Les 9

Morsecode

const int buzzerPin = 9;

const int ledPin = 7;

// tone frequency C

const int tonefreq = 523;

const int dotlength = 100;

const int dashlength = dotlength \* 3;

const int inter = dotlength;

const int lgap = dotlength \* 2;

const int wgap = dotlength \* 4;

void setup()

{

pinMode(buzzerPin, OUTPUT);

pinMode(ledPin, OUTPUT);

Serial.begin(9600);

}

void loop()

{

char thischar;

if (Serial.available())

{

thischar = Serial.read();

if (thischar>='a' && thischar<='z')

{

thischar = thischar -32;

}

if(thischar<65 || thischar>90)

{

thischar=' ';

}

soundLetter(thischar);

delay(lgap);

}

}

void dot()

{

tone(buzzerPin, tonefreq);

digitalWrite(ledPin, HIGH);

delay(dotlength);

noTone(buzzerPin);

digitalWrite(ledPin, LOW);

delay(inter);

}

void dash()

{

tone(buzzerPin, tonefreq);

digitalWrite(ledPin, HIGH);

delay(dashlength);

noTone(buzzerPin);

digitalWrite(ledPin, LOW);

delay(inter);

}

void soundLetter(char letter)

{

switch(letter)

{

case 'E':

dot();

return;

case 'T':

dash();

return;

case 'A':

dot();

dash();

return;

case 'O':

dash();

dash();

dash();

return;

case 'I':

dot();

dot();

return;

case 'N':

dash();

dot();

return;

case 'S':

dot();

dot();

dot();

return;

case 'H':

dot();

dot();

dot();

dot();

return;

case 'R':

dot();

dash();

dot();

return;

case 'D':

dash();

dot();

dot();

return;

case 'L':

dot();

dash();

dot();

dot();

return;

case 'C':

dash();

dot();

dash();

dot();

return;

case 'U':

dot();

dot();

dash();

return;

case 'M':

dash();

dash();

return;

case 'W':

dot();

dash();

dash();

return;

case 'F':

dot();

dot();

dash();

dot();

return;

case 'G':

dash();

dash();

dot();

return;

case 'Y':

dash();

dot();

dash();

dash();

return;

case 'P':

dot();

dash();

dash();

dot();

return;

case 'B':

dash();

dot();

dot();

dot();

return;

case 'V':

dot();

dot();

dot();

dash();

return;

case 'K':

dash();

dot();

dash();

return;

case 'J':

dot();

dash();

dash();

dash();

return;

case 'X':

dash();

dot();

dot();

dash();

return;

case 'Q':

dash();

dash();

dot();

dash();

return;

case 'Z':

dash();

dash();

dot();

dot();

return;

case ' ':

delay(wgap);

return;

}

}