



UPPSALA
UNIVERSITET

Signaler och system

www.signal.uu.se

Lars Ericsson
2002-03-18

Denna laboration är avsedd att ge en första bekantskap med en enkapseldator ur Atmels AVR-familj, utvecklingskortet STK200, C-kompilatorn CodeVision, samt simulatorn AVR Studio.

Slutprodukten från nedanstående uppgifter skall bli ett "stoppur" som visar tiden på de åtta lysdioderna på utvecklingskortet. En felvisning på upp till 1s/min är acceptabel. Varje deluppgift tillför en pusselbit och resultatet blir förhoppningsvis ett fungerade stoppur.

Det färdiga stoppuret redovisas genom demonstration för handledare; någon rapport behöver ej lämnas in.

Utvecklingskort och C-kompilator.

Laborationen går att utföra på de flesta kretsar i AVR-familjen, allt som behövs är nio I/O-pinnar. Förmodligen sitter det en 40-pinnars 90S8515 eller 90S8535 på utvecklingskortet; båda fungerar utmärkt för denna applikation. Lysdioderna på kortet är anslutna, via kortslutningsbyglar, till PORT B på AVR-chipet. Strömbrytarna är anslutna på samma sätt till PORT D. Kontrollera att samtliga byglar sitter på plats. Slå på matningsspänningen till utvecklingskortet och kontrollera att det är anslutet till PC:n.

För programutveckling används lämpligtvis C-kompilatorn CodeVision. Programmet erbjuder en integrerad grafisk miljö och följer det numera vanliga konceptet med sammanhållna "projekt" för varje applikation. Börja därför med att välja *New - Project*. Då ett nytt projekt skapas finns möjligheten att låta CodeVision automatiskt generera ett programskelett med färdig kod för initiering av de olika enheterna i AVR-chipet. Välj gärna att använda detta, det kan spara en hel del arbete. Efter inmatning av namn på projektfiler och källkodsfil är det bara att sätta igång med kodningen. Hjälptexter för kompilatorn med tillhörande biblioteksfunktioner samt för funktionerna i det grafiska gränssnittet finns som vanligt tillgängliga under *Help*. För att undvika onödigt kodningsarbete är lämpligt att skumma igenom förteckningen över tillgängliga biblioteksfunktioner innan kodningen påbörjas. En omfattande tutorial för CodeVision, framtagen av Cornell University, finns också tillgänglig på kursens hemsida.



UPPSALA
UNIVERSITET

1. Sekundvisning.

Implementera en "sekundvisare" som visar aktuellt antal sekunder räknat från spänningspåslag. Visningen skall ske på binär form med hjälp av lysdioderna PB.6-PB.0. Efter 1 minut skall sekundvisaren börja om på 0, som brukligt är.

- a) Innan en applikation körs på den verkliga hårdvaran är det lämpligt att verifiera funktionen med hjälp av en simulator. Börja därför med att skriva en programversion med väldigt "korta" sekunder, vilken kan provköras i simulatören AVR Studio. Denna kan startas direkt från *Tools - Debugger* i CodeVision, varefter den kompilerade objekt-kodsfilen kan laddas in via *File - Open* på vanligt sätt. Programmet under utveckling kan nu exekveras instruktion för instruktion och förändringar i variabelinnehåll, portar, e.dyl. kan följas. Välj *View - New I/O View* och öppna PORT B. Kör sedan programmet med t.ex. *Debug - Auto Step* och kontrollera att PORT B förändras som avsett. Beskrivning av simulatorns övriga möjligheter finns under *Help*.
- b) Då programmet fungerar som avsett i simulatören är det dags för provkörning på utvecklingskortet. Kompilera om programmet med normal längd på sekunderna och ladda ned den körbara hexkodsfilen till flashminnet på AVR-chipet. Detta kan antingen göras manuellt via *Tools - Chip Programmer* eller automatiskt genom att markera denna option under *Project - Configure*. Efter nedladdning sker en automatisk reset av AVR:en och programmet exekveras. Studera lysdioderna under minst en minut och verifiera att tiden visas på korrekt sätt och att noggrannheten uppfyller kraven.

2. Minutvisning

Kompletera programkoden så att antal minuter (upp till 3) visas på de återstående lysdioderna. Provkör på samma sätt som i uppgift 1.

3. Startknapp

Kompletera programkoden så att tidtagningen inte startas direkt vid spänningspåslag utan först då användaren trycker på knappen PD.0. Verifiera funktionen såväl i simulatören som på utvecklingskortet.

4. Start / Stopp / Nollställ

Kompletera slutligen programkoden så att samma knapp kan användas för start, stopp och nollställning av stoppuret.