```
; a2_template.asm
; CSC 230 - Summer 2018
;
; Some starter code for Assignment 2. You do not have
; to use this code if you'd rather start from scratch.
;
; B. Bird - 06/09/2018

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;                          Constants and Definitions                           ;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

; Stack pointer and SREG registers (in data space)
.equ SPH_DS = 0x5E
.equ SPL_DS = 0x5D
.equ SREG_DS = 0x5F

; Initial address (16-bit) for the stack pointer
.equ STACK_INIT = 0x21FF

; Port and data direction register definitions (taken from AVR Studio; note that m2560def.inc does
not give the data space address of PORTB)
.equ DDRB_DS = 0x24
.equ PORTB_DS = 0x25
.equ DDRL_DS = 0x10A
.equ PORTL_DS = 0x10B


; Definitions for button values from the ADC
; Some boards may use the values in option B
; The code below used less than comparisons so option A should work for both
; Option A (v 1.1)
;.equ ADC_BTN_RIGHT = 0x032
;.equ ADC_BTN_UP = 0x0FA
;.equ ADC_BTN_DOWN = 0x1C2
;.equ ADC_BTN_LEFT = 0x28A
;.equ ADC_BTN_SELECT = 0x352
; Option B (v 1.0)
.equ ADC_BTN_RIGHT = 0x032
.equ ADC_BTN_UP = 0x0C3
.equ ADC_BTN_DOWN = 0x17C
.equ ADC_BTN_LEFT = 0x22B
.equ ADC_BTN_SELECT = 0x316


; Definitions of the special register addresses for timer 0 (in data space)
.equ GTCCR_DS = 0x43
.equ OCR0A_DS = 0x47
.equ OCR0B_DS = 0x48
.equ TCCR0A_DS = 0x44
.equ TCCR0B_DS = 0x45
.equ TCNT0_DS  = 0x46
.equ TIFR0_DS  = 0x35
.equ TIMSK0_DS = 0x6E

; Definitions of the special register addresses for timer 1 (in data space)
.equ TCCR1A_DS = 0x80
.equ TCCR1B_DS = 0x81
.equ TCCR1C_DS = 0x82
.equ TCNT1H_DS = 0x85
.equ TCNT1L_DS = 0x84
.equ TIFR1_DS  = 0x36
.equ TIMSK1_DS = 0x6F
```

```
; Definitions of the special register addresses for timer 2 (in data space)
.equ ASSR_DS = 0xB6
.equ OCR2A_DS = 0xB3
.equ OCR2B_DS = 0xB4
.equ TCCR2A_DS = 0xB0
.equ TCCR2B_DS = 0xB1
.equ TCNT2_DS  = 0xB2
.equ TIFR2_DS  = 0x37
.equ TIMSK2_DS = 0x70


.cseg
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;                           Reset/Interrupt Vectors                           ;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
.org 0x0000 ; RESET vector
        jmp main_begin

; Add interrupt handlers for timer interrupts here. See Section 14 (page 101) of the datasheet for
addresses.

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;                                 Main Program                                ;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

; According to the datasheet, the last interrupt vector has address 0x0070, so the first
; "unreserved" location is 0x0074
.org 0x0074
main_begin:

        ; Initialize the stack
        ldi r16, high(STACK_INIT)
        sts SPH_DS, r16
        ldi r16, low(STACK_INIT)
        sts SPL_DS, r16

stop:
        rjmp stop



;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;                                 Data Section                                ;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

.dseg
.org 0x200
; Put variables and data arrays here...
```