

Mikroprosessorsystemer

Labøving 1 Porter og forsinkelser.

I denne øvingen skal vi implementere setting, sletting og veksling (toggling) av utgangspinner i C. Vi skal bruke ATmega328P i Arduino UNO.

Atmel Studio

Vi bruker det integrerte utviklingsmiljøet (IDE) for 8-bits AVR kretser som heter Atmel Studio 7. Det er gratis og kan også lastes ned på deres egen pc fra www.atmel.com/avr. Det er kompilerer assembly og c-kode, simulerer og kan programmere kretser med resulterende kode.

Installer Atmel Studio og Arduino (hvis du ikke allerede har det) og konfigurer Arduino som verktøy (tool) som beskrevet her:

<http://www.instructables.com/id/How-to-Load-Programs-to-an-Arduino-UNO-From-Atmel/>

Vi skal så lage et lite prosjekt som skriver til registre for å styre en utgangspinne.

- Start opp Atmel Studio.
- Lag et nytt prosjekt. Velg GCC C Executable Project
- Gi prosjektet et navn, f.eks. laboving1_DittNavn. Pass på at navnet ikke inneholder noen mellomrom, spesialtegn, eller norske bokstaver. Hvis du bruker en skole-PC (og den fungerer) så kan prosjektet ligge på c:\temp\ på den lokale maskinen. UNC nettverksstier fungerer ikke (dvs. //dinkatalog/..), men f.eks. J:\brukernavn\uPsys kan fungere så lenge det ikke er mellomrom eller andre ulumskheter som nevnt ovenfor..
- Trykk "next" og velg ATmega328P som device.
- Trykk "finish".

Oppgave 1

Kopier inn følgende program:

```
/* Defines */
#define F_CPU 16000000UL           // Tell delay.h clock frequency

/* Includes */
#include <avr/io.h>                /* Defines pins, ports, etc */
#include <util/delay.h>            /* Functions to waste time */

int main(void)
{
    // Initialization
    DDRB = 1<<DDB5;               // 0b00100000 - PB5 output - other PB pins inputs
    PORTB = 1<<PORTB5;            // PB5 high - turn on led

    // Main loop
    do {
        // empty
    } while (1);                  // End main loop

    // return (0);                 // Never reached/not needed
}
```

Kompiler og sjekk at programmet er feilfritt. Programmet kan også simuleres for å sjekke om det virker som forventet før det programmeres på Arduino UNO.

Oppgave 2

Utvid programmet så det i stedet for å slå på LEDen (endrer høy/lav). Vent ett sekund mellom hver togging.

Forsinkelse i avrgcc kan implementeres med `_delay_ms()` funksjonen i AVR Libc.

Bit-manipulering er beskrevet i notat på læringsplattformen.

Ekstraoppgave

Varier tiden mellom hvert blink.

Lever oversiktlig og kommentert c-fil(er) via It's learning.