

Mandag d. 3. september

Kigget på juraen og fundet processer og love for lastning og losning af bulkskibe

Mandag d. 10. september

Skrevet kvalitetsfaktorer

Tirsdag d. 11. september

Arbejdet på ordlisten

Onsdag d. 12. september

Læst 2. opinion på aktørbeskrivelsen

Søndag d. 23. september

Sat RPi op med SSH, FTP, VNC. Startet på GPIO undersøgelse.

Onsdag d. 26-9

Jeg har de sidste par dage arbejdet på hvordan vi får i2c til at virke på tærten. Det virker umiddelbart til at være en stor opgave hvor der enten skal læres python eller skrives kernemoduler. Det lader ikke til at der er så mange der har rodet med det.

Vi har fået lov til at låne en switch af HelpDesk således at vi kan begynde at arbejde med tærten på skolen også

Mandag 01-10-12

Da der ikke er skrevet tilstrækkelige drivere til I2C på Raspberry PI har vi valgt at skifte til seriel communication. Vi kunne i princippet godt have skrevet vores egne drivere, men det er ikke et ufravigeligt krav at anvende I2C hvorfor vi ikke prioriterer det nok til at vi bruger tid på at skrive vores egne drivere.

Jeg har undersøgt hvorledes timere virker i Qt og fik efter cirka fire timer et forsøgsprogram til at virke. Timeren er sådan set nem nok at bruge - når blot man lige får det til at lykkedes første gang. Problemet lå i hvorledes funktionen der skulle kaldes blot kunne være en almindelig funktion eller om den skulle være et public slot (det sidste viste sig at være det rigtige)

Mandag d. 08-10-12

Har arbejdet med GUI'en hele dagen. Selve GUI'en er lavet, og der er kommet placeholdere til en stor del af de funktioner der mangler. Timeren virker også som den skal, og der er mulighed for at sende til databasen hver 10. minut.

Der skal laves en handler til events når der bliver trykket på knapper.

Umiddelbart ser det lyst ud for programmet

Der er også blevet lavet en aktivitetslog således at man kan følge programmet.

Tirsdag d. 09-10-12

Har brugt fire timer her til aften på at undersøge mulighederne for FTP i Qt ved hjælp af QFtp (indbygget i Qt Creator). Det viser sig at være blevet ført over i QNetwork/

QNetworkAccessManager hvor dokumentationen er ret dårlig. Det burde dog ikke være noget

problem at skrive en ftp-klient uden om Qt og så anvende den. Det vil jeg kigge nærmere på når jeg har helt styr på IKN Øvelse nr. 8.

Alternativt har jeg fundet et plugin med sourcekode (QFtp) som muligvis kan anvendes. Der skal snakkes med Mick om hvordan det skal overføres ifht. om der skal appendes på en fil, overføres en fil eller sendes beskeder over FTP (eller SFTP?) - eller telnet?.

Derudover virker det ikke umiddelbart som så nemt at cross-compile til Raspberry Pi som jeg havde håbet. Programmet er skrevet i Qt4.x, men krafterne ligger fra Qt-udviklerne på at det skal være Qt5 som er kompatibel med RPi. Derfor kan det blive noget bøvl med Qt4. Det bør overvejes hvorvidt RPi er vigtig at anvende eller om der blot kan anvendes en bærbar med linux kørende og UART til PSoC.

10-10-12

Lidt om GUI'en

GUI'en ser således ud pt. Funktionaliteterne er implementeret i den og den er som sådan færdig. Dog kan det tænkes at den rent visuelt skifter karakter. Det er dog en ren æstetisk opgave og er derfor ikke prioriteret.

IGANGVÆRENDE	KOMMANDOER	STATUS
Trimning Hældningsregulering	<div>TRIM</div> <div>GETCAR AUTO</div> <div>%<div>JUSTER MANUELT</div></div> <div>Alarm trigger</div>	Skib Hældning: 5% Venstre vandtank: 5% Højre vandtank: 5% System Dataforbindelse SM-forbindelse Fejl: 0xFF
SENESTE AKTIVITET		

Når en alarm bliver triggeret vil der komme en 'rød' dialogboks op som man skal lukke før at man kan fortsætte. Det der har udløst alarmen vil blinke rødt, fx hældningsprocenten eller dataforbindelsen. Alarmen for hældning, venstre vandtank og højrevandtank stopper ved to scenarier: værdierne falder under det "kritiske" niveau eller de afbrydes manuelt ved klik på det røde felt.

Alarmerne dataforbindelse og SM-forbindelse kan ikke afbrydes manuelt og stopper derfor kun hvis problemet løses (forbindelsen genoprettes).

Der er også blevet tilføjet en "seneste aktivitet" kasse hvor man kan se de seneste aktiviteter for programmet. Her kan man se når der har været udløst en alarm og denne er blevet afblæst / annulleret igen manuelt eller af systemet selv (fordi problemet blev løst). Man kan også se

at fx trimning er blevet iværksat og fuldført. Der er pt. ikke tidsstempel på og de nyeste er de nederste (bør være de øverste).

Når trimning fx er blevet iværksat vil den lyse gul indtil opgaven er fuldført. Det ville være mere intuitivt hvis de fadede ud når de ikke var aktive og bold når de var aktive.

10-10-12

Det blev til endnu en aftenforestilling med cirka 5 timers arbejde. Det har vist sig at Qt 4 repositoret og Raspberry Pi ikke er kompatible med hinanden. I stedet skal der anvendes Qt 5. <http://qt-project.org/forums/viewthread/21015/#100413>

Dette er dog stadigvæk i beta fasen (forventes at det udkommer til november). Det giver nogle problemer i forhold til hvor meget hjælp man kan få i forbindelse med problemer. Og problemer var der nok af da jeg skulle skifte. Jeg endte med et setup hvor jeg har to installationer af Qt Creator med hvert deres repositorium (hhv. Qt 4.x og Qt 5.0). Dette lykkedes til sidst. Eneste problem lige nu er at den nye installation af Qt Creator tror min computer er 32bit i stedet for 64bit. Det har dog intet at sige da release-udgaven selvfølgelig compiles til RPi.

Jeg vil snarest prøve at få Qt Creator opsat til at kunne compile til RPi. Der skal bl.a. bruges denne guide samt en video:

http://qt-project.org/wiki/RaspberryPi_Beginners_guide

http://www.youtube.com/watch?v=pBhT_CINpJE&feature=related

4. november:

Compiling til RPi er sat på hylden indtil videre da vi i princippet godt kan bruge en almindelig computer. Vi ønsker stadigvæk at bruge RPi, men det er vigtigere at programmet rent faktisk bliver færdigt.

Programmet er i dag/nat blevet fuldstændig omskrevet fra bunden. Der er kommet meget mere struktur i koden og der er blevet rodet gevaldigt op. Der er indført en del flere klasser for at uddelegere arbejdsopgaverne således at en bestemt klasse ikke fungerer som en stor rodekasse.

Koden er struktureret således at vi på KI har hovedklasser: Kontrolinterface, Styringsmodul, VBTE, Hældningssensor, Server. Strukturen er således den samme som i selve systemet. KI klassen kender kun til server og styringsmodulet. Styringsmodulet kender til KI, VBTE og hældningssensoren. KI kan så bede SM om en statusopdatering. SM beder så de enkelte enheder (klasser) om deres værdier. Hver af disse klasser (VBTE og hældningssensor) har en pointer til rs232-klassen som så står for uart-kommunikationen. Det er SM der opretter rs232-klassen og giver VBTE og hældningssensor en pointer til dette objekt når SM opretter de objekter. SM bruger rs232-objektet til at aktivere eller deaktivere automatisk hældning eller til at sende en besked om en manuel vinkling.

Når KI (klassen) spørger SM-klassen om en statusopdatering får den returneret en struct med alle variabler:

vbte1niveau

vbte1status

vbte2niveau

vbte2status

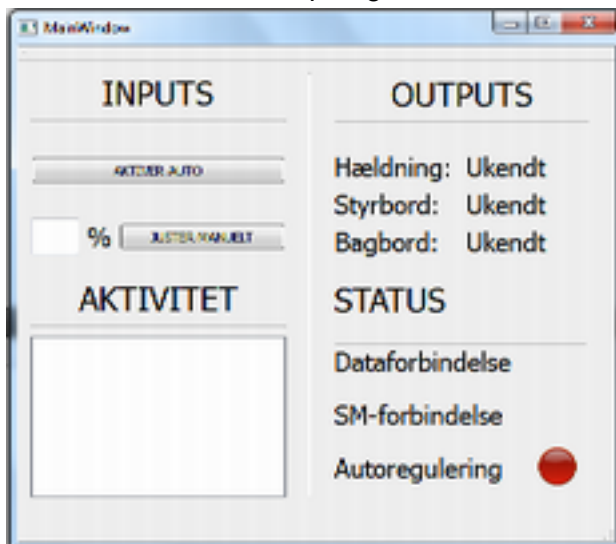
hældningsregulering

kommunikationsstatus (et 8bit-register?) som beskriver om der er forbindelse til sensor, vbte1 og 2.

INDSÆT FIGUR

7. november:

Jeg er begyndt at undersøge hvordan gui'en kan laves mere indbydende. Det er en lavprioritetsopgave og laves således kun når jeg gerne vil rode lidt med det i fritiden. I første omgang har jeg lavet vinduet mere kompakt og fjernet nogle overflødige elementer. Jeg vil lave en form for statuslampe ved henholdsvis dataforbindelse, forbindelse til styringsmodul og automatisk regulering. Det er så meningen at når programmet prøver at skabe forbindelse viser lampen gult. Ved succes skifter den til grøn, ved fejl skifter den til



rødt.

Her er gui'en som den ser ud netop nu. Autoregulering er her slået fra hvorfor "lampen" lyser rødt.