

[](#)

[](#)

[
](#)

[<div style="text-align: center;">](#)

[\[KEMBALI KE MENU
SEBELUMNYA\]](http://muhammaddimas211006.blogspot.com)[</div>](#)

[
](#)

[<center>](#)

[<div data-darkreader-inline-bgcolor="" data-darkreader-inline-border-bottom=""
data-darkreader-inline-border-left="" data-darkreader-inline-border-right=""
data-darkreader-inline-border-top="" style="--darkreader-inline-bgcolor: #191b1c;
--darkreader-inline-border-bottom: #1264ad; --darkreader-inline-border-left: #1264ad;
--darkreader-inline-border-right: #1264ad; --darkreader-inline-border-top: #1264ad;
background-color: white; border: 2px dashed rgb\(23, 128, 221\); height: 240px; overflow: auto;
padding: 10px; text-align: center; width: 330px;">](#)

[DAFTAR ISI](#)

[
](#)

[<div style="text-align: left;">](#)

[1. Prosedur</div>](#)

[<div style="text-align: left;">](#)

[2. Hardware dan Diagram Blok</div>](#)

[<div style="text-align: left;">](#)

[3. Rangkaian Simulasi dan Prinsip Kerja</div>](#)

[<div style="text-align: left;">](#)

[4. Flowchart dan Listing Program</div>](#)

[<div style="text-align: left;">](#)

[5. Kondisi</div>](#)

[<div style="text-align: left;">](#)

[6. Video Demo</div>](#)

[<div style="text-align: left;">](#)

[7. Soal Analisa</div>](#)

[<div style="text-align: left;">](#)

[8. Download File
](#)

src="https://blogger.googleusercontent.com/img/b/R29vZ2xl/AVvXsEifF22gdMGMmf3PCxWFkuYVfd
uLnk_dQWDqVhCXTdXSC1m-3_d1Hz7gvuotMoNX2f8pCSiddKdVFQOpFVJc435LaIvagJaVsspwJmBI4zy
ekuUkuBFMf7w56Mq-kkQ9qto670zKRPNOxHknyPAAZ1aZpjmOgYl3NngxIBC6VK7hW7TPe29dGs7pym
thDGo/s320/IMG_20240507_091611.jpg" width="241" /></div>
<div class="separator"
style="clear: both; text-align: center;">
</div>
<div class="separator" style="clear: both;
text-align: center;">
</div>
<div class="separator" style="clear: both; text-align:
justify;">1. Arduino</div><div class="separator" style="clear: both; text-align: center;"><a
href="https://blogger.googleusercontent.com/img/b/R29vZ2xl/AVvXsEiP2FZNlli2vS4dFiXq0-TqMce8N
FGqGqyfqNkyDWOwb04v1aWvCkNuTS4eFfspejRNiXBp7CVYghgByA9vR6XI7zhy2b_tMtaAgUrwfPMUP
vkzmZ4Qe1Ow9unKrRqt0bnjEikhQCRekqYjt_2JA3fOQRZatDXqrYN9fW3_7v4VeqEnHgNC0YQVi3ILWcY
/s320/arudino.png" style="color: #280f53; font-family: "Times New Roman"; font-size:
13.2px; margin-left: 1em; margin-right: 1em; text-decoration-line: none;"></div><div
class="separator" style="clear: both; text-align: center;">
</div><div class="separator"
style="clear: both; text-align: center;"><a
href="https://blogger.googleusercontent.com/img/b/R29vZ2xl/AVvXsEjnhqhx40cpMFICKDeu9OWQX
luIRKMBt8vflDDDe5EsAFh1J7rUDsXNKxILa2mPsAg1wMHJA9UDmd4W2ZiSkaoFRS3Kgdle-fapdf2JOFa
pMebFHKu0VvbWIGzBNpawb41htu11y27hiYLnS_EGgk5QkrXv3t8ZnF9gsAGkRYLbWi5oUBuqURBEAx
Gaoi4/s320/WhatsApp%20Image%202024-04-04%20at%2016.42.00_b85418c1.jpg"
style="margin-left: 1em; margin-right: 1em;"></div>
<div class="separator" style="clear: both; text-align: center;">
</div><div
class="separator" style="clear: both; text-align: justify;">
</div><div class="separator"
style="clear: both; text-align: center;">
</div>
<div class="separator" style="clear: both;
text-align: center;">
</div><div class="separator" style="clear: both; text-align: justify;">2.
Seven Segment 2 Digit</div><div class="separator" style="clear: both; text-align: center;"><a
href="https://blogger.googleusercontent.com/img/b/R29vZ2xl/AVvXsEj64R1uAufJdhEsRerrZyYno2JzZ
fsp-9fqh1hGzQUSsowel4H6mRtbv4-OUXiS-xODVlxtGnRviuCdeJngAPpSVnk4nkaJ0Srmq1p-qqD5TySw
7ZN7VyK3gXfwaKj8ltqXoV0U5Oj0n7kgrRxdE5nAKRzBADpwevW0i11M-ThSwa5ArcSDoCSwcXB1I9Q/s
224/download.png" style="color: #280f53; font-size: 13.2px; margin-left: 1em; margin-right: 1em;
text-decoration-line: none;"></div><div class="separator" style="clear: both; text-align: center;">
</div><div class="separator" style="clear: both; text-align: center;"></div>
<div class="separator" style="clear: both; text-align: justify;">3. Breadboard</div>
<div class="separator" style="clear: both; text-align: center;"></div>4. Dipswitch</div><div style="text-align: center;"><div class="separator" style="clear: both; text-align: center;"></div><div class="separator" style="clear: both; text-align: center;">
</div><div class="separator" style="clear: both; text-align: center;"></div>
<div class="separator" style="clear: both; text-align: center;">
</div><div class="separator" style="clear: both; text-align: center;">
</div><div class="separator" style="clear: both; text-align: center;">
</div><div class="separator" style="clear: both; text-align: justify;">
</div>

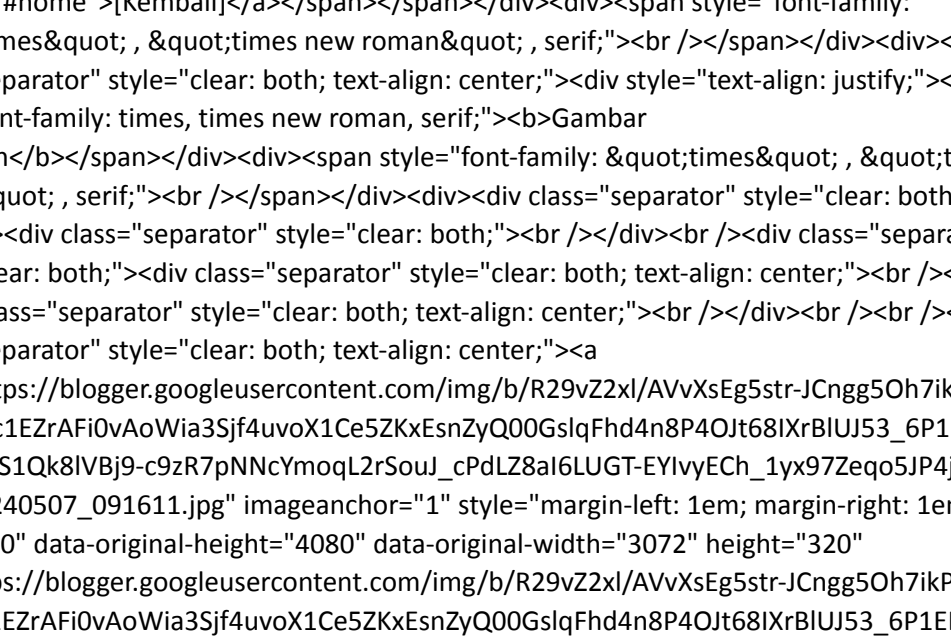
</div><div class="separator" style="clear: both; text-align: justify;">5. Jumper</div><div class="separator" style="clear: both; text-align: center;"></div>
<div class="separator" style="clear: both; text-align: justify;">
</div>
<div class="separator" style="clear: both; text-align: center;">
</div>

</div><p data-darkreader-inline-bgcolor="" data-darkreader-inline-border-bottom="" data-darkreader-inline-border-left="" data-darkreader-inline-border-right="" data-darkreader-inline-border-top="" data-darkreader-inline-color="" style="--darkreader-inline-bgcolor: #191b1c; --darkreader-inline-border-bottom: initial; --darkreader-inline-border-left: initial; --darkreader-inline-border-right: initial; --darkreader-inline-border-top: initial; --darkreader-inline-color: #c6c0b8; background-color: white; border: 0px; box-sizing: border-box; color: #444444; font-family: "Times New Roman"; margin: 0px; padding: 0px; text-align: justify;">Diagram Blok:</p><p data-darkreader-inline-bgcolor="" data-darkreader-inline-border-bottom="" data-darkreader-inline-border-left="" data-darkreader-inline-border-right="" data-darkreader-inline-border-top="" data-darkreader-inline-color="" style="--darkreader-inline-bgcolor: #191b1c; --darkreader-inline-border-bottom: initial; --darkreader-inline-border-left: initial; --darkreader-inline-border-right: initial; --darkreader-inline-border-top: initial; border: 0px; box-sizing: border-box; margin: 0px; padding: 0px; text-align: center;"> </p><div class="separator" style="clear: both; text-align: center;"></div>
<div class="separator" style="clear: both; text-align: center;">
</div>
<div class="separator" style="clear: both; text-align: center;">
</div>
<p><p>

Rangkaian Simulasi dan Prinsip Kerja

[Kembali]

Gambar rangkaian



Prinsip Kerja :

Pada percobaan ini, kita menggunakan 2 buah arduino. Arduino pertama berperan sebagai master, sedangkan arduino kedua berperan sebagai slave. Untuk inputan dip switch terhubung dengan arduino (master). Sedangkan output seven segment terhubung dengan arduino (slave).

Ketika Arduino Master menerima data input dari DIP switch, data tersebut akan dikirimkan menggunakan Arduino Uno yang bertindak sebagai Slave. Jika terdapat DIP switch yang aktif, seven segment akan menampilkan angka yang sesuai dengan data yang diterima.

Untuk percobaan ini, ketika dip switch terbesar aktif, maka seven segment akan menampilkan angka tersebut ditambah


```

style="background-color: white; outline: 0px; transition: all 0.3s ease 0s;"><div style="outline: 0px;
transition: all 0.3s ease 0s;"><span style="font-family: times, times new roman, serif;">//Master
Arduino</span></div><div style="outline: 0px; transition: all 0.3s ease 0s;"><span style="font-family:
times, times new roman, serif;">#include<SPI.h> //Library for SPI</span></div><div
style="outline: 0px; transition: all 0.3s ease 0s;"><span style="font-family: times, times new roman,
serif;">int dip[] = {2,3,4,5,6,7,8,9};</span></div><div style="outline: 0px; transition: all 0.3s ease
0s;"><span style="font-family: times, times new roman, serif;">int dipvalue[] = {};</span></div><div
style="outline: 0px; transition: all 0.3s ease 0s;"><span style="font-family: times, times new roman,
serif;">void setup ()</span></div><div style="outline: 0px; transition: all 0.3s ease 0s;"><span
style="font-family: times, times new roman, serif;">Serial.begin(9600); //Starts Serial Communication
at Baud Rate 115200</span></div><div style="outline: 0px; transition: all 0.3s ease 0s;"><span
style="font-family: times, times new roman, serif;">for(int i = 0; i < 8; i++){</span></div><div
style="outline: 0px; transition: all 0.3s ease 0s;"><span style="font-family: times, times new roman,
serif;">pinMode(dip[i], INPUT_PULLUP);</span></div><div style="outline: 0px; transition: all 0.3s
ease 0s;"><span style="font-family: times, times new roman, serif;"></span></div><div
style="outline: 0px; transition: all 0.3s ease 0s;"><span style="font-family: times, times new roman,
serif;">SPI.begin(); //Begins the SPI communication</span></div><div style="outline: 0px; transition:
all 0.3s ease 0s;"><span style="font-family: times, times new roman,
serif;">SPI.setClockDivider(SPI_CLOCK_DIV8); //Sets clock for SPI communication at 8
(16/8=2Mhz)</span></div><div style="outline: 0px; transition: all 0.3s ease 0s;"><span
style="font-family: times, times new roman, serif;">digitalWrite(SS,HIGH); // Setting SlaveSelect as
HIGH (So master doesnt connect with slave)</span></div><div style="outline: 0px; transition: all
0.3s ease 0s;"><span style="font-family: times, times new roman, serif;"></span></div><div
style="outline: 0px; transition: all 0.3s ease 0s;"><span style="font-family: times, times new roman,
serif;">void loop(void){</span></div><div style="outline: 0px; transition: all 0.3s ease 0s;"><span
style="font-family: times, times new roman, serif;">byte Mastersend;</span></div><div
style="outline: 0px; transition: all 0.3s ease 0s;"><span style="font-family: times, times new roman,
serif;">int x = 1;</span></div><div style="outline: 0px; transition: all 0.3s ease 0s;"><span
style="font-family: times, times new roman, serif;">for(int i = 0; i < 8; i++){</span></div><div
style="outline: 0px; transition: all 0.3s ease 0s;"><span style="font-family: times, times new roman,
serif;">dipvalue[i] = digitalRead(dip[i]);</span></div><div style="outline: 0px; transition: all 0.3s ease
0s;"><span style="font-family: times, times new roman, serif;">if(dipvalue[i] ==
LOW){</span></div><div style="outline: 0px; transition: all 0.3s ease 0s;"><span style="font-family:
times, times new roman, serif;">x = dip[i];</span></div><div style="outline: 0px; transition: all 0.3s
ease 0s;"><span style="font-family: times, times new roman, serif;"></span></div><div
style="outline: 0px; transition: all 0.3s ease 0s;"><span style="font-family: times, times new roman,
serif;"></span></div><div style="outline: 0px; transition: all 0.3s ease 0s;"><span style="font-family:
times, times new roman, serif;">digitalWrite(SS, LOW); //Starts communication with Slave connected
to master</span></div><div style="outline: 0px; transition: all 0.3s ease 0s;"><span
style="font-family: times, times new roman, serif;">Mastersend = x;</span></div><div
style="outline: 0px; transition: all 0.3s ease 0s;"><span style="font-family: times, times new roman,
serif;">Serial.println(Mastersend);</span></div><div style="outline: 0px; transition: all 0.3s ease
0s;"><span style="font-family: times, times new roman, serif;">SPI.transfer(Mastersend); //Send the
mastersend value to slave also receives value from slave</span></div><div style="outline: 0px;
transition: all 0.3s ease 0s;"><span style="font-family: times, times new roman,
serif;">delay(1000);</span></div><div style="outline: 0px; transition: all 0.3s ease 0s;"><span
style="font-family: times, times new roman, serif;"></span></div><div style="color: #5e5e5e;
font-family: times, &quot;times new roman&quot;, serif; font-size: 15px; outline: 0px; transition: all

```

```

0.3s ease 0s;"><br /></div></div><div style="background-color: white; outline: 0px; transition: all
0.3s ease 0s;"><div style="color: #5e5e5e; font-family: times, &quot;times new roman&quot;, serif;
font-size: 15px; outline: 0px; transition: all 0.3s ease 0s;"><br /></div><div style="color: #5e5e5e;
font-family: times, &quot;times new roman&quot;, serif; font-size: 15px; outline: 0px; transition: all
0.3s ease 0s;"><br /></div><div style="outline: 0px; transition: all 0.3s ease 0s;"><span style="color:
black; font-family: times; font-size: small;">SLAVE</span></div><div style="outline: 0px; transition:
all 0.3s ease 0s;"><span style="color: black; font-family: times; font-size: small;"><br
/></span></div><div style="outline: 0px; transition: all 0.3s ease 0s;"><div style="outline: 0px;
transition: all 0.3s ease 0s;"><div style="outline: 0px; transition: all 0.3s ease 0s;"><div
style="outline: 0px; transition: all 0.3s ease 0s;"><span style="color: black; font-family: times;
font-size: small;">//Slave Arduino:</span></div><div style="outline: 0px; transition: all 0.3s ease
0s;"><span style="color: black; font-family: times; font-size:
small;">#include<SPI.h></span></div><div style="outline: 0px; transition: all 0.3s ease
0s;"><span style="color: black; font-family: times; font-size: small;">const int segmentPins[] = {9, 8, 7,
6, 5, 4, 3, 2};</span></div><div style="outline: 0px; transition: all 0.3s ease 0s;"><span style="color:
black; font-family: times; font-size: small;">volatile boolean received = false;</span></div><div
style="outline: 0px; transition: all 0.3s ease 0s;"><span style="color: black; font-family: times;
font-size: small;">volatile byte Slaverceived;</span></div><div style="outline: 0px; transition: all
0.3s ease 0s;"><span style="color: black; font-family: times; font-size: small;">int
index;</span></div><div style="outline: 0px; transition: all 0.3s ease 0s;"><span style="color: black;
font-family: times; font-size: small;">void setup(){</span></div><div style="outline: 0px; transition:
all 0.3s ease 0s;"><span style="color: black; font-family: times; font-size:
small;">Serial.begin(9600);</span></div><div style="outline: 0px; transition: all 0.3s ease 0s;"><span
style="color: black; font-family: times; font-size: small;">for (int i = 0; i < 8; i++) {</span></div><div
style="outline: 0px; transition: all 0.3s ease 0s;"><span style="color: black; font-family: times;
font-size: small;">pinMode(segmentPins[i], OUTPUT);</span></div><div style="outline: 0px;
transition: all 0.3s ease 0s;"><span style="color: black; font-family: times; font-size:
small;">}</span></div><div style="outline: 0px; transition: all 0.3s ease 0s;"><span style="color:
black; font-family: times; font-size: small;">SPCR |= _BV(SPE); //Turn on SPI in Slave
Mode</span></div><div style="outline: 0px; transition: all 0.3s ease 0s;"><span style="color: black;
font-family: times; font-size: small;">SPI.attachInterrupt(); //Interuupt ON is set for SPI
commnucation</span></div><div style="outline: 0px; transition: all 0.3s ease 0s;"><span
style="color: black; font-family: times; font-size: small;">}</span></div><div style="outline: 0px;
transition: all 0.3s ease 0s;"><span style="color: black; font-family: times; font-size: small;">ISR
(SPI_STC_vect){ //Inerrrupt routine function</span></div><div style="outline: 0px; transition: all 0.3s
ease 0s;"><span style="color: black; font-family: times; font-size: small;">Slaverceived = SPDR; //
Value received from master if store in variable slaverceived</span></div><div style="outline: 0px;
transition: all 0.3s ease 0s;"><span style="color: black; font-family: times; font-size: small;">received
= true; //Sets received as True</span></div><div style="outline: 0px; transition: all 0.3s ease
0s;"><span style="color: black; font-family: times; font-size: small;">}</span></div><div
style="outline: 0px; transition: all 0.3s ease 0s;"><span style="color: black; font-family: times;
font-size: small;">void loop(){</span></div><div style="outline: 0px; transition: all 0.3s ease
0s;"><span style="color: black; font-family: times; font-size:
small;">Serial.println(Slaverceived);</span></div><div style="outline: 0px; transition: all 0.3s ease
0s;"><span style="color: black; font-family: times; font-size: small;">if(received){//Logic to SET LED
ON OR OFF depending upon the value recerived from master</span></div><div style="outline: 0px;
transition: all 0.3s ease 0s;"><span style="color: black; font-family: times; font-size:
small;">displayCharacter(Slaverceived);</span></div><div style="outline: 0px; transition: all 0.3s

```

```

ease 0s;"><span style="color: black; font-family: times; font-size:
small;">delay(1000);</span></div><div style="outline: 0px; transition: all 0.3s ease 0s;"><span
style="color: black; font-family: times; font-size: small;"></span></div><div style="outline: 0px;
transition: all 0.3s ease 0s;"><span style="color: black; font-family: times; font-size:
small;"></span></div><div style="outline: 0px; transition: all 0.3s ease 0s;"><span style="color:
black; font-family: times; font-size: small;">void displayCharacter(int ch) {</span></div><div
style="outline: 0px; transition: all 0.3s ease 0s;"><span style="color: black; font-family: times;
font-size: small;">byte patterns[10][7] = {</span></div><div style="outline: 0px; transition: all 0.3s
ease 0s;"><span style="color: black; font-family: times; font-size: small;">{0, 0, 0, 0, 0, 0, 1}, //
0</span></div><div style="outline: 0px; transition: all 0.3s ease 0s;"><span style="color: black;
font-family: times; font-size: small;">{1, 0, 0, 1, 1, 1, 1}, // 1</span></div><div style="outline: 0px;
transition: all 0.3s ease 0s;"><span style="color: black; font-family: times; font-size: small;">{0, 0, 1,
0, 0, 1, 0}, // 2</span></div><div style="outline: 0px; transition: all 0.3s ease 0s;"><span style="color:
black; font-family: times; font-size: small;">{0, 0, 0, 0, 1, 1, 0}, // 3</span></div><div style="outline:
0px; transition: all 0.3s ease 0s;"><span style="color: black; font-family: times; font-size: small;">{1, 0,
0, 1, 1, 0, 0}, // 4</span></div><div style="outline: 0px; transition: all 0.3s ease 0s;"><span
style="color: black; font-family: times; font-size: small;">{0, 1, 0, 0, 1, 0, 0}, // 5</span></div><div
style="outline: 0px; transition: all 0.3s ease 0s;"><span style="color: black; font-family: times;
font-size: small;">{0, 1, 0, 0, 0, 0, 0}, // 6</span></div><div style="outline: 0px; transition: all 0.3s
ease 0s;"><span style="color: black; font-family: times; font-size: small;">{0, 0, 0, 1, 1, 1, 1}, //
7</span></div><div style="outline: 0px; transition: all 0.3s ease 0s;"><span style="color: black;
font-family: times; font-size: small;">{0, 0, 0, 0, 0, 0, 0}, // 8</span></div><div style="outline: 0px;
transition: all 0.3s ease 0s;"><span style="color: black; font-family: times; font-size: small;">{0, 0, 0,
0, 1, 0, 0} // 9</span></div><div style="outline: 0px; transition: all 0.3s ease 0s;"><span style="color:
black; font-family: times; font-size: small;"></span></div><div style="outline: 0px; transition: all
0.3s ease 0s;"><span style="color: black; font-family: times; font-size: small;">if ((ch >= 0
&amp;&amp; ch <= 9)) {</span></div><div style="outline: 0px; transition: all 0.3s ease 0s;"><span
style="color: black; font-family: times; font-size: small;">// Get the digit index (0-9) from the
character</span></div><div style="outline: 0px; transition: all 0.3s ease 0s;"><span style="color:
black; font-family: times; font-size: small;">int index = ch;</span></div><div style="outline: 0px;
transition: all 0.3s ease 0s;"><span style="color: black; font-family: times; font-size: small;">// Write
the pattern to the segment pins</span></div><div style="outline: 0px; transition: all 0.3s ease
0s;"><span style="color: black; font-family: times; font-size: small;">for (int i = 0; i <= 7; i++)
{</span></div><div style="outline: 0px; transition: all 0.3s ease 0s;"><span style="color: black;
font-family: times; font-size: small;">digitalWrite(segmentPins[i],
patterns[index][i]);</span></div><div style="outline: 0px; transition: all 0.3s ease 0s;"><span
style="color: black; font-family: times; font-size: small;"></span></div><div style="outline: 0px;
transition: all 0.3s ease 0s;"><span style="color: black; font-family: times; font-size:
small;"></span></div><div style="color: #5e5e5e; font-family: times, &quot;times new
roman&quot;, serif; font-size: 15px; outline: 0px; transition: all 0.3s ease
0s;"></div></div></div></div></div><p style="text-align: left;"><span style="font-family:
times, times new roman, serif;"></span></p><div style="text-align: left;"><b style="font-family:
times, &quot;times new roman&quot;, serif;">5. Kondisi </b><span style="font-family: times,
&quot;times new roman&quot;, serif;">
</span><a name="Kondisi" style="font-family: times, &quot;times new roman&quot;,
serif;"></a><span style="font-family: times, &quot;times new roman&quot;, serif;">

```

[\[Kembali\]](#home)

[illegible]