

```

int led = 13; // led to use for output (13 is built-in)

int dit_delay = 500; // length of a dit in ms
int dah_delay = dit_delay * 3; // length of a dah in ms

// dit() and dat() already include dit_delay
int letter_delay = dah_delay - dit_delay;

// letters already include letter delay
int word_delay = dit_delay * 7 - letter_delay;

void dit() {
    Serial.print(".");

    // send a dit
    digitalWrite(led, HIGH);
    delay(dit_delay);

    digitalWrite(led, LOW);
    delay(dit_delay);
}

void dah() {
    Serial.print("-");

    // send a dah
    digitalWrite(led, HIGH);
    delay(dah_delay);

    digitalWrite(led, LOW);
    delay(dit_delay);
}

void pause_letter() {
    Serial.println("");
    delay(letter_delay);
}

void pause_word() {
    Serial.println("");
    delay(word_delay);
}

void morse_S() {
    dit(); dit(); dit();
    pause_letter();
}

void morse_0() {
    dah(); dah(); dah();
    pause_letter();
}

void morse_A(){
    dit(); dah();
}

```

```
    pause_letter();
}

void morse_R(){
    dit(); dah(); dit();
    pause_letter();
}

void morse_L(){
    dit(); dah(); dit(); dit();
    pause_letter();
}

void morse_N(){
    dah(); dit();
    pause_letter();
}

void morse_D(){
    dah(); dit(); dit();
    pause_letter();
}

void morse_U(){
    dit(); dit(); dah();
    pause_letter();
}

void morse_I(){
    dit(); dit();
    pause_letter();
}

void morse_E(){
    dit();
    pause_letter();
}

void morse_V(){
    dit(); dit(); dit(); dah();
    pause_letter();
}

void morse_T(){
    dah();
    pause_letter();
}

void morse_Y(){
    dah(); dit(); dah(); dah();
    pause_letter();
}

void morse_SOS() {
    morse_S(); morse_O(); morse_S();
}
```

```

    pause_word();
}

void morse_Saarland() {
    morse_S(); morse_A(); morse_A(); morse_R(); morse_L(); morse_A(); morse_N(); morse_L();
    pause_word();
}

void morse_University() {
    morse_U(); morse_N(); morse_I(); morse_V(); morse_E(); morse_R(); morse_S(); morse_L();
    pause_word();
}

// send n in morse code
void morse_digit(int n) {
    Serial.print("morse_digit(");
    Serial.print(n);
    Serial.println(")...");

    if (n == 0) {
        dah(); dah(); dah(); dah(); dah();
    }
    if (n == 1) {
        dit(); dah(); dah(); dah(); dah();
    }
    if (n == 2) {
        dit(); dit(); dah(); dah(); dah();
    }
    if (n == 3) {
        dit(); dit(); dit(); dah(); dah();
    }
    if (n == 4) {
        dit(); dit(); dit(); dit(); dah();
    }
    if (n == 5) {
        dit(); dit(); dit(); dit(); dit();
    }
    if (n == 6) {
        dah(); dit(); dit(); dit(); dit();
    }
    if (n == 7) {
        dah(); dah(); dit(); dit(); dit();
    }
    if (n == 8) {
        dah(); dah(); dah(); dit(); dit();
    }
    if (n == 9) {
        dah(); dah(); dah(); dah(); dit();
    }
    pause_letter();

    Serial.print("morse_digit(");
    Serial.print(n);
    Serial.println(")...done");
}

```

```

void morse_digit_even(int n){
    if (n % 2 == 0){
        Serial.print("\ndigit ");
        Serial.print(n);
        Serial.print(" is even!  \n");
        morse_digit(n);
    }else{
        Serial.print("\ndigit ");
        Serial.print(n);
        Serial.print(" is NOT even!  \n");
    }
}

void morse_number(int n) {
    Serial.print("morse_number(");
    Serial.print(n);
    Serial.println(")...");

    if (n >= 10) {
        morse_number(n / 10);
    }
    morse_digit(n % 10);

    Serial.print("morse_number(");
    Serial.print(n);
    Serial.println(")...done");
}

void setup() {
    // put your setup code here, to run once:

    // Configure LED as output
    pinMode(led, OUTPUT);

    // Use serial port as output
    Serial.begin(9600);
}

void loop() {
    // put your main code here, to run repeatedly:
    morse_digit_even(9);
    morse_digit_even(8);
    morse_digit_even(7);
    morse_digit_even(6);
    morse_digit_even(5);
    morse_digit_even(4);
    morse_digit_even(3);
    morse_digit_even(2);
    morse_digit_even(1);
    morse_digit_even(0);
}

```