**Lab 1: Git version-control system, AVR tools – David Garcia Torre**

1. Submit the GitHub link to your Digital-electronics-2 repository.

<https://github.com/davidgarcia23/digital-electronics-2>

1. What is the meaning of |, &, ^, ~, <<, >> binary operators? Write a truth table and explain the use of operators with examples.

| 🡪 OR

|  |  |  |
| --- | --- | --- |
| 0 | 0 | 0 |
| 0 | **1** | **1** |
| 1 | **0** | **1** |
| 1 | **1** | **1** |

& 🡪 AND

|  |  |  |
| --- | --- | --- |
| 0 | 0 | 0 |
| 0 | **1** | **0** |
| 1 | **0** | **0** |
| 1 | **1** | **1** |

^ 🡪 XOR

|  |  |  |
| --- | --- | --- |
| 0 | 0 | 0 |
| 0 | **1** | **1** |
| 1 | **0** | **1** |
| 1 | **1** | **0** |

~ 🡪 Complement

|  |  |
| --- | --- |
| 0 | 1 |
| 1 | **0** |

<< 🡪 Left shifting.

0001 << 2 🡪 0100

1. Morse code application.
2. \*
3. \* morse.c
4. \* Author : TheGT23
5. \*/
6. /\* Defines -----------------------------------------------------------\*/
7. #define LED\_GREEN PB5 // AVR pin where green LED is connected
8. #define SHORT\_DELAY 500 // Delay in milliseconds
9. #define LONG\_DELAY 1000 // Delay in milliseconds
10. #ifndef *F\_CPU*
11. #define *F\_CPU* 16000000 // CPU frequency in Hz required for delay
12. #endif
13. /\* Includes ----------------------------------------------------------\*/
14. #include <util/delay.h> // Functions for busy-wait delay loops
15. #include <avr/io.h> // AVR device-specific IO definitions
16. #include <avr/io.h>
18. int main(void){
19. // Set pin as output in Data Direction Register
20. // DDRB = DDRB or 0010 0000
21. DDRB = DDRB | (1<<LED\_GREEN);
22. // Set pin LOW in Data Register (LED off)
23. // PORTB = PORTB and 1101 1111
24. PORTB = PORTB & ~(1<<LED\_GREEN);
26. for(;;) {
28. // Letter D
30. //LONG
32. // Set pin HIGH in Data Register (LED on)
33. // PORTB = PORTB and 1101 1111
34. PORTB = PORTB | (1<<LED\_GREEN);
36. // Pause several milliseconds
37. *\_delay\_ms*(LONG\_DELAY);
39. // Set pin LOW in Data Register (LED off)
40. // PORTB = PORTB and 1101 1111
41. PORTB = PORTB & ~(1<<LED\_GREEN);
43. // Pause several milliseconds
44. *\_delay\_ms*(SHORT\_DELAY);
46. //SHORT
48. // Set pin HIGH in Data Register (LED on)
49. // PORTB = PORTB and 1101 1111
50. PORTB = PORTB | (1<<LED\_GREEN);
52. // Pause several milliseconds
53. *\_delay\_ms*(SHORT\_DELAY);
55. // Set pin LOW in Data Register (LED off)
56. // PORTB = PORTB and 1101 1111
57. PORTB = PORTB & ~(1<<LED\_GREEN);
59. // Pause several milliseconds
60. *\_delay\_ms*(SHORT\_DELAY);
62. //SHORT
64. // Set pin HIGH in Data Register (LED on)
65. // PORTB = PORTB and 1101 1111
66. PORTB = PORTB | (1<<LED\_GREEN);
68. // Pause several milliseconds
69. *\_delay\_ms*(SHORT\_DELAY);
71. // Set pin LOW in Data Register (LED off)
72. // PORTB = PORTB and 1101 1111
73. PORTB = PORTB & ~(1<<LED\_GREEN);
75. // Pause several milliseconds
76. *\_delay\_ms*(SHORT\_DELAY);

79. // Letter E
81. //SHORT
83. // Set pin HIGH in Data Register (LED on)
84. // PORTB = PORTB and 1101 1111
85. PORTB = PORTB | (1<<LED\_GREEN);
87. // Pause several milliseconds
88. *\_delay\_ms*(SHORT\_DELAY);
90. // Set pin LOW in Data Register (LED off)
91. // PORTB = PORTB and 1101 1111
92. PORTB = PORTB & ~(1<<LED\_GREEN);
94. // Pause several milliseconds
95. *\_delay\_ms*(SHORT\_DELAY);

98. // Number 2
100. //SHORT
102. // Set pin HIGH in Data Register (LED on)
103. // PORTB = PORTB and 1101 1111
104. PORTB = PORTB | (1<<LED\_GREEN);
106. // Pause several milliseconds
107. *\_delay\_ms*(SHORT\_DELAY);
109. // Set pin LOW in Data Register (LED off)
110. // PORTB = PORTB and 1101 1111
111. PORTB = PORTB & ~(1<<LED\_GREEN);
113. // Pause several milliseconds
114. *\_delay\_ms*(SHORT\_DELAY);
116. //SHORT
118. // Set pin HIGH in Data Register (LED on)
119. // PORTB = PORTB and 1101 1111
120. PORTB = PORTB | (1<<LED\_GREEN);
122. // Pause several milliseconds
123. *\_delay\_ms*(SHORT\_DELAY);
125. // Set pin LOW in Data Register (LED off)
126. // PORTB = PORTB and 1101 1111
127. PORTB = PORTB & ~(1<<LED\_GREEN);
129. // Pause several milliseconds
130. *\_delay\_ms*(SHORT\_DELAY);
132. //LONG
134. // Set pin HIGH in Data Register (LED on)
135. // PORTB = PORTB and 1101 1111
136. PORTB = PORTB | (1<<LED\_GREEN);
138. // Pause several milliseconds
139. *\_delay\_ms*(LONG\_DELAY);
141. // Set pin LOW in Data Register (LED off)
142. // PORTB = PORTB and 1101 1111
143. PORTB = PORTB & ~(1<<LED\_GREEN);
145. // Pause several milliseconds
146. *\_delay\_ms*(SHORT\_DELAY);
148. //LONG
150. // Set pin HIGH in Data Register (LED on)
151. // PORTB = PORTB and 1101 1111
152. PORTB = PORTB | (1<<LED\_GREEN);
154. // Pause several milliseconds
155. *\_delay\_ms*(LONG\_DELAY);
157. // Set pin LOW in Data Register (LED off)
158. // PORTB = PORTB and 1101 1111
159. PORTB = PORTB & ~(1<<LED\_GREEN);
161. // Pause several milliseconds
162. *\_delay\_ms*(SHORT\_DELAY);
164. //LONG
166. // Set pin HIGH in Data Register (LED on)
167. // PORTB = PORTB and 1101 1111
168. PORTB = PORTB | (1<<LED\_GREEN);
170. // Pause several milliseconds
171. *\_delay\_ms*(LONG\_DELAY);
173. // Set pin LOW in Data Register (LED off)
174. // PORTB = PORTB and 1101 1111
175. PORTB = PORTB & ~(1<<LED\_GREEN);
177. // Pause several milliseconds
178. *\_delay\_ms*(SHORT\_DELAY);
180. }
181. }

* Screenshot of SimulIDE circuit.

