
Proj_1F_PCI_and_T1_interrupt

MULTI PATTERN DISPLAY

Uses switch presses to select any one of six displays. A more ambitious program in which the inevitable complexity is handled by two subroutines (see below) and two interrupt service routines one triggered by a timer and the other by a switch press. One of the subroutines is used to initiate the display and the other to increment it.

IT INTRODUCES

1. The Watch Dog Timer: Part of the Atmega HW that can be programmed to generate a reset if program execution unexpectedly halts for any reason.
2. Interrupts from multiple sources: Combining a timer interrupt to increment the display with a PCI interrupt to select the display and initialise it.
3. Lots of logic; not really of interest here.
4. Subroutines: A subroutine executes a segment of code on behalf of the main routine. Subroutines may be used to make a program more readable especially if the code segment is required several times. Alternatively the subroutine may provide a service like some arithmetic, that may be used by just one or by many programs.

Consider the project subroutine `void Inc_Display(char mode)`.
It is called by the statement `Inc_Display(switch_press);`

`void Inc_Display(char mode)` defines a variable and gives it the name `mode`. The initial value of `mode` is supplied by the statement `Inc_Display(switch_press)` in the main routine in which `switch_press` is defined as a char variable.

The subroutine can change the value of `mode` but these changes will not effect the variable `switch_press` in the main routine. The term `void` indicates that the subroutine will not be returning any value to the main routine as might be the case for an arithmetic function for example.