

1. Inledning

Denna laboration ger insikt i hur man programmerar nära hårdvara och även hur det är att programmera i par, målet som man vill nå varierar från person till person och personligen var målet att förstå så mycket att jag kunde förklara det så enkelt som möjligt för någon annan.

1.1 Bakgrund

En ingenjör som studerar vid ICT-skolan kan se denna uppgift som ett intro till lätt programmering nära hårdvaran. Detta då man programmerar i ett språk som liknar C kallat NXC, man får även lära sig att felsöka och lösa problem som båda två är ytterst viktiga egenskaper att ha som ingenjör. Man får även pröva på hur de är att jobba i grupp och tekniken "parprogrammering" som kan komma till nytta på otaliga tillfällen.

1.2 Syfte och målsättning

Syftet med laboration kan ses vara att man introduceras till programmering och även hur den utnyttjas i en robot, jag har tidigare kunskaper av programmering och såg de även som ett tillfälle att försöka göra mig förstådd då min partner ej hade samma förutsättningar, så mitt största mål var att lära mig hur NXC fungerar med roboten samt att kunna förklara allting så enkelt som möjligt.

2. Genomförande

Innan laborationen ens hade tagit sin början så var det ett par saker som man var tvungna att ha gjorda, dessa var att ladda ner både programvaran och drivrutiner för roboten. Dessa kunde lätt hittas under kursens material på bilda, då jag inte hade en laptop med Windows OS så fick min partner ladda hem dessa till sin bärbara. Detta skedde utan några trubbel. Både jag och min partner kollade även igenom den filen "Introduktion till NXC" som även den fanns på bilda under kursen för att få en översikt över språket.

När man väl var på plats så tog man en robot samt en kabel och kopplade ihop dessa med datorn som hade all programvara och man kunde börja laborera.

När väl laborationen var igång började man med att öppna ett dokument där koden för hur roboten skulle fungera redan fanns, det som var själva uppgiften var att få koden att fungera, detta betydde att roboten skulle följa en svart linje som gick in i en vägg, efter att den stött in i väggen skulle den skriva ut namnen på gruppmedlemmarna. Man var även tvungen att välja vilken port som roboten var insatt i, detta var då självklart USB porten.

Det första som man stötte på när man skummade igenom koden var att sätta in sina namn, ska tilläggas att under loppet av laborationen så prövade vi koden ett otalt antal gånger för att testa steg för steg vad ändringarna i koden gjorde.

Ganska snabbt märktes att roboten inte gjorde det man ville den gjorde någon konstig dans och skrev inte ut någonting.

Det var alltså nu som felsökningen började, jag och min partner tog tur att sitta och kolla i koden och den andra satt bredvid med "Introduktion till NXC" samt kollade över axeln för att korrigera fel om denne såg några. Den taktik vi tillämpade var att testa varje steg vi gjorde och se om vi fick några fel och sinns emellan försöka klura ut vad detta var, de flesta rader var ganska enkla och under dessa så gjorde jag mitt bästa för att förklara vad som var fel och hur man lättast kunde korrigera dessa till min partner. Lite mer ingående på exakt vad som ändrades i koden och varför finns under resultat.

Detta gjorde att vi ofta ändrade på en rad och i vissa fall bara en bokstav eller siffra skickade den nya koden till roboten via programmet och testade för att se om vi fick någon respons.

Det största misstaget vi gjorde som medförde att det tog mer tid än det kanske borde var att vi trodde att all kod hade en funktion i programmet, detta visade sig då inte vara fallet med den funktion som kallades "Dance", denna fanns endast där för att ställa till det och när vi till slut fick för oss att "kommentera bort den" så fungera programmet prima!

3. Resultat

Det resultat som man kan direkt säga att man fick var att se roboten göra det som man faktiskt själv hade skrivit (ändrat till) att den skulle göra vilket var väldigt tillfredställande. På långsikt kan man säga att man fick en större förståelse till varför man ska definiera saker och även hur olika funktioner fungerar tillsammans och gör det lättare att felsöka om allting är byggt med detta i åtanke. Här följer även en tabell på de ändringar vi gjorde under laborationen.

Radnummer	Kod	Kommentar
#34	Int blev string IN1 blev IN2 Tillägg av våra namn	Eftersom våra namn inte är en sifferkod så var vi tvungna att byta ut integer mot string IN1 är touchsensor IN2 är displayen.
#43	2 blev 1 + blev -	Dessa ändringar gjordes så att våra namn skulle komma på rätt plats i displayen
#73	IN1 byttes till IN3	Här ville man ha en ljussensor(IN3) och inte en touchsensor(IN1)
#89	A blev B	Detta byttes för att roboten skulle styras åt rätt håll med sensorns hjälp
#97	B blev A	Samma som ovan

#2	80 blev 50	Sänka hastigheten för att se till att roboten kan svänga ordentligt utan att åka av vägen
#3	100 blev 70	Samma som ovan
#112	//Dance()	Robotar ska INTE dansa (inte den här iaf)

4. Analys

Vad mig förskaffar så gick laborationen väldigt bra, som jag skrev innan så var det största problemet vi hade våra egna förutfattade meningar att allting hade en plats i programmet och detta ledde till små frustrationer. Annars så va det mest parprogrammering som gjorde att de gick väldigt bra, en kunde försöka hitta problemet med hjälp av den info vi hade och den andre kunde testa detta väldigt snabbt och effektivt.

Som jag såg på det var det två funktioner som främst möjliggjorde robotens framfärd, dessa var "followLine" samt "readTouchSensors", dessa för att jag skulle säga att dem gör programmet. Man gör en massa olika funktioner men dessa två "tasks" är vad som gör att roboten fungerar så som det är planerat. Det är även dessa två som körs igång i "main-aktiviteten", detta möjliggörs eftersom du kan använda dem definitioner som gjorts i början och kan hämta ifrån de funktioner som skrivits lite här och var i programmet.

Man kan även säga att task main() har en stor del i det hela då den sätter igång alla sensorer, men det är de två andra funktionerna som är själva hjärtat i roboten.

5. Diskussion

Efter laborationen och nu efter att ha skrivit hela rapporten så inser jag att det finns mycket jag kan jobba på när det gäller att arbeta effektivt., och kommer se över hur jag i framtiden tar itu med ett problem. Detta eftersom när problemet att roboten bara dansade när den gick igång uppstod, så kunde jag inte se förbi att det var menat att hända. Personligen tyckte jag programvaran och det syntax som användes var lite krångligt i början men det gick snabbt över till att det var ganska simpelt och lättförståeligt när man fick en helhetsbild på koden. Det ska också tilläggas att den metod som vi tillämpade mest under laborationen var trial and error, detta menas med att vi skrev en bit kod och testa hur roboten reagera.

6. Referenser

Rapportmall.pdf har använts som en mall. // finns på Labb-PM på bilda.

7. Bilaga

The screenshot shows a web browser window with the URL <https://www.kth.se/social/home/private/>. The page is titled "KTH / SOCIAL / DAGBOK" and "Dagbok". It contains a search bar, a "Sök" button, and sections for "Privata sidor", "Portfolio", and "Taggar". A post from August 2013 is visible, titled "Egen anteckning | 29 augusti kl. 14:26" with the content "Chacha bloggen! Idag fick jag en robot att göra en halvmåne :D". The footer contains navigation links for KTH, Organisation, Tjänster, and Kontakt.