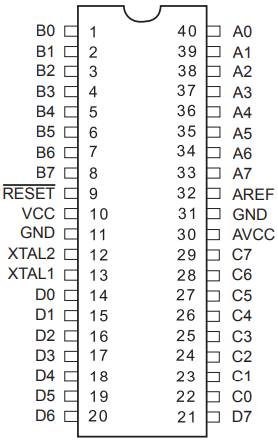
AtMega16A

Två AtMega16A processorer användes till projektet. Processor 1 stod för utskrivning till en 16x8 ledmatris, hantering av indata från knapar samt spellogik. Processor 2 var kopplad till en högtalare som processorn skickade signaler till och ljud kunde därefter komma ut ur högtalaren. AtMega16A har tillgång till 32 register som är 8 bitar långa.

AtMega16A har 4 portar: A, B, C och D. Varje port innehåller 8 pinnar som är numrerade 0 – 7 och kan användas som in- eller utgångar av logiska signaler som bestäms via initialisering Det finns även pinnar på processorn som inte tillhör någon port, de pinnarna är 9 – 13 samt 30 – 32. Av de som används så är de kopplade från:

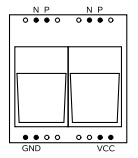
* Pin 10 och pin 30 till jord (GND)
* Pin 11 och pin 31 till ström (VCC)

För de övriga pinnarna kan de användas till processorns interna funktioner. Pinnarna för portA kan ta emot analoga signaler till den inbyggda A/D-omvandlaren. PortD kan hantera avbrottssignaler. I detta projekt används de interna funktionerna däremot inte. För processor 1 hanterar förövrigt portA indata från tryckknappsmodulen som sköter horisontell förflyttning, portB hanterar kommunikation med display, portC hanterar JTAG och SPI och portD hanterar kommunikation till Processor 2 samt indata från tryckknappsmodul som sköter rotation. För processor 2 hanterar portA indata från processor 1 och portC hanterar JTAG samt skrivning till högtalaren.



Källa: AtMega16A databladet

Bildkälla: AtMega16A databladet fast ändrat på den :O

Dubbelknappar

Det är de avstudsade tryckknappsmodulerna som används för att spela det egengjorda Tetriset. Med två 2x1 tryckknappsmoduler med inbyggt motstånd finns det totalt 4 knappar att använda varav två sköter den vertikala flyttningen och en av dem sköter rotationen. En av knapparna är inte inkopplad och inte heller tänkt att användas. (Orsaken till att vi har den knappen över är på grund av dålig resurs av enkelknappar). För konstruktionen av kretskortet kopplades tre pinnar från processor 1 till knapparnas positiva utgångar markerade P på bild X. Dessa ger en positiv utsignal vid nedtryckning av knapp.

* Inte ha kopplingsschema över det interna i knappen för varför?
* Antingen sån bild eller på riktigt

Källa: Vanheden liu

Bildkällaälla: Vanheden liu

Resetknapp

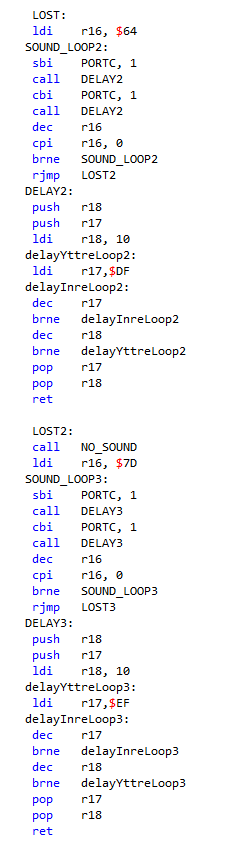
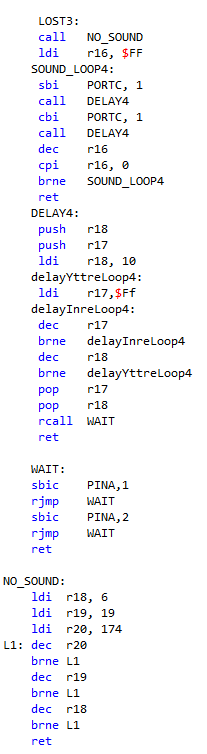
Resetknappen monterades för att under arbetets gång kunna felsöka bra, men även som en spelfunktion för att ge användaren en trevlig spelupplevelse. Enkelknappen installerades så att en positiv flank avges när knappen är intryckt enligt bild X och processor 1 programmerades utefter detta. Knappen som användes har inget inbyggt motstånd och kopplades därav till ett XXX (hur mycket resistans?) motstånd för att inte kortsluta hela kretskortet.

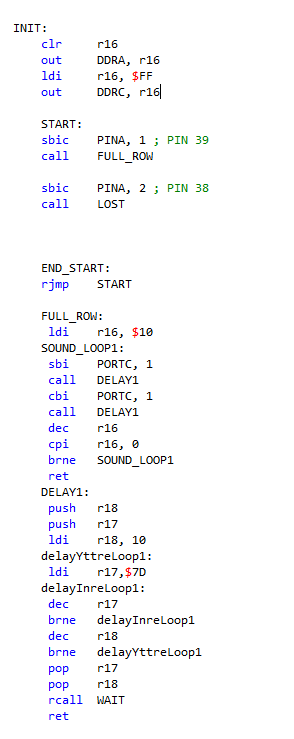
* Bild på enkelknappen?
* Bild på kopplingsgrej i knapp, Jesper hittade några bra!

Högtalare

Högtalaren används till spelet för att generera ett retro ljud om en rad försvinner och tre på varandra följande ljudsignaler om man dör. Högtalaren är kopplad till jord (GND), men även till pinne 23 på processor 2. I processor 2 sköts logiken för vilket ljud som ska spelas enligt kod på bild X. Där skickas även x antal signaler ut till högtalaren som omarbetas till ljudsignaler vi kan höra.

* Bild på högtalare?





Tillvägagångssätt och planering

I början av projektet var tempot lugnt för att ta sin tid att läsa datablad och skaffa sig en riklig grund. Processorerna började därefter kopplas på kretskortet och möjligheten till SPI skapades genom att koppla ihop rätt pinnar med varandra. När det blev dags för kodning ökade tempot. En prick displayen började snart röra sig vertikalt neråt med hjälp av en avbrottsfunktion. Knappar kopplades in för att förflytta pricken horisontellt och med lite spellogik hade ett väl fungerande Tetris skapats. Ett Tetris med bara en prick det vill säga.

Det gick därefter till att skapa figurer och hitta ett sätt att rotera dessa. Under tiden detta gjordes arbetades det även på högtalaren för att skapa ljud när man rensade en hel rad eller förlorade. Efter två veckor hade spelet tagit sin form och efter lite slutlig buggtestning ansågs det klart och redo för redovisning.

Planeringschema

I tabell X nedan visas en sammanfattning av planeringsschemat över hur veckorna såg ut i form av vilket arbete som utfördes.

|  |  |
| --- | --- |
| Vecka | Arbetsuppgift |
| v.5 | Krav specialisering inlämnad (Fredag) |
| v.6 | Läsa datablad och börja koppla |
| v.7 | Gravitation och Tetris med en prick |
| v.8 | Figurer |
| v.9 | Rotation och högtalare |
| v.10 | Rotation och buggar |
| v.11 | Skrivning av redovisning och buggar |
| v.12 | Redovisning (Måndag) |