

# How to Program an Arduino

## to run a 7 Segment Display – Automatically 0-9 @ 1 second intervals

Pat McMahon - V1 - 10/6/2019

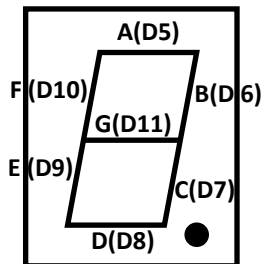
### ANSWER SHEET

Note– If the Segment is on, its HIGH. If the Segment is off, its LOW. Highlight the Segments to show the numbers 0 to 9 below, fill in HIGH or LOW in code and transfer to the Arduino IDE . Upload to run 0 to 9 Automatically, on the 7 Segment Display, at one second intervals.

Note– DelayTime =1000 milliseconds = 1 second.

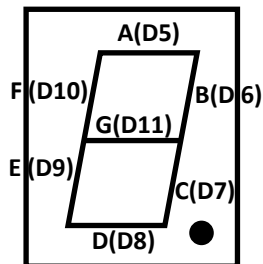
//Button 0

```
digitalWrite(APin, HIGH);
digitalWrite(BPin, HIGH);
digitalWrite(CPin, HIGH);
digitalWrite(DPin, HIGH);
digitalWrite(EPin, HIGH);
digitalWrite(FPin, HIGH);
digitalWrite(GPin, LOW);
delay(DelayTime);
```



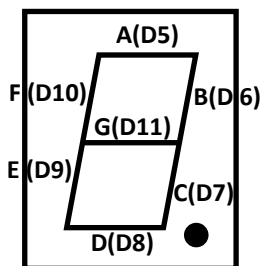
//Button 1

```
digitalWrite(APin, LOW);
digitalWrite(BPin, HIGH);
digitalWrite(CPin, HIGH);
digitalWrite(DPin, LOW);
digitalWrite(EPin, LOW);
digitalWrite(FPin, LOW);
digitalWrite(GPin, LOW);
delay(DelayTime);
```



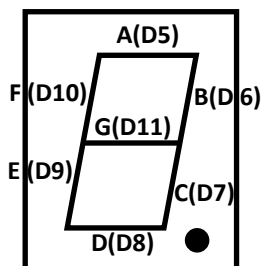
//Button 2

```
digitalWrite(APin, HIGH);
digitalWrite(BPin, HIGH);
digitalWrite(CPin, LOW);
digitalWrite(DPin, HIGH);
digitalWrite(EPin, HIGH);
digitalWrite(FPin, LOW);
digitalWrite(GPin, HIGH);
delay(DelayTime);
```



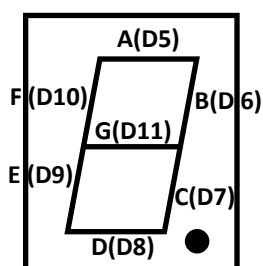
//Button 3

```
digitalWrite(APin, HIGH);
digitalWrite(BPin, HIGH);
digitalWrite(CPin, HIGH);
digitalWrite(DPin, HIGH);
digitalWrite(EPin, LOW);
digitalWrite(FPin, LOW);
digitalWrite(GPin, HIGH);
delay(DelayTime);
```



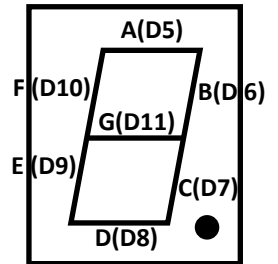
//Button 4

```
digitalWrite(APin, LOW);
digitalWrite(BPin, HIGH);
digitalWrite(CPin, HIGH);
digitalWrite(DPin, LOW);
digitalWrite(EPin, LOW);
digitalWrite(FPin, HIGH);
digitalWrite(GPin, HIGH);
delay(DelayTime);
```



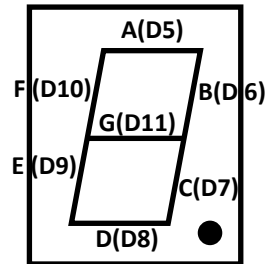
//Button 5

```
digitalWrite(APin, HIGH);
digitalWrite(BPin, LOW);
digitalWrite(CPin, HIGH);
digitalWrite(DPin, HIGH);
digitalWrite(EPin, LOW);
digitalWrite(FPin, HIGH);
digitalWrite(GPin, HIGH);
delay(DelayTime);
```



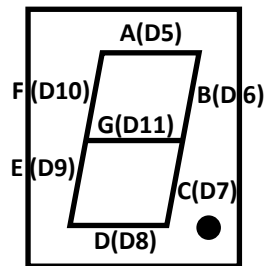
//Button 6

```
digitalWrite(APin, HIGH);
digitalWrite(BPin, LOW);
digitalWrite(CPin, HIGH);
digitalWrite(DPin, HIGH);
digitalWrite(EPin, HIGH);
digitalWrite(FPin, HIGH);
digitalWrite(GPin, HIGH);
delay(DelayTime);
```



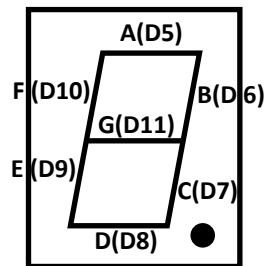
//Button 7

```
digitalWrite(APin, HIGH);
digitalWrite(BPin, HIGH);
digitalWrite(CPin, HIGH);
digitalWrite(DPin, LOW);
digitalWrite(EPin, LOW);
digitalWrite(FPin, LOW);
digitalWrite(GPin, LOW);
delay(DelayTime);
```



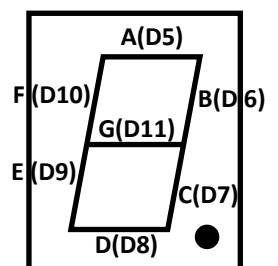
//Button 8

```
digitalWrite(APin, HIGH);
digitalWrite(BPin, HIGH);
digitalWrite(CPin, HIGH);
digitalWrite(DPin, HIGH);
digitalWrite(EPin, HIGH);
digitalWrite(FPin, HIGH);
digitalWrite(GPin, HIGH);
delay(DelayTime);
```



//Button 9

```
digitalWrite(APin, HIGH);
digitalWrite(BPin, HIGH);
digitalWrite(CPin, HIGH);
digitalWrite(DPin, LOW);
digitalWrite(EPin, LOW);
digitalWrite(FPin, HIGH);
digitalWrite(GPin, HIGH);
delay(DelayTime);
```



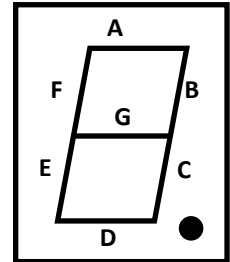
# How to Program an Arduino

## to run a 7 Segment Display – Infrared with IR Remote

Pat McMahon - V1 - 10/6/2019

### Explanation SHEET

Note– DelayTime =1000 milliseconds = 1 second.



#### Note

- Use the previous completed “How to Program an Arduino to run a 7 Segment Display 0 to 9 Automatically, at one second intervals.” Question /Answer Sheet.
- Also use the previous “Decoding a Universal Remote - Remote M” Answer Sheet- Arduino.
- Using both previous documents, see the changes required below in **Blue** to Code the 7 Segment Display from Automatic to Infrared.
- Complete for all digits 0 to 9 as below and then add counting up 0 - 9, counting Down 9 - 0 etc.

### Automatic 0 - 9 Code

```
// Button 0
digitalWrite(APin, HIGH);
digitalWrite(BPin, HIGH);
digitalWrite(CPin, HIGH);
digitalWrite(DPin, HIGH);
digitalWrite(EPin, HIGH);
digitalWrite(FPin, HIGH);
digitalWrite(GPin, LOW);
delay(DelayTime);
```

```
// Button 1
digitalWrite(APin, LOW);
digitalWrite(BPin, HIGH);
digitalWrite(CPin, HIGH);
digitalWrite(DPin, LOW);
digitalWrite(EPin, LOW);
digitalWrite(FPin, LOW);
digitalWrite(GPin, LOW);
delay(DelayTime);
```

```
// Button 2
digitalWrite(APin, HIGH);
digitalWrite(BPin, HIGH);
digitalWrite(CPin, LOW);
digitalWrite(DPin, HIGH);
digitalWrite(EPin, HIGH);
digitalWrite(FPin, LOW);
digitalWrite(GPin, HIGH);
delay(DelayTime);
```

### Infrared 0 - 9 Code

Add Infrared Code →

```
case 0x910:
    // Button 0
    digitalWrite(APin, HIGH);
    digitalWrite(BPin, HIGH);
    digitalWrite(CPin, HIGH);
    digitalWrite(DPin, HIGH);
    digitalWrite(EPin, HIGH);
    digitalWrite(FPin, HIGH);
    digitalWrite(GPin, LOW);
    break;
```

Change delay(DelayTime); to break; →

Add Infrared Code →

```
case 0x010:
    // Button 1
    digitalWrite(APin, LOW);
    digitalWrite(BPin, HIGH);
    digitalWrite(CPin, HIGH);
    digitalWrite(DPin, LOW);
    digitalWrite(EPin, LOW);
    digitalWrite(FPin, LOW);
    digitalWrite(GPin, LOW);
    break;
```

Change delay(DelayTime); to break; →

Add Infrared Code →

```
case 0x810:
    // Button 2
    digitalWrite(APin, HIGH);
    digitalWrite(BPin, HIGH);
    digitalWrite(CPin, LOW);
    digitalWrite(DPin, HIGH);
    digitalWrite(EPin, HIGH);
    digitalWrite(FPin, LOW);
    digitalWrite(GPin, HIGH);
    break;
```

Change delay(DelayTime); to break; →

# How to Program an Arduino

## to run a 7 Segment Display – Automatically 0-9 @ 1 second intervals

Pat McMahon - V1 - 10/6/2019

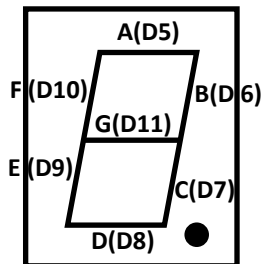
### QUESTION SHEET

Note– If the Segment is on, its HIGH. If the Segment is off, its LOW. Highlight the Segments to show the numbers 0 to 9 below, fill in HIGH or LOW in code and transfer to the Arduino IDE . Upload to run 0 to 9 Automatically, on the 7 Segment Display, at one second intervals.

Note– DelayTime =1000 milliseconds = 1 second.

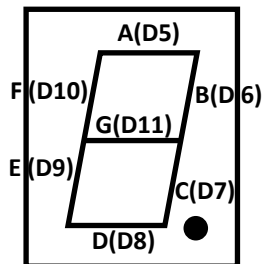
//Button 0

```
digitalWrite(APin, );
digitalWrite(BPin, );
digitalWrite(CPin, );
digitalWrite(DPin, );
digitalWrite(EPin, );
digitalWrite(FPin, );
digitalWrite(GPin, );
delay(DelayTime);
```



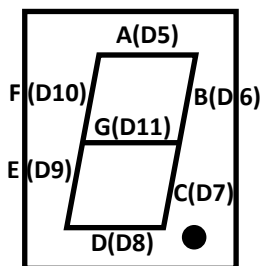
//Button 1

```
digitalWrite(APin, );
digitalWrite(BPin, );
digitalWrite(CPin, );
digitalWrite(DPin, );
digitalWrite(EPin, );
digitalWrite(FPin, );
digitalWrite(GPin, );
delay(DelayTime);
```



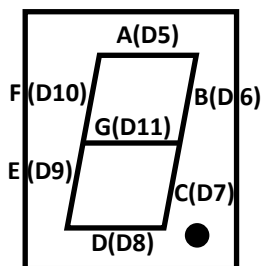
//Button 2

```
digitalWrite(APin, );
digitalWrite(BPin, );
digitalWrite(CPin, );
digitalWrite(DPin, );
digitalWrite(EPin, );
digitalWrite(FPin, );
digitalWrite(GPin, );
delay(DelayTime);
```



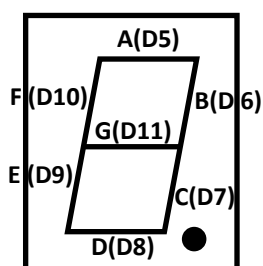
//Button 3

```
digitalWrite(APin, );
digitalWrite(BPin, );
digitalWrite(CPin, );
digitalWrite(DPin, );
digitalWrite(EPin, );
digitalWrite(FPin, );
digitalWrite(GPin, );
delay(DelayTime);
```



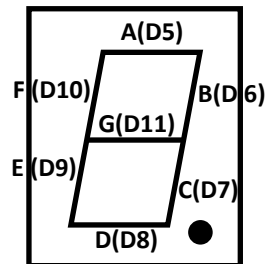
//Button 4

```
digitalWrite(APin, );
digitalWrite(BPin, );
digitalWrite(CPin, );
digitalWrite(DPin, );
digitalWrite(EPin, );
digitalWrite(FPin, );
digitalWrite(GPin, );
delay(DelayTime);
```



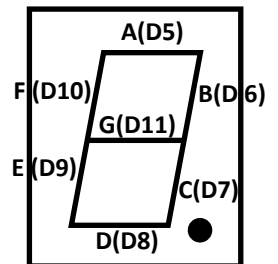
//Button 5

```
digitalWrite(APin, );
digitalWrite(BPin, );
digitalWrite(CPin, );
digitalWrite(DPin, );
digitalWrite(EPin, );
digitalWrite(FPin, );
digitalWrite(GPin, );
delay(DelayTime);
```



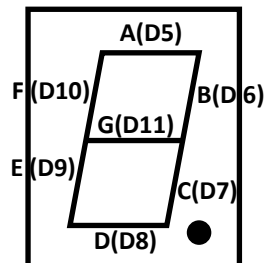
//Button 6

```
digitalWrite(APin, );
digitalWrite(BPin, );
digitalWrite(CPin, );
digitalWrite(DPin, );
digitalWrite(EPin, );
digitalWrite(FPin, );
digitalWrite(GPin, );
delay(DelayTime);
```



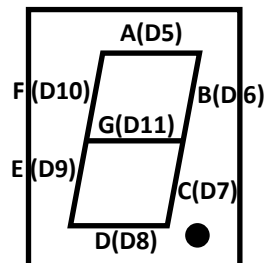
//Button 7

```
digitalWrite(APin, );
digitalWrite(BPin, );
digitalWrite(CPin, );
digitalWrite(DPin, );
digitalWrite(EPin, );
digitalWrite(FPin, );
digitalWrite(GPin, );
delay(DelayTime);
```



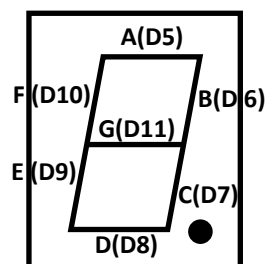
//Button 8

```
digitalWrite(APin, );
digitalWrite(BPin, );
digitalWrite(CPin, );
digitalWrite(DPin, );
digitalWrite(EPin, );
digitalWrite(FPin, );
digitalWrite(GPin, );
delay(DelayTime);
```



//Button 9

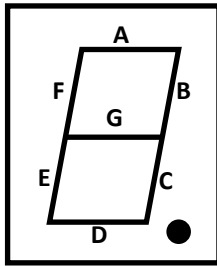
```
digitalWrite(APin, );
digitalWrite(BPin, );
digitalWrite(CPin, );
digitalWrite(DPin, );
digitalWrite(EPin, );
digitalWrite(FPin, );
digitalWrite(GPin, );
delay(DelayTime);
```



# Pin/Wire Connections for the 7 Segment Displays (Small & Jumbo, for Arduino & Picaxe)

Pat McMahon - V1 - 10/6/2019

Note- This sheet is to assist with soldering Wire Connections to the 7 Segment Displays ( view from back) and coding pin connections information, for the Arduino UNO / Nano and the Picaxe 14M2 Micro-controllers.

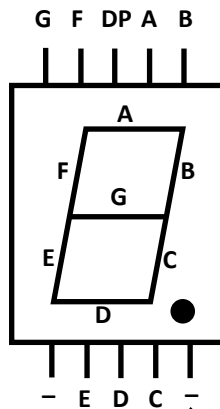


View from the front.

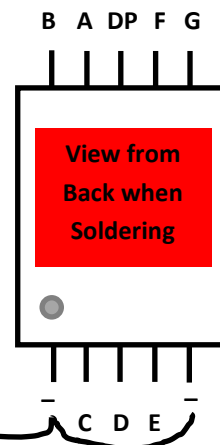
7 Segment	Arduino Nano	Arduino UNO	Picaxe 14M2
A	D5	5	0
B	D6	6	1
C	D7	7	2
D	D8	8	3
E	D9	9	4
F	D10	10	5
G	D11	11	C.0

Note - DP is for Decimal Point, don't connect for single display, leave free.

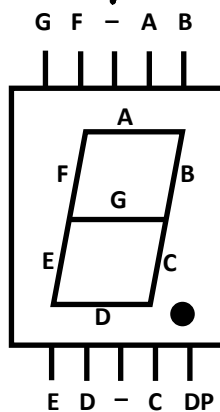
**JUMBO**  
7 Segment  
(70 mm high)  
Common Cathode



View from the front.



**SMALL**  
7 Segment  
(13 mm high)  
Common Cathode



View from the front.

