

Snake

Digitala Projekt (EITF11)

Fredrik Jansson, I-12

Oskar Petersen, I-12

Lunds Tekniska Högskola, 2015-05-18

Handledare: Bertil Lindvall

Abstract

Denna rapport beskriver ett projekt där ett klassiskt Snake-spel konstruerats. Projektet omfattar såväl lödning och sammankoppling av valda komponenter som programmering av själva applikationen i C.

Innehållsförteckning

INLEDNING	3
PRODUKT	3
PRODUKTBESKRIVNING.....	3
KRAVSPECIFIKATION.....	3
MJUKVARA.....	3
<i>Källkod</i>	3
HÅRDVARA	4
<i>Display</i>	4
<i>Lampdiod</i>	4
<i>Processor</i>	4
<i>J-Tag</i>	4
<i>Knappar</i>	4
KOPPLINGSSCHEMA.....	4
METOD	4
ETAPP 1.....	4
ETAPP 2.....	4
ETAPP 3.....	4
ETAPP 4.....	5
ETAPP 5.....	5
ETAPP 6.....	5
RESULTAT	5
DISKUSSION	5
BILAGA 1 – KOPPLINGSSCHEMA	6
BILAGA 2 – SNAKE IN ACTION	6

Inledning

Kursen digitala projekt låter studenten arbeta med hårdvaruprogrammering i projektnivå, där hela processen från koppling och lödning, till programmering och slutttestning integreras i projektet. Projektet börjar med att studenter i grupper väljer ut en lämplig produkt för att sedan arbeta fram denna under kommande sex veckor. Kursen låter studenten få en uppfattning om hur ett projekt av denna typ generellt kan se ut, vilket blir en viktig pusselbit i dennes annars väldigt teoretiska utbildning.

Produkt

Produktbeskrivning

Vi har valt att konstruera ett Snake spel. Spelet går ut på att man styr en orm över en spelplan utan att krocka med sig själv eller planens kanter. Spelplanen innehåller alltid ett äpple som placerats ut på en slumpmässig plats. Åker man över äpplet äter ormen upp det och blir en enhet längre, och ett nytt äpple placeras ut. Varje äpple ger ett poäng och den totala poäng man uppnått då man krockar blir ens slutliga resultat.

Kravspecifikation

- Spelet ska presenteras på en LCD-skärm
- Tre knappar ska finnas tillgängliga:
 - Svänga vänster
 - Svänga höger
 - Knapp för play/pause
- När spelet startas genereras en orm som är tre pixlar lång och är placerad i mitten av spelplanen.
- Ormen rör sig konstant framåt tills spelaren byter riktning med 90 grader åt vänster eller höger med hjälp av knapparna.
- När spelet startas innehåller spelplanen ett äpple. När äpplet plockas upp av ormen förlängs ormen med en pixel och ett nytt äpple genereras slumpmässigt någonstans på spelplanen.
- Spelaren får en poäng för varje äpple som plockats upp.
- Spelet fortgår tills man "krockar" (då spelaren åker in i sig själv eller i spelplanens väggar). Då avbryts spelet och spelarens slutgiltiga poäng visas på den högra skärmhalvan.
- För att starta ett nytt spel trycker man på play/pause-knappen. När som helst under spelets gång ska man kunna trycka på knappen igen för att pausa. Ett ytterligare knapptryck sätter igång spelet igen.
- En grön LED-lampa blinkar varje gång ormen äter upp ett äpple.

Mjukvara

För att rita kopplingsschema användes PowerLogic.

Källkod

Programkoden skrevs i C med hjälp av programmet AVR Studio.

Hårdvara

Display

Displayen som använts till spelet är en LCD-skärm av typen GDM12864C. Skärmen är uppdelad i två halvor med 64x64 pixlar vardera, det vill säga sammanlagt 128x64 pixlar för hela skärmen.

Lampdiod

Varje gång ormen äter upp ett äpple och därmed ökar sin poäng blinkar en grön LED-lampa.

Processor

Processorn som använts till spelet är av typen ATmega16. Processorn har 40 pinnar att sammankoppla med övriga komponenter (dock är åtta stycken av dem reserverade till vår J-Tag vilket lämnar kvar 32 stycken). Hur dessa kopplingar är gjorda syns i kopplingsschemat.

J-Tag

För att få processorn att kopplas samman och kommunicera med en dator användes en J-Tag.

Knappar

Till spelet används tre knappar. Processorn får signal då en knapp trycks ned. Då en knapp inte är nedtryckt är spänningen till processorn 0 V genom att knappen på andra sidan av processorn är kopplad till en resistor (se kopplingsschema).

Kopplingsschema

Ovanstående hårdvara är sammankopplade enligt det kopplingsschema som återfinns under bilaga 1.

Metod

Etapp 1

Processen började med att en produkt skulle beskrivas. Detta gjordes genom att förslag på produkt lämnades in till handledaren, och till denna en kravspecifikation för att beskriva produkten och problemet. Inte förrän handledaren godkänt produktvalet och kravspecifikationen kunde arbetet fortsätta till nästa steg.

Etapp 2

Nästa steg var att rita ett kopplingsschema över produkten och dess komponenter. För att se till att inga kortslutningar skulle komma att uppstå eller andra misstag göras var det tvunget att lämna in ett fullständigt kopplingsschema och få detta godkänt innan arbetet kunde fortskrida. Efter dialog med handledaren valdes det att revidera den ursprungliga kravspecifikationen i detta skede, genom att en funktionerna från en lampdiod lades till.

Etapp 3

Med ett godkänt kopplingsschema kunde en verktygslåda hämtas ut från handledaren, innehållande de verktyg som skulle komma att behövas under konstruktionen samt de

hårdvarukomponenter som konstruktionen behövde. Nu kunde arbetet med att koppla samman komponenterna börjas.

Etapp 4

Med en färdig konstruktion av hårdvarukomponenterna kunde programmeringen påbörjas. Med J-Tag kopplades modellen ihop med en dator och grundläggande test av kopplingarna kunde göras. Eventuella fel rättades till och processen kunde fortsätta till programmeringsetappen

Etapp 5

Under denna etapp skulle programmet skrivas. Små steg togs i början för att lära sig hur datorn kommunicerade med processorn, och processorn med LCD-skärmen. Sedan fortsattes programmeringen och färdigställdes utefter den nu reviderade kravspecifikation som skrivits till spelet.

Etapp 6

Efter sluttetning av programmeringen konstateras att spelet uppfyller de krav som ställts, och projektet avslutas med rapportskrivning.

Resultat

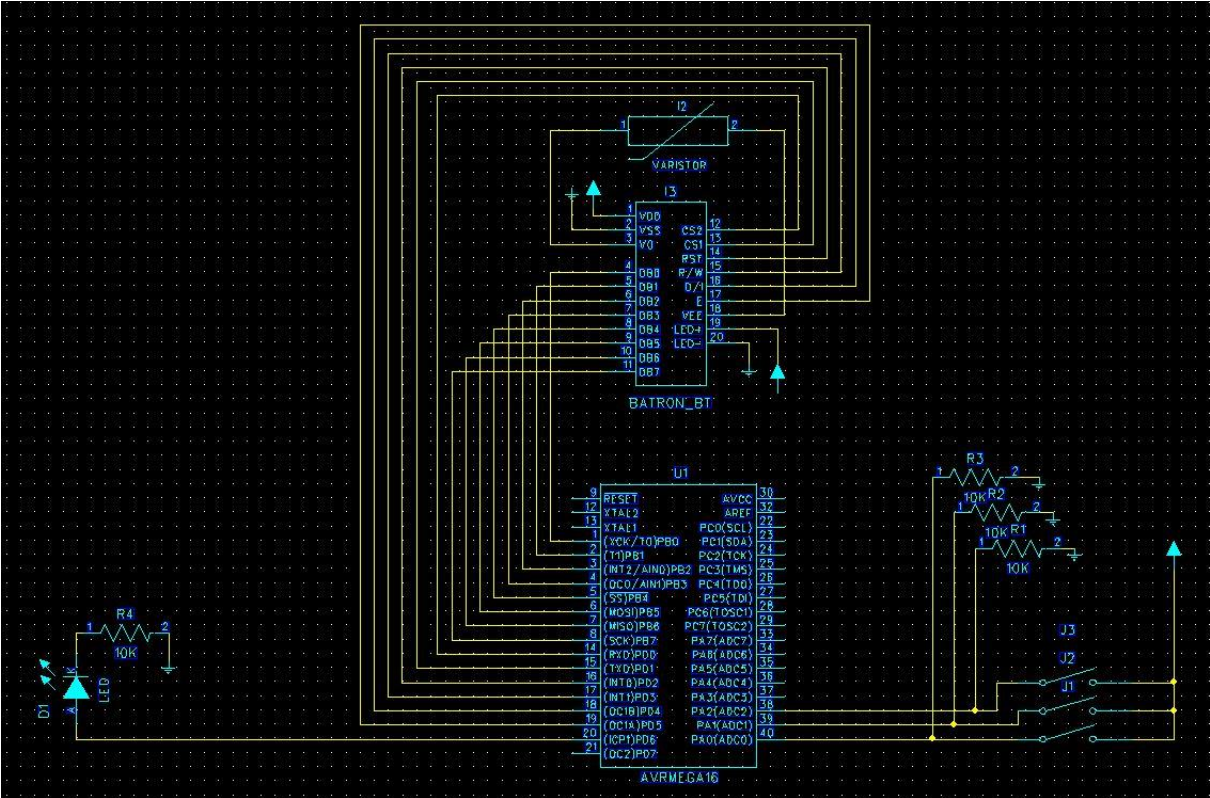
En screenshot av vårt spel syns under bilaga 2. Vårt spel uppfyller den kravspecifikation som vi skapade vid projektets början.

Diskussion

Det vi inte hade räknat med innan projektets början var att det var en tidskrävande process att sätta sig in i hur hårdvaran, framförallt LCD-skärmen, fungerade. Efter att ha ägnat mycket tid åt att prova sig fram tillsammans med läsande av produkternas datablad förstod vi hur vår kod behövdes utformas för att fungera tillsammans med hårdvaran. Vi stötte under processens gång på problem med vår hårdvara, som helt enkelt inte fungerade som den skulle. Efter successiva felsökningar var vi tvungna att byta ut vår JTAG, vår LCD-skärm och vår processor, vilket ställde ännu större tidspress på vårt projekt. Genom att vi lyckades felsöka oss till att vi var drabbade av hårdvaruproblem lyckades vi med nöd och näppe bli klara med projektet i tid.

Vi känner att projektet har givit oss en genomgående förståelse för hur hårdvara och mjukvara fungerar tillsammans, ända ned till binär nivå, med ettor och nollor. Det har varit väldigt intressant att få en inblick i hur ett projekt av denna typ kan se ut på en småskalig nivå, vilket ger en uppfattning om vilken omfattning samma process har på en större nivå.

Bilaga 1 – Kopplingschema



Bilaga 2 – Snake in action

