// PROJECT  :ReverseBits

// PURPOSE  :Inline assembly code to reverse the order of the bits in a byte

// AUTHOR   :C. D'Arcy

// DATE     :2025 02 13

// uC       :\*

// COURSE   :ICS4U-E

// STATUS   :Working

// REFERENCE:http://darcy.rsgc.on.ca/ACES/Datasheets/InstructionSetSummary.pdf

volatile uint8\_t data = B11100001;  //source byte to be reversed

volatile uint8\_t reversed = 0;      //target byte for reversal

void setup() {

  Serial.begin(9600);

  while (!Serial);

  asm(

    "   lds  r18,(data)       \n" //load the SRAM variable into a GP register

    "   clr  r17              \n" //zero the target register of the reversal

    "   ldi  r16,8            \n" //set the loop control variable for 8 iterations

    "1: rol  r18              \n" //rotate source data left to load C bit of SREG

    "   ror  r17              \n" //rotate target variable right to obtain C bit

    "   dec  r16              \n" //one less iteration

    "   brne 1b               \n" //if we're not done go back

    "   sts  (reversed),r17   \n" //store reversed data into the target variable

  );

  Serial.println("Before:\t" + String(data, BIN));   //confirm source is preserved

  Serial.println("After:\t" + String(reversed, BIN)); //conform target is reversed

}

void loop() {}