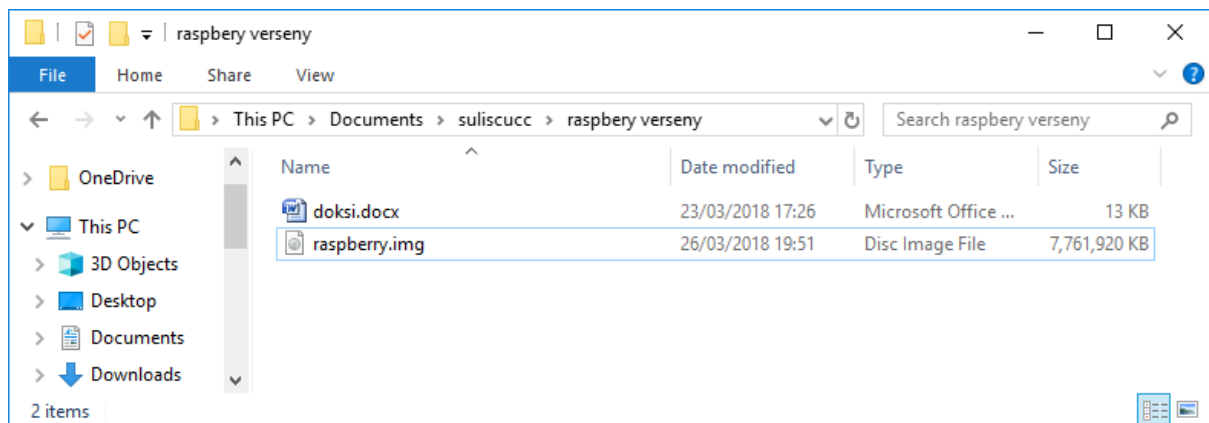
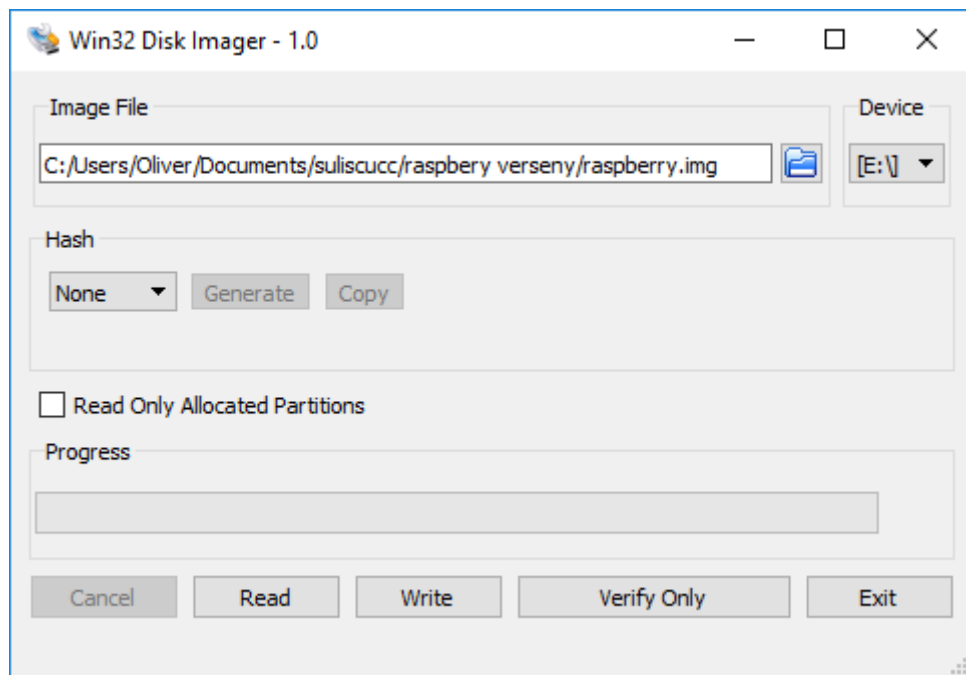


Terepasztal vezérlés Raspberry Pi és Arduino segítségével

Mielőtt belekezdenék bármibe érdemes biztonsági mentést csinálni a rendszerről: Win32 Disk Imager programot használtam erre.



SSH föl van, telepítve az nem kell.

User: pi

Password: raspberry

<https://www.youtube.com/watch?v=ygVxs7FSCeo>

```
apt-get install mcedit
```

apache2 php-t föl kell telepíteni.

```
apt-get install apache2
```

```
-y
```

```
sudo chown -R pi:www-data /var/www/html/
```

```
sudo chmod -R 777 /var/www/html/
```

Miután föltelepült:

Hostname – I

Ide majd az IP címet fogja kiírni, amivel a web szerveret, működtetem.

A fájlokat itt érem el: var/www/html/index.html

PHP könyvtár telepítése.

```
sudo apt-get install php libapache2-mod-php -y
```

```
mcedit /var/www/html/index.php
```

```
chown pi:www-data index.php
```

```
chmod 777 var/www/html/index.php
```

A fájlokat itt érem el: var/www/html/index.php

```
apt-get install mcedit
```



```

var imageName = imageDir[ imageDir.Length - 1 ];

if(imageName == "kockavball23.jpg"){
    image1.src = "kockavball1.jpg";
    var xmlHttp = new XMLHttpRequest();
    xmlHttp.open( "GET", "script0.php?set=1", true );
    xmlHttp.send( null );
    var result = xmlHttp.responseText;
}

else if (imageName == "kockavball1.jpg"){
    image1.src = "kockavball23.jpg";
    var xmlHttp = new XMLHttpRequest();
    xmlHttp.open( "GET", "script1.php?set=1", true ); // false for
synchronous request
    xmlHttp.send( null );
    var result = xmlHttp.responseText;
}
}

```

A felület nagy részét táblázattal oldottam meg.

```

<tr>
<td style="padding: 0 0 0 0;" >

</td>

<td style="padding: 0 0 0 0;" >

</td>

<td style="padding: 0 0 0 0;" >

```

```

</td>
```

```
<td style="padding: 0 0 0 0;" >

</td>
```

```
<td style="padding: 0 0 0 0;" >

</td>
```

```
<td style="padding: 0 0 0 0;" >

</td>
```

```
<td style="padding: 0 0 0 0;" >

</td>
```

```
<td style="padding: 0 0 0 0;" >

</td>
```

```
<td style="padding: 0 0 0 0;" >

</td>
```

```

<td style="padding: 0 0 0 0;" >

</td>

```

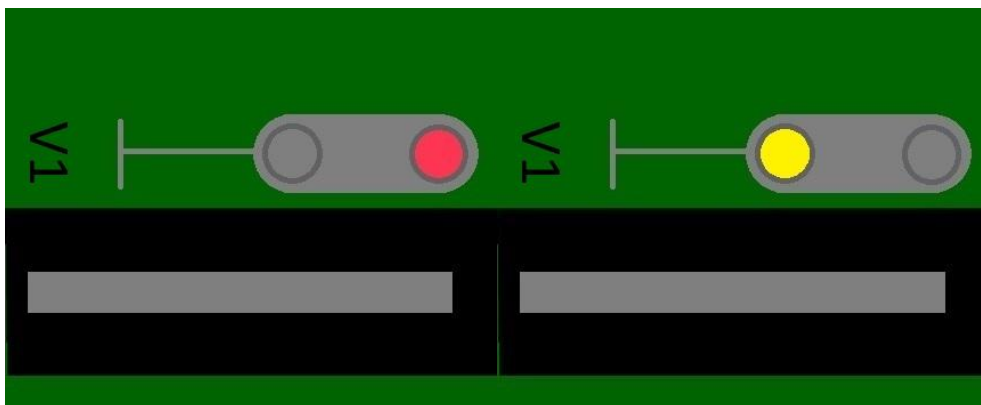
```

<td style="padding: 0 0 0 0;" >

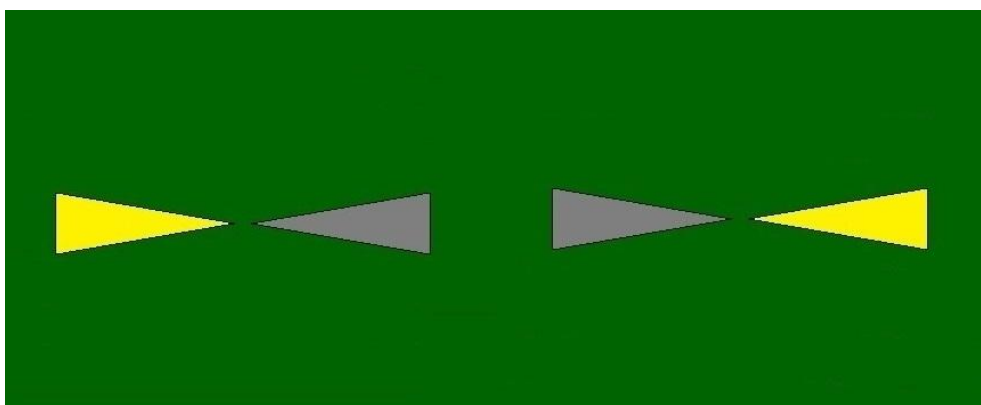
</td>

```

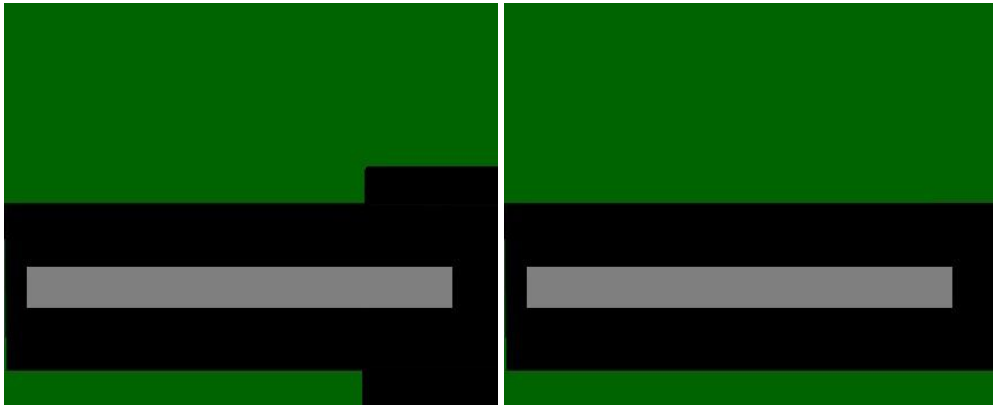
A táblázatban szereplő képeket paintba szerkesztettem.



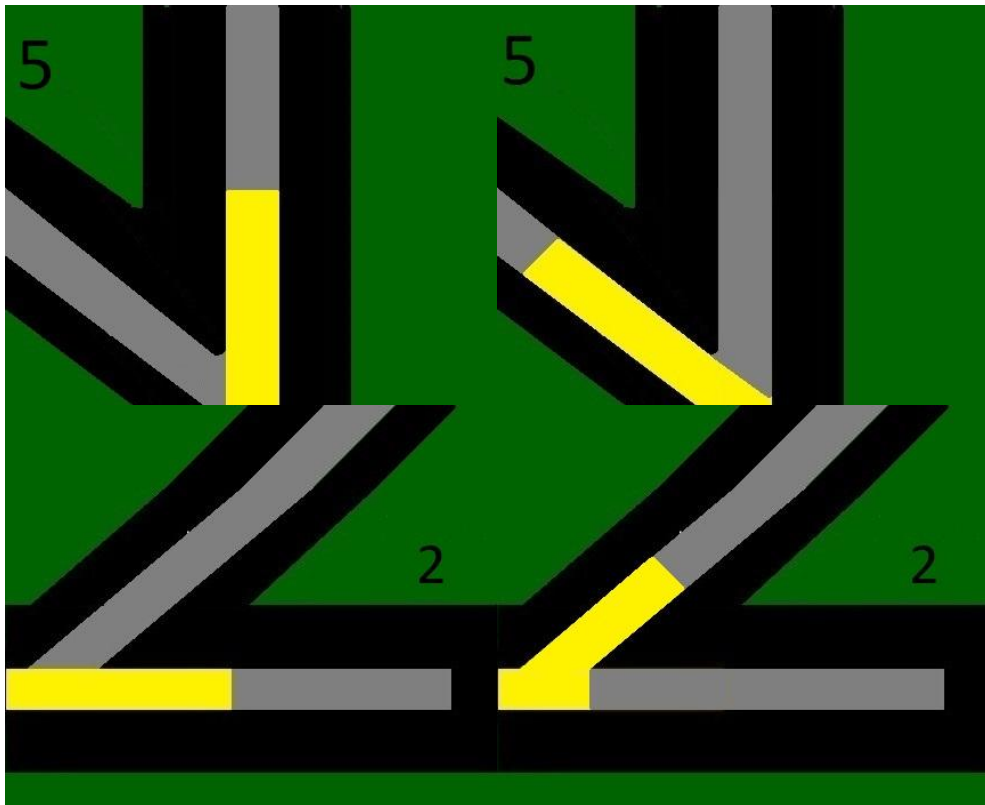
Tilos jelző, szabad jelző



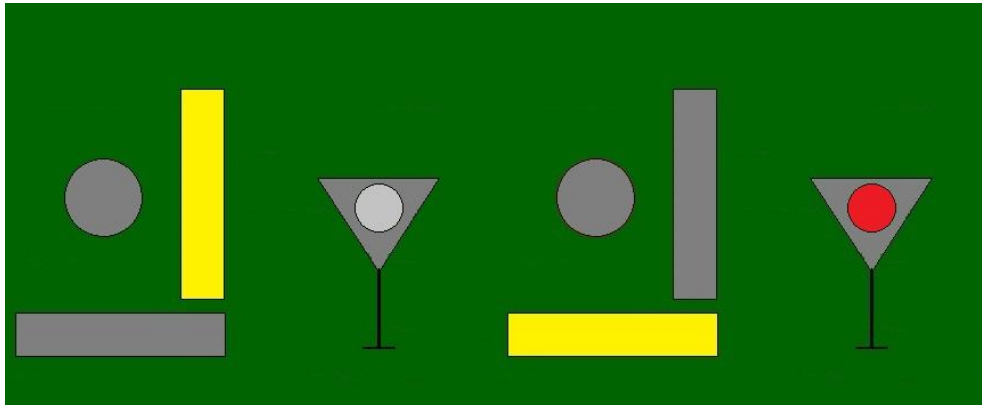
Menetirány jelölése



Vakvágány, vágány



Váltók egyenes és kitérő állása.



Sorompó zárt és nyitott állapot.

A **script.php** fájlok tartalma mely futtatja, a Python scriptet a Python script működteti a vezérlő elektronikát.

```
<?php
print_r($_GET);
if (isset($_GET["set"])){
if ($_GET["set"] == "1"){
echo "running..";
exec("/var/www/html/terepasztal/LED0.py");}}
?>
```

Python script tartalma:

Az utolsó sorban található karaktert küldi ki az Arduinonak USB porton keresztül.

```
#!/usr/bin/env python
import time
import serial
ser = serial.Serial('/dev/ttyACM0', 9600)
ser.write('0')
```

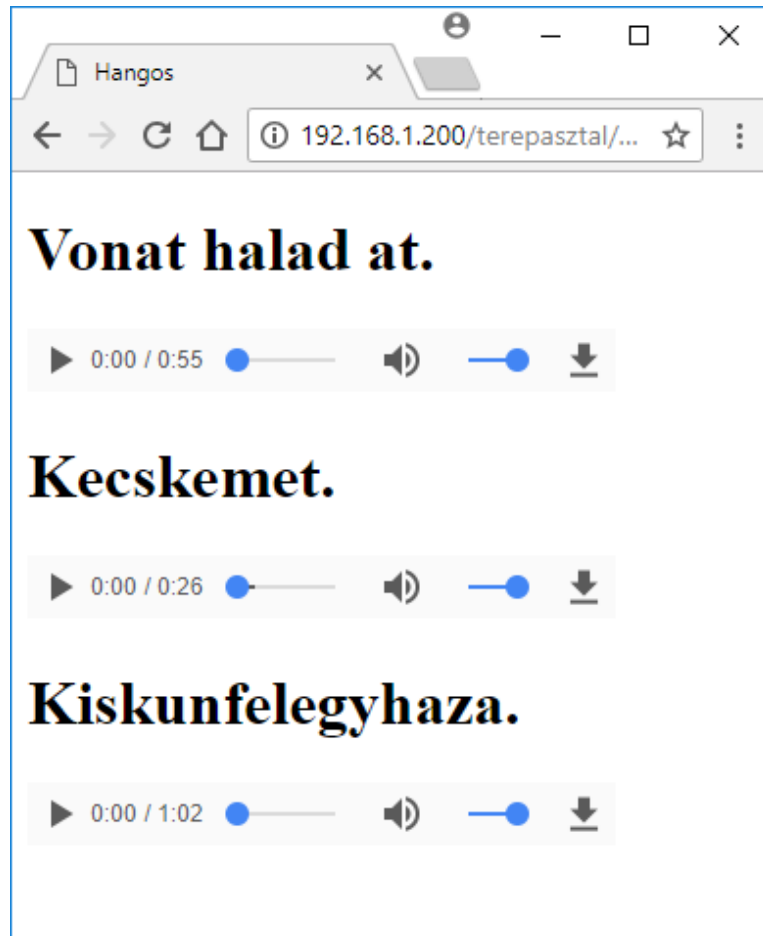
Arduino kód:

A LED-eket vezérli a LED-ek helyére:

```
if(szam == '0'){
    digitalWrite(13, HIGH);
    delay(1000);
    digitalWrite(13, LOW);
}
```

Elérés: <http://192.168.X.X/terepasztal/terepasztal.php>

A hangos utas tájékoztatót működtető html file a hangos.html amely a **Kecskemet.wav**, **Hangos765Fh.wav**, **Hangos1.wav** fájlokat játssza le.



```
<h1>Kiskunfelegyhaza.</h1>
<audio controls>
  <source src="Hangos765Fh.ogg" type="audio/ogg">
  <source src="Hangos765Fh.wav" type="audio/mpeg">
</audio>
```

Elérés: <http://192.168.X.X/terepasztal/hangos.html>