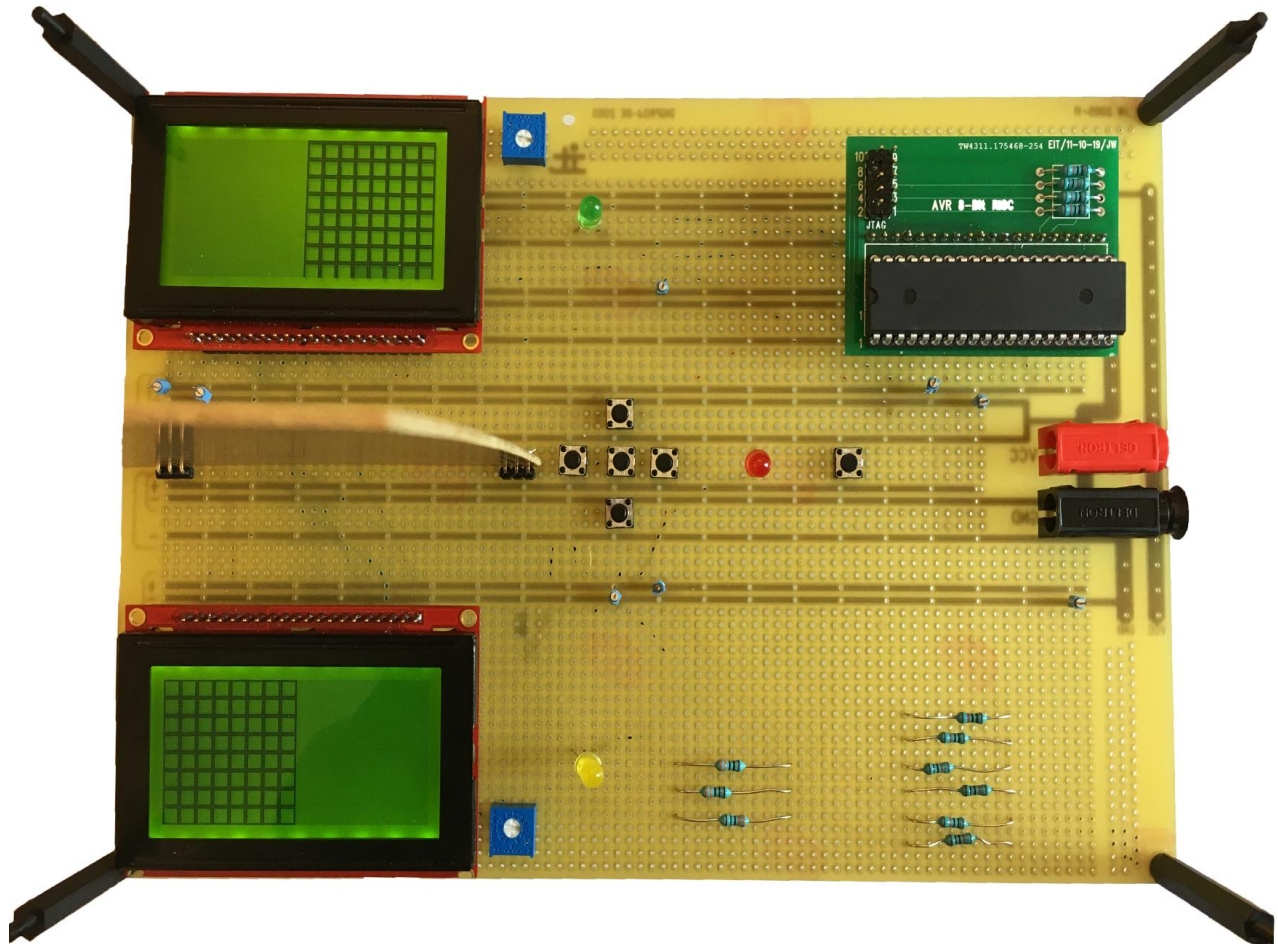


Battleships



Lunds tekniska högskola
EITF11 Digitala projekt

Gustaf Backman, Anton Nystedt & Nicholas Peebo

1. Inledning

1.1 Bakgrund

Digitala projekt (EITF11) är en konstruktionskurs där studenter delas in i grupper och får i uppdrag att tillsammans skapa en konstruktion av låg eller medelhög komplexitet. Med hjälp av en handledare och relevant dokumentation för elektronikkomponenter ansvarar varje grupp för att planera, bygga, programmera och testa sin konstruktion för att slutligen kunna redovisa en fungerande prototyp.

Följande rapport ämnar att i närmare detalj beskriva projektet, arbetsprocessen samt ge läsaren en förståelse för konstruktionen och dess funktioner. Konstruktionen som valdes att tas fram är en prototyp av spelet "Sänka Skepp".

1.2 Syfte

Syftet med projektet var att ge gruppen en djupare förståelse för industriellt utvecklingsarbete samt ge en introduktion till elektronikarbete. Detta innebar bland annat:

- att effektivt tillgodogöra sig information genom läsning av datablad och informationssökning online
- att skapa ett kopplingsschema för konstruktionen i Eagle/Autodesk
- att ta fram en konstruktion bestående av ett antal elektronikkomponenter
- att felsöka och testa konstruktionen fysiskt samt kodmässigt i Atmel Studios 7

2. Kravspecifikation

Kravspecifikationen var projektgruppens ledstjärna genom hela arbetsprocessen. Att konstruktionen skulle uppfylla ett visst antal krav var hjälpsamt både i att styra arbetet framåt samt att sätta upp gränser för projektets komplexitet. Preliminära krav utformades i början av projektet och justerades sedan allteftersom det upptäcktes svårigheter eller möjligheter att förbättra produkten ytterligare. De slutgiltiga kraven är angivna nedan.

2.1 Funktionella krav

Funktionella krav beskriver vilka krav vi ställer på konstruktionens funktionalitet. Dessa delades i detta projekt in i kategorierna nedan:

2.1.1 Startkrav:

- **2.1.1.1.** När spelet startas ska vänster sida av båda skärmar visa ett spelält med rutnät av storlek 8x8.
- **2.1.1.2.** När spelet startas ska höger sida av båda skärmar visa ett spelält utan rutnät av storlek 8x8.
- **2.1.1.3.** Vid spelstart ska spelaren med grön LED-lampa få första turen att placera ut skepp.
- **2.1.1.4.** Vems tur det är att spela ska indikeras av vilken LED-lampa som sitter bredvid skärmen som lyser.
- **2.1.1.5.** När en spelare trycker på reset-knappen återställs spelet från början.

2.1.2 Förberedelsekrav:

- **2.1.2.1.** Spelaren vars tur det är ska vid skeppplaceringen kunna orientera sig på spelbrädet med hjälp av vänster-, höger-, ner- och uppknappen.
- **2.1.2.2.** En spelare ska inte kunna röra sig utanför spelbrädet gränser.
- **2.1.2.3.** Båda spelare ska placera ut ett vardera av följande skepp: hangarfartyg (storlek 5x1), slagskepp (storlek 4x1), kryssare (storlek 3x1), torpedbåt (storlek 2x1), och slutligen u-båt (storlek 1x1).
- **2.1.2.4.** Skeppen ska placeras ut i ordningen: hangarfartyg, slagskepp, kryssare, torpedbåt ,och slutligen u-båt.
- **2.1.2.5.** Skeppen ska placeras ut i lodrätt och vågrätt läge om vartannat.
- **2.1.2.6.** När skeppets form blinkar på spelbrädet ska det indikera vilket skepp som ska placeras härnäst.
- **2.1.2.7.** När en spelare har valt plats för ett skepp ska den trycka på mittenknappen för att placera det.
- **2.1.2.7.** När en spelare har valt plats för ett skepp visas det genom att det placerades skeppets form avtecknas på spelbrädet.
- **2.1.2.8.** Ett skepp ska inte kunna angränsa till ett annat skepp varken lodrätt, vågrätt eller diagonalt.
- **2.1.2.10.** En spelare ska inte kunna sätta ut ett skepp ovanpå en del av ett annat skepp.

- **2.1.2.11.** När den första spelaren har satt ut alla sina skepp övergår turen att sätta ut skepp till den andra spelaren.

2.1.3 Spelkrav

- **2.1.3.1.** När skeppplaceringen är klar ska spelet sätta igång och spelarnas tur slumpas ut.
- **2.1.3.2.** Varje spelare ska kunna se sina egna utsatta skepp på den vänstra sidan av skärmen.
- **2.1.3.3.** Spelaren vars tur det är ska på den högra sidan av skärmen kunna navigera med vänster-, höger-, ner- och uppknapparna.
- **2.1.3.4.** När en spelare trycker på mittenknappen skjuter spelaren i den ruta spelaren står i.
- **2.1.3.5.** Vid träff ska den röda LED-lampan blinka 3 gånger och spelarens fortsätter sin tur..
- **2.1.3.6.** Vid träff ska den ruta som sköts i markeras med ett kryss på ens egna skärm.
- **2.1.3.7.** Vid träff ska den ruta som sköts i markeras med ett kryss på den andra spelarens skärm.
- **2.1.3.8.** Vid miss ska den ruta som sköts i markeras med en kvadrat på ens egna skärm.
- **2.1.3.9.** Vid miss ska den ruta som sköts i markeras med ett streck på den andra spelarens skärm.
- **2.1.3.10.** Vid miss går turen över till nästa spelare.

2.1.4 Slutkrav

- **2.1.4.1.** När en spelare har sänkt samtliga av motståndarens skepp är spelet över.
- **2.1.4.2.** När spelet är över blinkar samtliga lampor 10 gånger.

2.2 Kvalitetskrav

Kvalitetskrav beskriver krav på konstruktionens robusthet, fördröjning, lätthanterlighet etc.

- **2.2.1.** Spelet ska ta mindre än 10 sekunder att starta upp
- **2.2.2.** Spelknapparna får som högst ha en fördröjning på 0.5 sekunder
- **2.2.3.** Spelet får som högst krascha en gång var femte spel.

3. Hårdvara

Följande avsnitt anger de hårdvarukomponenter som användes i projektet med en kort beskrivning av varje del. Se även kopplingsschemat (bilaga 8.1) för komponenternas samspel.

3.1 Processor

Processorn som användes var en 8-bitars ATmega16 AVR microcontroller. Processorn har totalt 40 stycken pinnar varav 32 av dem är I/O-pinnar fördelade i grupper om 8 mellan PORT A, B, C och D. Processorn har även en 8- och 16-bitars timer som kan används för att generera avbrott.

3.2 JTAG (Joint Test Action Group)

En JTAG (vars namn kommer från Joint Test Action Group som först kodade den) används för att föra över kodinstruktioner till processorn. Fyra pinnar på processorns C-port reserverades för JTAGen.

3.3 Display

För projektet användes två stycken grafiska LCD-skärmar av modell GDM12864H med dimensioner 128x64 pixlar. LCD-skärmen har totalt 20 pinnar uppdelade i data-, kontroll- och ström-pinnar. På grund av användningen av två stycken LCD-skärmar och ett begränsat antal pinnar på processorn delar skärmarnas databus på en port på processorn. Skärmarnas olika instruktioner hålls isär genom att aktivera Chip Select på skärmen som avses ta emot data och inaktivera den på den andra. På detta sätt kan man särskilja de olika skärmarnas instruktioner.

3.4 LED-lampor

För projektet användes en gul, en röd och en grön LED-lampa som kopplades till lämpliga motstånd om 330 ohm.

3.5 Knappar

Knapparna som används var strömbrytare kopplade till jord samt till VCC genom en resistor på 10k Ω . En icke-intryckt knapp visar hög spänning, medan en intryckt knapp visar låg spänning.

3.6 Varistor

En varistor är ett variabelt motstånd som gör det möjligt att reglera motståndet. Varistorn användes för att justera LCD-skärmens ljusstyrka och därigenom kontrast.

3.7 Resistorer

Motstånden som användes för projektet var på $10k\Omega$ samt 330Ω .

4 Mjukvara

All kod för programmet skrevs i programspråket C i programmet Atmel Studios 7. Atmel Studios kompilerar sedan koden och översätter den till maskinkod som genom JTAGen överförs till minnet i processorn.

Utöver Atmel Studios användes Eagle/Autodesk i ritandet av ett kopplingsschema. Programmet gjorde det enkelt att skapa ett tydligt och enkelt kopplingsschema, bland annat genom att erbjuda möjligheten att importera relevanta komponentbibliotek.

5 Genomförande

5.1 Planering

Förberedelse och planering av projektet gjordes under kursens första läsperiod. Projektgruppen valde att målet med projektet skulle vara att utveckla en konstruktion att kunna spela "sänka skepp" på med två separata skärmar för de olika spelarna. Därefter gjordes efterforskning i vad som skulle krävas för ett sådant projekt: vilken sorts hårdvara som skulle användas, hur reglerna skulle utformas, vilka sorts funktioner spelet skulle behöva etc. Planeringsarbetet delades upp bland gruppmedlemmarna och mynnade till slut ut i en preliminär kravspecifikation, en arbetsplan samt ett levande arbetsdokument.

Efter att detta tagits fram började processen att skapa ett preliminärt kopplingsschema. Efter noggrann läsning av databladerna för de olika komponenterna samt en tillväjningsperiod i programmet Eagle lyckades gruppen ta fram ett kopplingsschema att använda som guide för konstruktionen.

5.2 Konstruktion

Efter att ha blivit klara med kopplingsschemat och blivit tilldelade nödvändiga komponenter påbörjades konstruktionsprocessen. Komponenterna löddes fast vid kopplingsplattan på ett funktionellt och estetiskt tilltalande sätt. Färgkodade kopplingstrådar virades sedan mellan de olika komponenterna enligt kopplingsschemat.

5.3 Programmering

Efter att konstruktionen var färdigställd började arbetet med programmeringen där makros definierades för att förenkla kodandet och se till att rätt instruktion skrivs till rätt pinne på processorn. Trevande inledande försök att få LED-lampor att lysa vid en knapptryckningen gjordes för att se om konstruktionen var korrekt kopplad. Arbetet fortsatte därefter med att förstå skärmarnas funktioner och att skriva ut enkla instruktioner på dem.

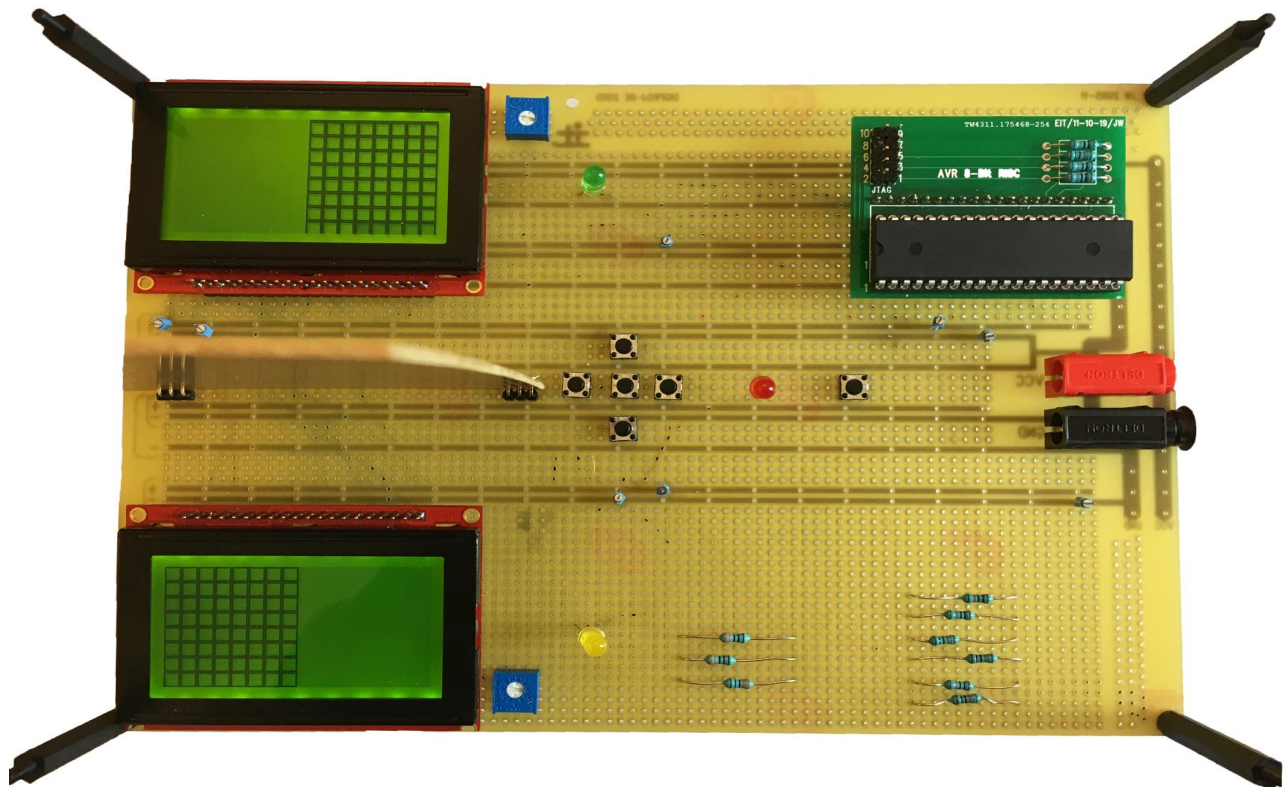
Efter ett antal lyckade försök att rita ut pixlar på skärmarna fördelades resterande programmering mellan de olika gruppmedlemmarna. Koden delades upp i fyra huvuddelar: att hantera knapptryckningar, att rita olika mönster på LCD-skärmarna, att placera ut skepp på skärmen samt att definiera spelregler och avgränsningar.

5.4 Testning

Efter att programmering var avklarad gjordes tester för att felsöka programmet. Kravspecifikationen användes här som guide då den beskriver programmets funktioner och krav på den färdigställda produkten. Vid upptäckt av fel gjordes ändringar i programmet och testningen påbörjades på nytt.

6 Resultat

Slutprodukten av projektet är en konstruktion med funktionalitet för att låta två spelare spela sänka skepp mot varandra. Spelarna börjar med att i tur och ordning placera ut sina skepp på sina spelplaner. Därefter börjar den ena spelaren med att skjuta på någon av motståndarens rutor. Vid miss markeras en miss ut på spelplanen och turen går vidare till den andre spelaren, vid träff markeras en träff ut på spelplanen och spelaren får fortsätta skjuta. Spelet är över när den ena spelaren har lyckats skjuta på samtliga av rutor som motståndarens skepp befinner sig på. En illustration av konstruktionen vid uppstart visas i Figur 1 nedan.



Figur 1. Konstruktion vid uppstart

7 Diskussion

Arbetet med detta projekt har fört med sig många lärdomar och insikter. Det har framförallt varit intressant att få en inblick i konstruktionsprocessen och få se fysiska resultat av kod man har skrivit. Arbetet har dessutom resulterat i att samtliga medlemmar i projektgruppen har lärt sig grunderna i att konstruera en enklare elektronikmodell och se kopplingen mellan hårdvara och mjukvara på ett tydligare sätt.

På mjukvarusidan har det även gett insikt i hur Atmel Studio 7 fungerar, hur kod översätts till assembler- och maskinkod och gett oss en introduktion i programspråket C. På hårdvarusidan har samtliga medlemmar fått grundläggande kunskap i att löda, vira och använda olika elektronikverktyg vilket har lett till en ökad förståelse av elektronik i sin helhet.

8 Bilagor

8.1 Kopplingschema

Kopplingschemat nedan visar hur komponenter är kopplade till varandra. Att anmärka är att pinnarnas beskrivning på LCD-skärmarna (CS, D1, D2 etc.) ej stämmer överens med skärmen som användes i verkligheten pga. variationer i ritningar av LCD-skärmen. För rätt avläsning använd databladet för GDM12864H och jämför med pinnarnas siffror.

