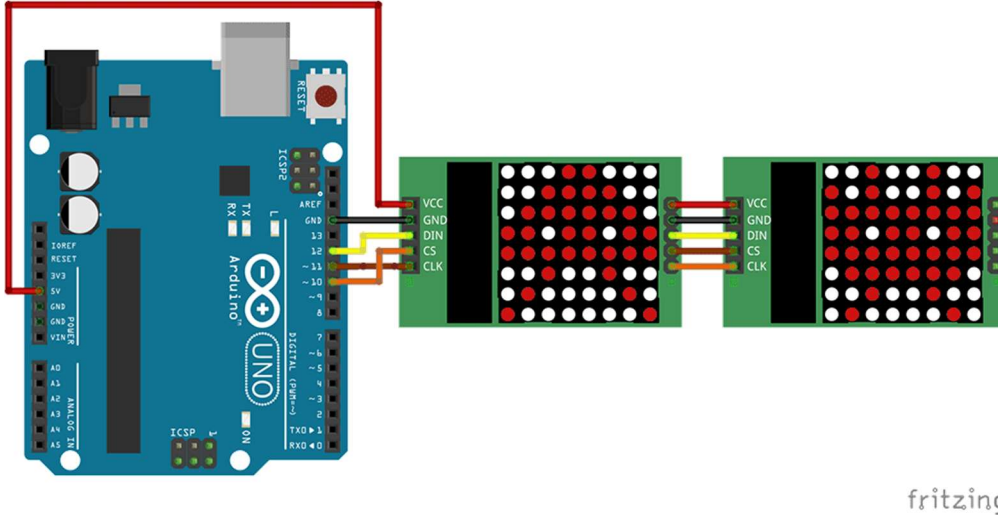
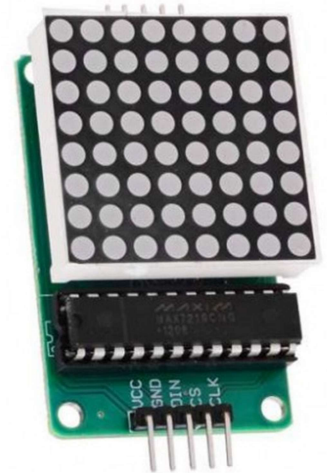


**Beschreibung:**

Diese Variante des MAX7219 erlaubt dem Benutzer jede Variante der Datenauswahl für die Darstellung zu nutzen. Generell können jedoch mit der 4-Draht-Schnittstelle alle möglichen Mikroprozessoren verbunden werden. Die 3 IO-Variante gewährleistet eine flimmerfreie Darstellung. Eine Kaskadierung dieser Module ist ebenfalls ohne Probleme möglich, wobei Ausgang der ersten Steuerung mit dem Eingang der zweiten Steuerung verbunden werden müssen usw.

**Beschaltung:**

fritzing

**Bild:****Sketch: (ein Display)**

```
#include "LedControl.h"
```

```
LedControl lc=LedControl(12,11,10,1); // Erzeugt Objekt ;Anschlußpins und Anzahl der Displays
```

```
unsigned long delayTime=200; // Verzögerung zwischen den Frames
```

```
byte invader1a[] = // schreibt Werte in die Arrays
```

```
{
  B11110000, B11110000, B11110000, B11110000, B11110000, B11110000, B11110000, B11110000,
};
```

```
byte invader2a[] =
```

```
{
  B00001111, B00001111, B00001111, B00001111, B00001111, B00001111, B00001111, B00001111,
};
```

```
void setup()
```

```
{
  lc.shutdown(0,false); // weckt das Display
  lc.setIntensity(0,1); // setzt das intensitäts Level
  lc.clearDisplay(0); // lösche Display
}
```

```
void sinvader1a()
```

```
{ for (int i = 0; i < 8; i++)
  { lc.setRow(0,i,invader1a[i]);
  }}
```

```
void sinvader2a()
```

```
{ for (int i = 0; i < 8; i++)
  { lc.setRow(0,i,invader2a[i]);
  }}
```

```
void loop()
```

```
{
  // Put #1 frame on both Display
  sinvader1a();
  delay(delayTime);
  sinvader2a();
  delay(delayTime);
}
```