

```

#include <Arduino.h> // Инициализация специфичной для Ардуино
библиотеки
// -----
==-- //
// Светодиодная волна для ArduinoDevBoard 2.0
// Вариант № 1
// -----
==-- //

void setup() { // Инициализация настроек. Выполняется однократно
  pinMode(8, OUTPUT); // Настраиваем пин 8 на управление (ВЫХОД)
  pinMode(9, OUTPUT);
  pinMode(10, OUTPUT);
  pinMode(11, OUTPUT);
  pinMode(12, OUTPUT);
}

void loop() { // Бесконечный цикл
  digitalWrite(8, HIGH); // HIGH - 5V, строка включает светодиод
  delay(200); // Задержка 0,2 сек.
  digitalWrite(8, LOW); // LOW - 0V, строка выключает светодиод
  delay(200);
  digitalWrite(9, HIGH);
  delay(200);
  digitalWrite(9, LOW);
  delay(200);
  digitalWrite(10, HIGH);
  delay(200);
  digitalWrite(10, LOW);
  delay(200);
  digitalWrite(11, HIGH);
  delay(200);
  digitalWrite(11, LOW);
  delay(200);
  digitalWrite(12, HIGH);
  delay(200);
  digitalWrite(12, LOW);
  delay(200);
}

```

```

#include <Arduino.h> // Инициализация специфичной для Ардуино
библиотеки
// -----
==-- //
// Светодиодная волна для ArduinoDevBoard 2.0
// Вариант № 2
// -----
==-- //

void setup() { // Инициализация настроек. Выполняется однократно
  pinMode(8, OUTPUT); // Настраиваем пин 8 на управление (ВЫХОД)
  pinMode(9, OUTPUT);
  pinMode(10, OUTPUT);
  pinMode(11, OUTPUT);
  pinMode(12, OUTPUT);
}

void loop() { // Бесконечный цикл
  // ! - восклицательный знак означает логическое отрицание.
  digitalWrite(8, !digitalRead(8)); // Считываем состояние пина № 8 и
инвертируем (изменяем) его значение.
  delay(200);
  digitalWrite(9, !digitalRead(8)); // Считываем состояние пина № 8 и
инвертируем (изменяем) его значение.
  delay(200);
  digitalWrite(10, !digitalRead(8)); // Считываем состояние пина № 8 и
инвертируем (изменяем) его значение.
  delay(200);
  digitalWrite(11, !digitalRead(8)); // Считываем состояние пина № 8 и
инвертируем (изменяем) его значение.
  delay(200);
  digitalWrite(12, !digitalRead(8)); // Считываем состояние пина № 8 и
инвертируем (изменяем) его значение.
  delay(200);
}

```

```
#include <Arduino.h> // Инициализация специфичной для Ардуино
библиотеки
// -----
// ----- //
// Светодиодная волна для ArduinoDevBoard 2.0
// Вариант № 3
// -----
// ----- //
```

```
void setup() { // Инициализация настроек. Выполняется однократно
  pinMode(8, OUTPUT); // Настраиваем пин 8 на управление (ВЫХОД)
  pinMode(9, OUTPUT);
  pinMode(10, OUTPUT);
  pinMode(11, OUTPUT);
  pinMode(12, OUTPUT);
}
```

```
void loop() { // Бесконечный цикл
  for (int i = 8; i < 12; ++i) // ++i идеинтично i = i + i
  {
    // ! - восклицательный знак означает логическое отрицание.
    digitalWrite(i, !digitalRead(i)); // Считываем состояние пина и
инвертирует (изменяет) его значение.
    delay(200);
  }
}
```

```
/*
  Анализ кода.
  Оператор for обеспечивает условное циклическое (итеральное)
выполнение действий.
```

```
for (П = НЗ; П != КЗ; П += Ш)
{
  операция;
}
где П — параметр, НЗ — начальное значение, КЗ — конечное значение
(в общем случае — условие продолжения цикла, Ш — шаг):
*/
```

```

#include <Arduino.h> // Инициализация специфичной для Ардуино
библиотеки
// -----
==- //
// Светодиодная волна для ArduinoDevBoard 2.0
// Вариант № 4 с использованием массива чисел
// -----
==- //

void setup() { // Инициализация настроек. Выполняется однократно
  pinMode(8, OUTPUT); // Настраиваем пин 8 на управление (ВЫХОД)
  pinMode(9, OUTPUT);
  pinMode(10, OUTPUT);
  pinMode(11, OUTPUT);
  pinMode(12, OUTPUT);
}

void loop() { // Бесконечный цикл
  const int leds[5] = {8, 9, 10, 11, 12};
  // В C++ действует правило - всюду, где только можно ставьте const. Это
  повышает надёжность программы
  for (int ledNum = 0; ledNum < 5; ++ledNum) // Переменная ledNum хранит
  номер позиции в массиве. For - перебирает номера в массиве leds
  {
    digitalWrite(leds[ledNum], !digitalRead(leds[ledNum]));
    delay(200);
  }
}

/*
Анализ кода.
// ++i идентично i = i + i
{
  // ! - восклицательный знак означает логическое отрицание.
  digitalWrite(i, !digitalRead(i)); // Считываем состояние пина № 8 и
инвертируем (изменяем) его значение.
  delay(200);
}
Оператор for обеспечивает условное циклическое (итеральное)
выполнение действий.

```

```
for (П = НЗ; П != КЗ; П += Ш)
{
    операция;
}
```

где П — параметр, НЗ — начальное значение, КЗ — конечное значение  
(в общем случае — условие продолжения цикла, Ш — шаг):

Массив - коллекция элементов. Все элементы массива - одного типа.

```
const int leds[5] = {8, 9, 10, 11, 12}; // Где 5 - это размер массива (сколько в
массиве чисел), а {} - содержимое массива
*/
```

```

#include <Arduino.h> // Инициализация специфичной для Ардуино
библиотеки
// -----
==-- //
// Светодиодная волна для ArduinoDevBoard 2.0
// Вариант № 5 с random
// -----
==-- //

void setup() { // Инициализация настроек. Выполняется однократно
  pinMode(8, OUTPUT); // Настраиваем пин 8 на управление (ВЫХОД)
  pinMode(9, OUTPUT);
  pinMode(10, OUTPUT);
  pinMode(11, OUTPUT);
  pinMode(12, OUTPUT);
}

void loop() { // Бесконечный цикл
  int ledNum = 0;

  ledNum = random(8, 13);

  digitalWrite(ledNum, !digitalRead(ledNum));
  delay(200);
}

/*
  Анализ кода.
  Функция random() возвращает псевдослучайное число

  random(max);
  random(min, max);
  Где: min: нижняя граница случайных значений, включительно.
  (опционально), max: верхняя граница случайных значений, не
  включительно
*/

```

```

#include <Arduino.h> // Инициализация специфичной для Ардуино
библиотеки
// -----
==- //
// Светодиодная волна для ArduinoDevBoard 2.0
// Вариант № 6 с random и массивом чисел
// -----
==- //

void setup() { // Инициализация настроек. Выполняется однократно
  pinMode(8, OUTPUT); // Настраиваем пин 8 на управление (ВЫХОД)
  pinMode(9, OUTPUT);
  pinMode(10, OUTPUT);
  pinMode(11, OUTPUT);
  pinMode(12, OUTPUT);
}

void loop() { // Бесконечный цикл
  const int leds[5] = {8, 9, 10, 11, 12};
  int ledNum = 0;

  ledNum = random(0, 6);

  digitalWrite(leds[ledNum], !digitalRead(leds[ledNum]));
  delay(200);
}

```