktivere og deaktivere når sensorer skal trigge.
- Logg som lagrer når alarmer går av.
- Tastatur og nøkkelring for identifisering

Listen ovenfor gir en klar speiling på en ekte liste, men også en stort prosjekt å fullføre. Som senere i rapporten dokumentere så blir ikke alt fullt implementert og noe ble ikke legget til i det hele tatt.

For sensorer ble Bevegelse og lyd sensor benytter. Tanken er at man trenger ikke å ha en videokamera for å vite om noe beveger seg. Siden vi har begrenset med ressurser vill en bevegelse sensor spare masse her. Lyd sensor faller naturligvis noen bryter inn med kraft, eksempelvis igjennom vindu eller å bryte ned en dør.

For tastatur var valget mellom infrarødt kontroller eller 4x4 tastatur. Siden vi skal benytte både SPI og I2C protokoller ble det begrenset med pinner tilgjengelig. Så her ble IR kontroller benytter for å spare pinner og bedre forståelse på hvordan IR fungerer med embedded systemer.

Det å ha med RFID ble bevist gjort for to grunner. Ene var for å lære å sette opp SPI med flere eksterne enheter og dypere forståelse for RFID.

TFT skjer som benyttes skal ikke bruke grafiske elementer. Begrunnelsen er at jeg har lite erfaring med slike biblioteker og funksjonalitet skal prioriteres over fine effekter på skjerm. Det innebære ingen bruk av SD-kort og en notering i oppkobling.

Problemstillinger underveis

Selve oppkoblingen var det lite problemer, men det første som kom opp var TFT skjerm.

Med lite erfaring innen bruk av grafiske biblioteker tok det lang tid å få til vanlig bruk av skjermen.

Det ble valgt å ikke benytte SD kort for grafiske bilder og i stede benytte vanlig tekst og fargete bakgrunn.

Med denne skjermen ble en metode funnet som ikke gjør som den sier: `invertDisplay()`. Denne skal speile fargene hvis argument er «true» og vise normal farge med «false». I min kode ble denne beskrivelsen motsatt. Hypotese en er at oppsetter noe feil eller hypotese to er det er bug i biblioteket.

En spennende problemstilling som kommer til bruk av IR kontroller og buzzer. Det ble senere vurdert bruk av buzzer for å signalisere at alarmen har gått av, men det gjorde at IR kontroller ikke kunne bli lest av IR sensor. Det ble landet på tre hypoteser:

1. Sensor og Buzzer benytter så mye strøm at Arduino sin strømgiver ikke klare å forsyne nok. Dette medfører at lesing av IR kontroller ikke fungerer.
2. CPU'en til Arduinon klarer ikke å sende signal til buzzer og ta i mot signal fra IR kontroller som mistenkes å være ganske høyt krevende.
3. Buzzer forurenses med IR signaler som er den minst sannsynlige.

Oppkobling, kildekode og biblioteker:

I denne oppkoblingen ble det benyttet to prokoller: SPI og I2C.

SPI ble benyttet av TFT skjerm og RFID leser. Viktig å notat er at MISO for TFT skjerm er ikke benyttet grunnet at det skal kun sendes signaler til skjermen. Skal SD kort benyttes må MISO kobles inn.

I2C ble benyttet av klokke enheten som gir tiden i time/min/sek dag/måned/år.

Resten er koblet opp direkte til Arduino-brette på vanligt vis.

Benyttede biblioteker:

- Adafruit_BusIO
- Adafruit-GFX-Library
- Adafruit-ST7735-Library
- Adafruit_Seesaw
- Arduino-Irremote
- rfid
- Rtc
- SD

Lenker til kilder kan finnes med følgende steg:

1. Gå til kildemappen med terminalen.
2. Skriv «pio home» og åpne gitt lenke
3. Last inn prosjekt og sjekk fanen for biblioteker

Funksjonalitet som ikke ble inkludert:

- Slå av og på er ikke fult implementert. Det eneste som skjer når man skur den av er at skjermen er svart.
- Sikker kontoer for pin og dongel. Dette er hardkodet inn og har ingen måte å legge eller fjerne brukere, men fungerende innlogging er tilstede.
- Menyvalg for å sjekke tidligere alarmer som har gått av. Var ønsket å benytte EEPROM til ZS-042.
- Buzzer som lager lyd når alarmen går, fant ingen løsning som gjør at man kan benytte IR kontroller og buzzer samtidig.

Referanse, vedlegg og bibloteker:

Vedlegg 1	SecurityAlarmFritzing.png	-	Fritzing tegning over prosjektet.
Vedlegg 2	SecurityAlarm.zip	-	Kildekode til prosjektet. Skrevet med IDE Clion og byggeverktøy PlatformIO.

Kilder:

Ingen kilder referert direkte i rapporten. Kilder til biblioteket blir funnet i kildekoden.