Miles Van de Wetering (vandewmi)

Matthew Poposwki (popowskm)

Lab 3 Report

1) Initial DDRB: 0x00
2) Initial PORTB: 0x00
3) Port B is all outputs set to 0.
4) Stack Pointer: 0x10FF
5) R00: 0xFF
6) The Program looped 4 times.
7) To change the number of loops, change: ldi I, $04
8) R01: 0xAA
9) R02: 0x0F
10) R03: 0x0F
11) Stack Pointer: 0x10FD
12) ba0e

Challenge:

1) This function adds two words (two sets of two bytes) together. It does this by indirect reference using x and y, with z storing the result. First, A gets M(X), and X is incremented. Then, B gets M(Y), and Y is incremented. Then, M(Z) gets A + B, so we've added the lower bytes of each word. Then, Z is incremented and A gets M(X) and B gets M(Y), and the new M(Z) gets A + B + the carry out from the previous operation, and Z is incremented one last time. If this final addition has a carry, instead of exiting we store the value of XH (which happens to be 01) in M(Z).
2) For example, if we added FF and FF, the first 'exit' branch would be skipped and there would be a carry bit stored at the final Z.
3) This instruction stores the carry bit.