```
;*****************************************************************************
;* Title:           read_DHT11.asm
;* Author:          Jason Chen
;* Version:         1
;* Last updated:    11/21/2022
;* Target:          AVR128DB48
;*
;* DESCRIPTION
;*      Design Task 2:
;*          This program sends a start signal by taking the DATA line down
;*          for 18ms and then releasing it for 20us. Then the DHT11 will
;*          take hold of the line and send a response signal before sending
;*          data bits to the MCU.
;*
;* VERSION HISTORY
;* 1.0 Original version
;*****************************************************************************

start:              ; Configure I/O ports
    ldi r16, 0xFF
    out VPORTD_DIR, r16
    ldi r16, 0xF0
    out VPORTA_DIR, r16
    ldi r16, 0x00
    out VPORTC_DIR, r16
    cbi VPORTE_DIR, 0

main_loop:
    rcall send_start
    rcall wait_for_response_signal
    rcall read_DHT11
    rjmp main_loop

read_DHT11:
    in r16, VPORTB_IN
    andi r16, 0x01
    ret

wait_for_0:
    rcall read_DHT11
    cpi r16, 0x00
    brne wait_for_0
    ret

wait_for_1:
    rcall read_DHT11
    cpi r16, 0x01
    brne wait_for_1
    ret

wait_for_response_signal:
    rcall wait_for_0
```

```
53      rcall delay_80us
54      rcall wait_for_1
55      rcall delay_80us
56      ret
57
58  read_DHT11_data_bit:    ; records bit in bit 0 of r16
59      rcall wait_for_0
60      rcall delay_20us
61      rcall wait_for_1
62      rcall delay_30us
63      rcall read_DHT11            ; record logic level of DATA line as input
64      ret
65
66
67
68  ;*********************************************************************
69  ;* "send_start" - Send start signal
70  ;* Author:          Jason Chen
71  ;* Description:
72  ;*      Makes DATA line 0 for 18ms then a 1 for 20us (or 40us).
73  ;*********************************************************************
74
75  send_start:
76      rcall write_0_to_DHT11
77      rcall delay_18ms
78      rcall write_1_to_DHT11
79      rcall delay_20us
80      /*rcall delay_20us*/
81      ret
82
83  write_0_to_DHT11:
84      cbi VPORTB_DIR, 0
85      ret
86
87  write_1_to_DHT11:
88      sbi VPORTB_DIR, 0
89      ret
90
91  ;*********************************************************************
92  ;* "delay_18ms" "delay_50us" "delay_20us" - Delay
93  ;* Author:          Jason Chen
94  ;* Description:
95  ;*      Delay 18ms : 72000 clock cycles accounting for 2 clocks from rcall
96  ;*      Delay 50us : 200 clock cycles accounting for 2 clocks from rcall
97  ;*      Delay 20us : 40 clock cycles accounting for 2 clocks from rcall
98  ;*********************************************************************
99
100 delay_18ms:         ; 72000 clocks total (2 from rcall)
101     push r16                ; 1 clock
102     push r17                ; 1 clock
103     ldi r16, 106            ; 1 clock (m)
104     outer_18ms:             ; (4 + N)m - 1 = 71867 clocks
```

```
105         ldi r17, 225            ; 1 clock (n)
106         inner_18ms:             ; 3n - 1 = 674 clocks (N)
107             dec r17                 ; 1 clock
108             brne inner_18ms         ; 1/2 clocks
109         dec r16                 ; 1 clock
110         brne outer_18ms         ; 2/1 clocks
111         rcall delay_30us    ; 120 clocks
112     pop r17                 ; 2 clocks
113     pop r16                 ; 2 clocks
114     ret                     ; 4 clocks
115
116 delay_80us:         ; 320 clocks total (2 from rcall)
117     push r16                ; 1 clock
118     ldi r16, 103            ; 1 clock (n)
119     loop_80us:              ; 3n - 1 = 308 clocks
120         dec r16                 ; 1 clock
121         brne loop_20us          ; 2/1 clocks
122     nop nop                 ; 2 clock
123     pop r16                 ; 2 clocks
124     ret                     ; 4 clocks
125
126 delay_50us:         ; 200 clocks total (2 from rcall)
127     push r16                ; 1 clock
128     ldi r16, 63             ; 1 clock (n)
129     loop_50us:              ; 3n - 1 = 188 clocks
130         dec r16                 ; 1 clock
131         brne loop_50us          ; 2/1 clocks
132     nop nop                 ; 2 clock
133     pop r16                 ; 2 clocks
134     ret                     ; 4 clocks
135
136 delay_30us:         ; 80 clocks total (2 from rcall)
137     push r16                ; 1 clock
138     ldi r16, 34             ; 1 clock (n)
139     loop_30us:              ; 3n - 1 = 110 clocks
140         dec r16                 ; 1 clock
141         brne loop_20us          ; 2/1 clocks
142     pop r16                 ; 2 clocks
143     ret                     ; 4 clocks
144
145 delay_20us:         ; 80 clocks total (2 from rcall)
146     push r16                ; 1 clock
147     ldi r16, 23             ; 1 clock (n)
148     loop_20us:              ; 3n - 1 = 68 clocks
149         dec r16                 ; 1 clock
150         brne loop_20us          ; 2/1 clocks
151     nop nop                 ; 2 clock
152     pop r16                 ; 2 clocks
153     ret                     ; 4 clocks
```