

Enseignes et afficheurs à LED

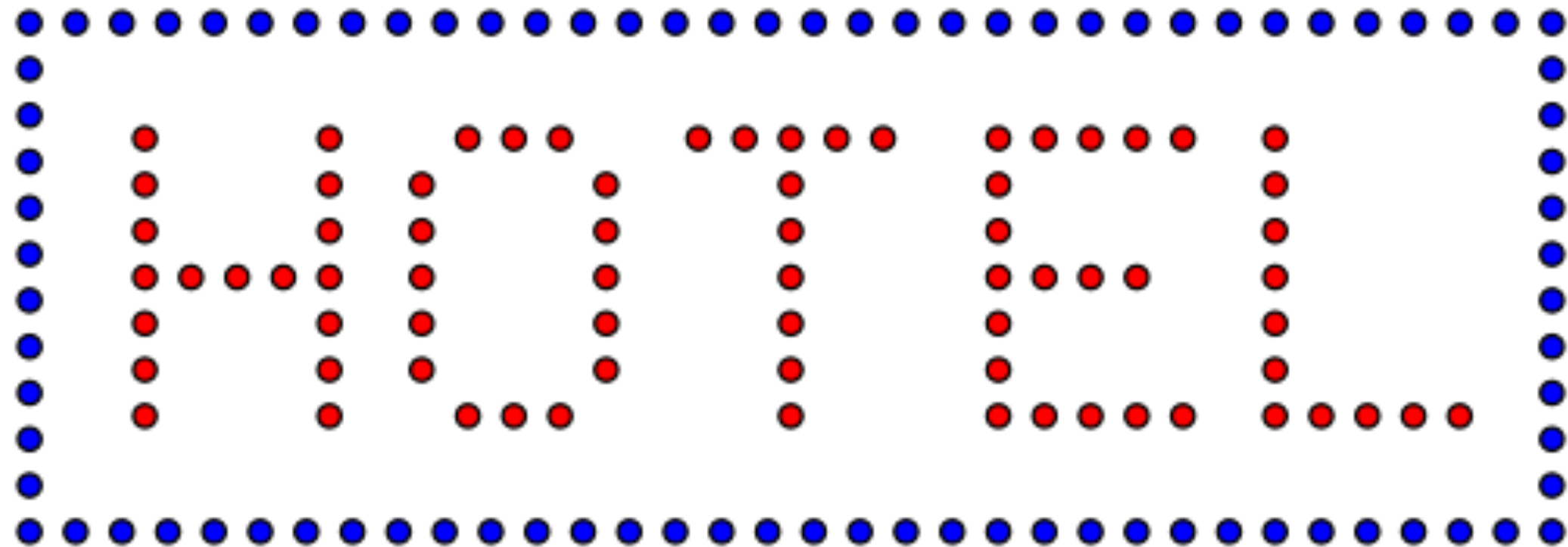
# Enseignes à motifs fixes

# Enseignes à motifs fixes

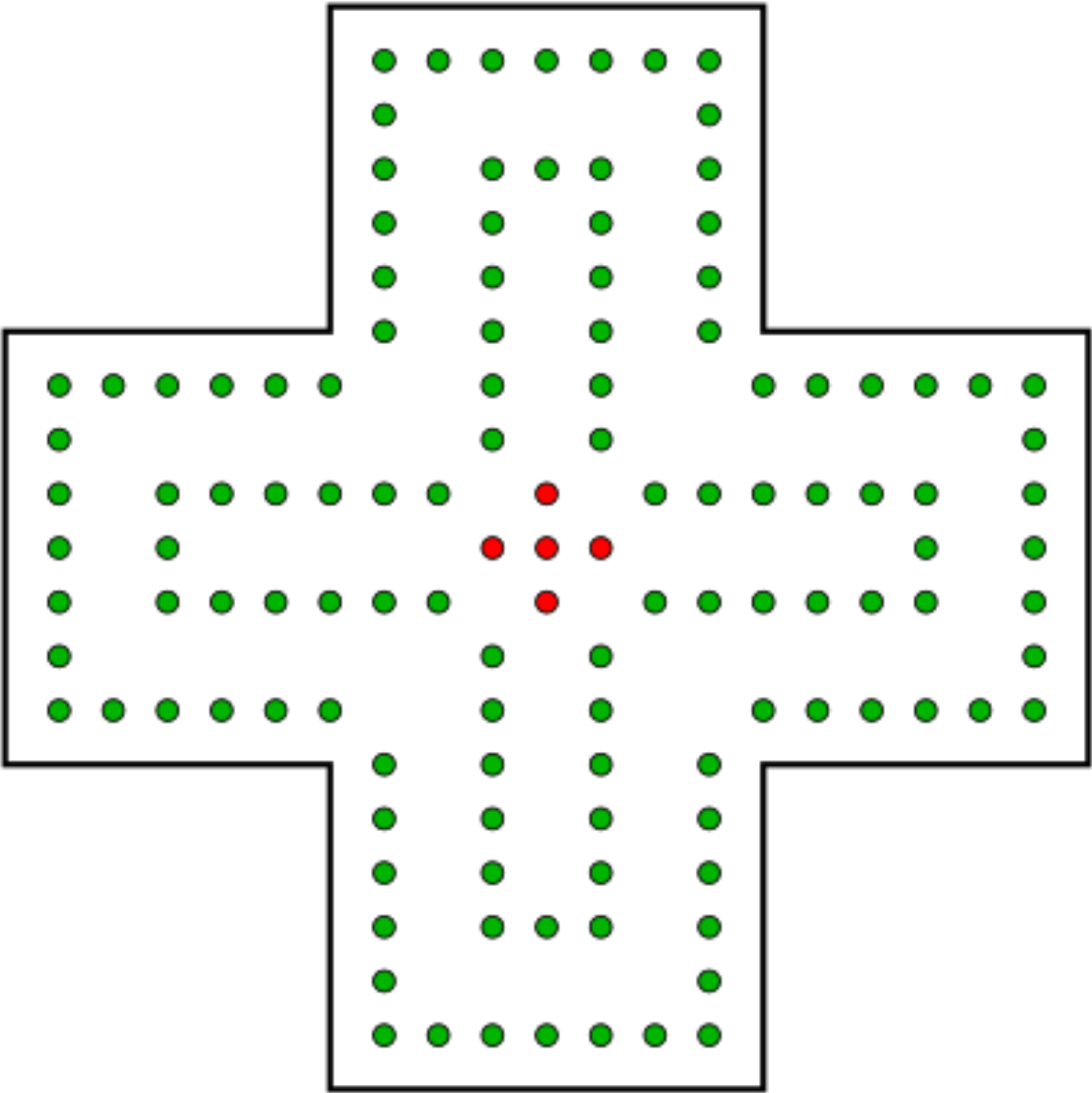
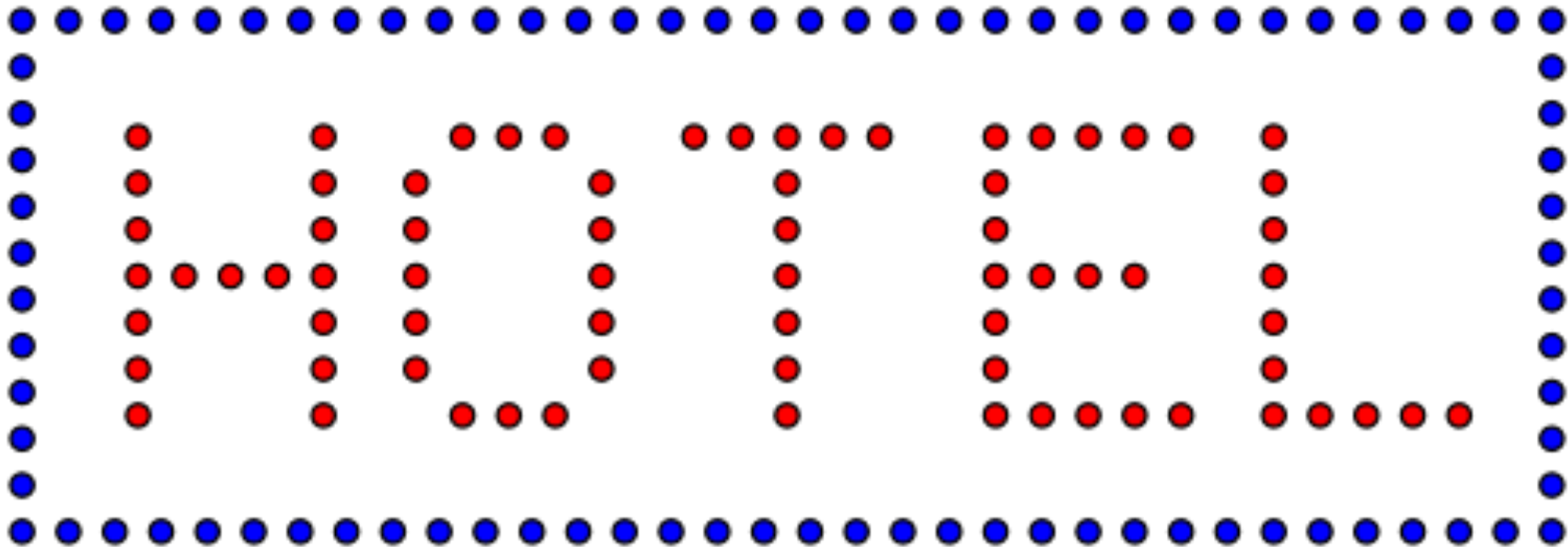
**Pierre-Yves Rochat**

- Principe des enseignes à motifs fixes
- Découpage en segments
- Schémas de commande à transistor
- Montages série et parallèle
- Programmation des animations

# Enseignes à motifs fixes

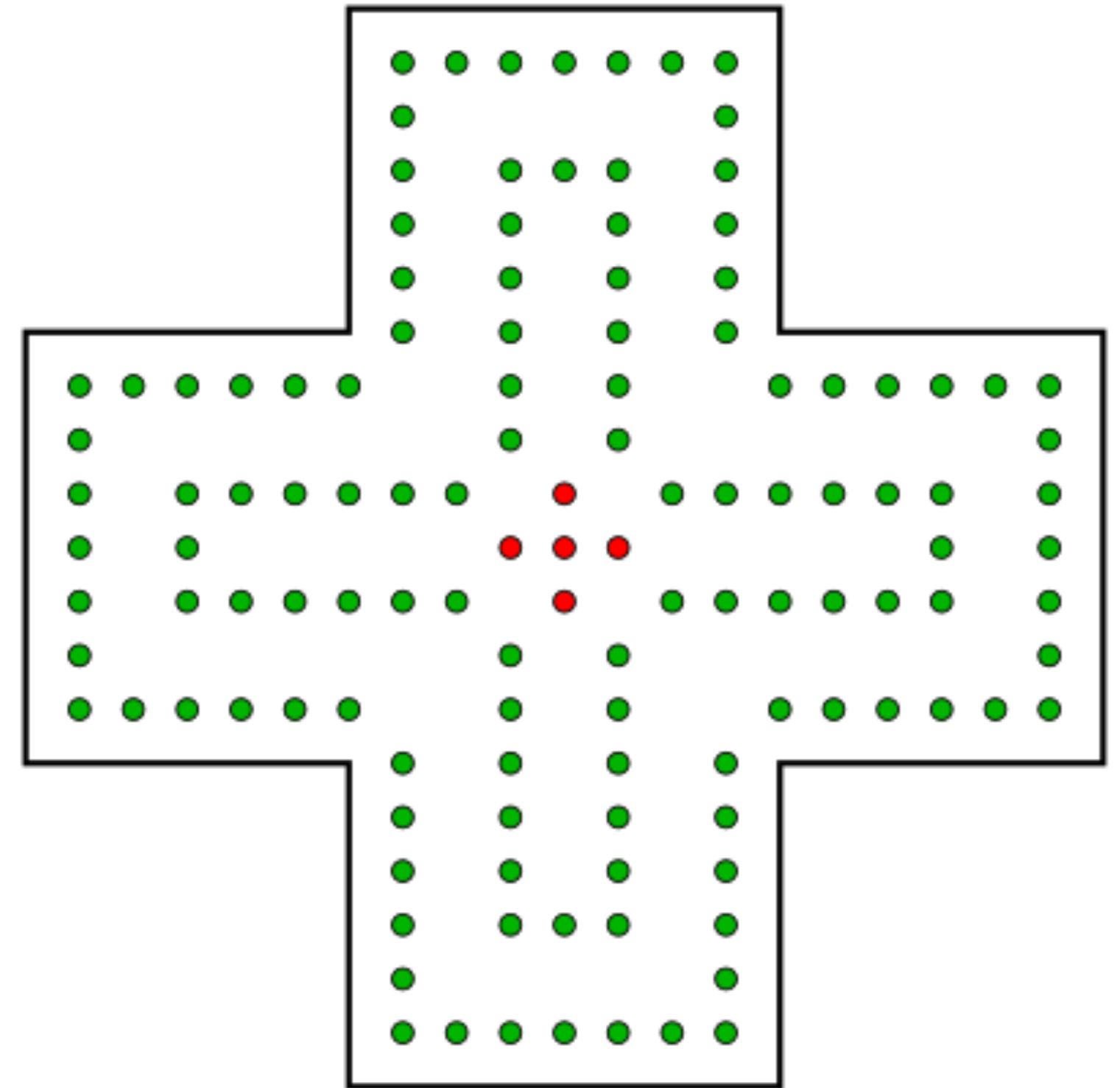
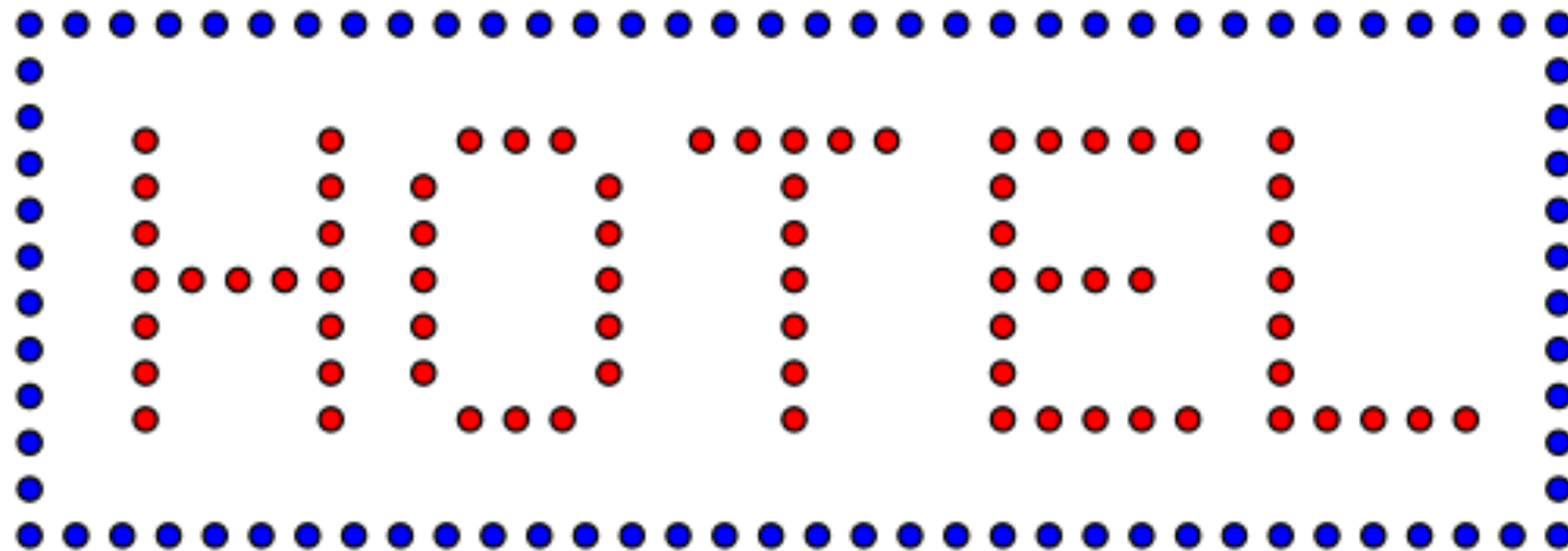


# Enseignes à motifs fixes



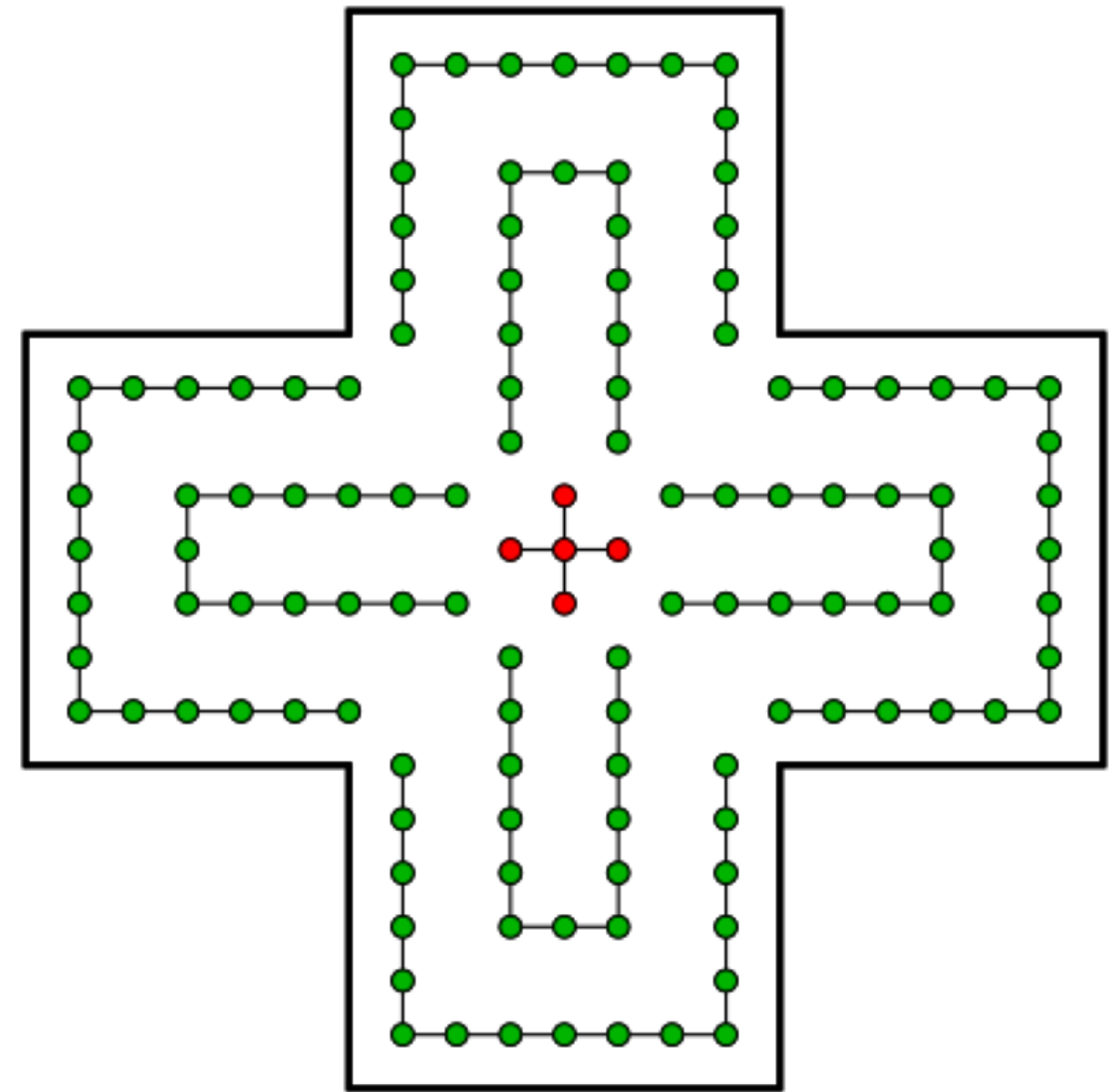
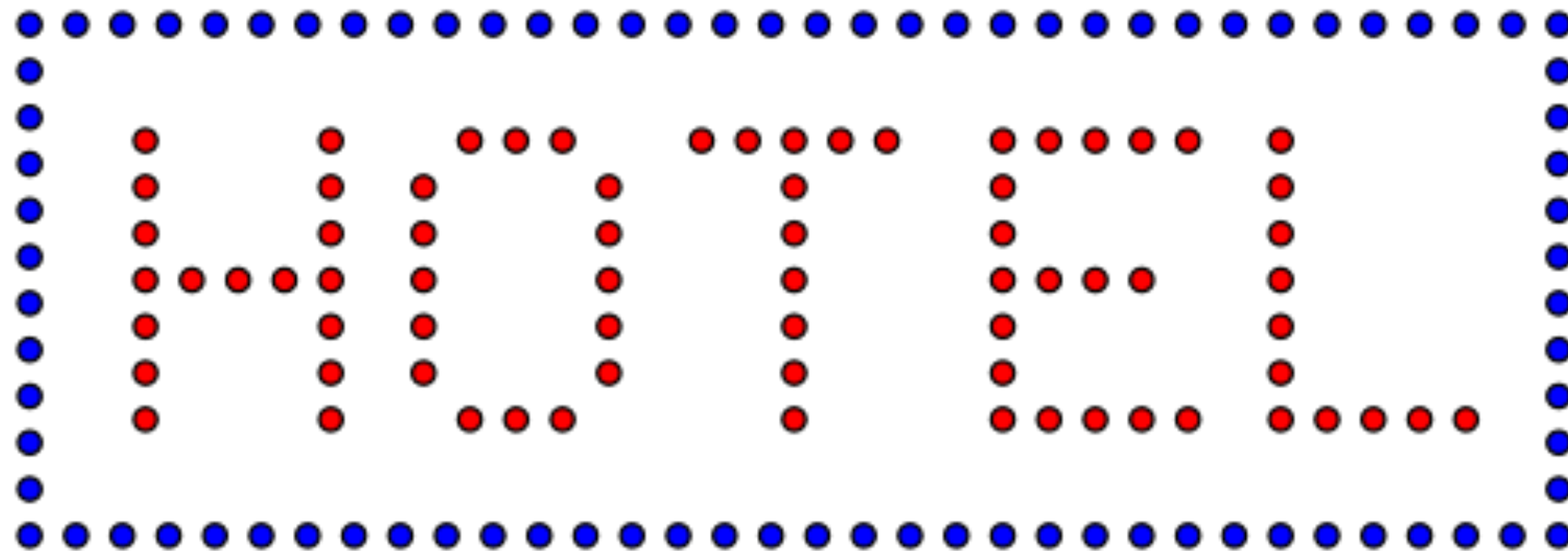
# Enseignes à motifs fixes

Regrouper les LED en motifs.



# Enseignes à motifs fixes

Regrouper les LED en motifs.



# Commande des LED

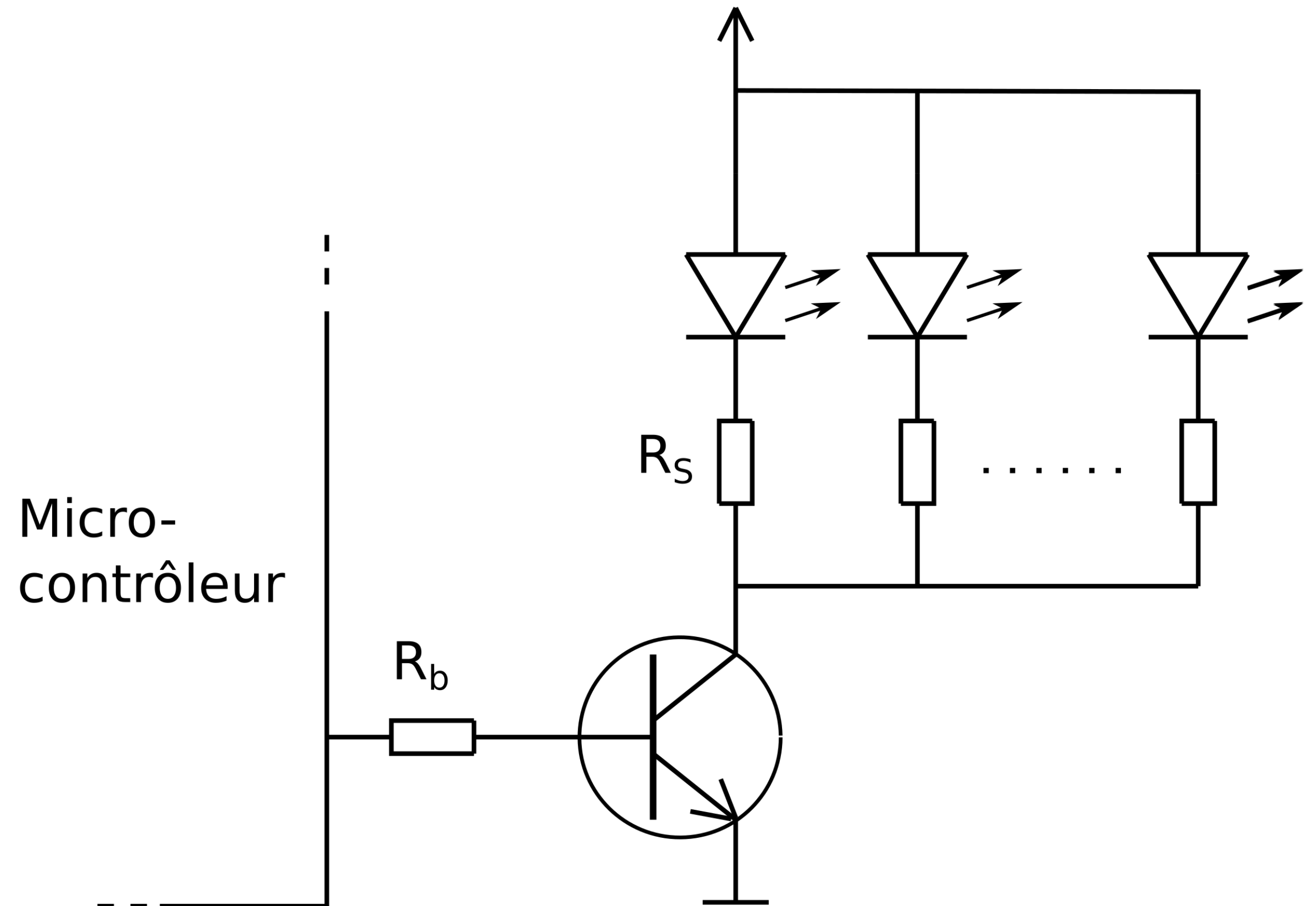
- Une LED consomme 10 mA



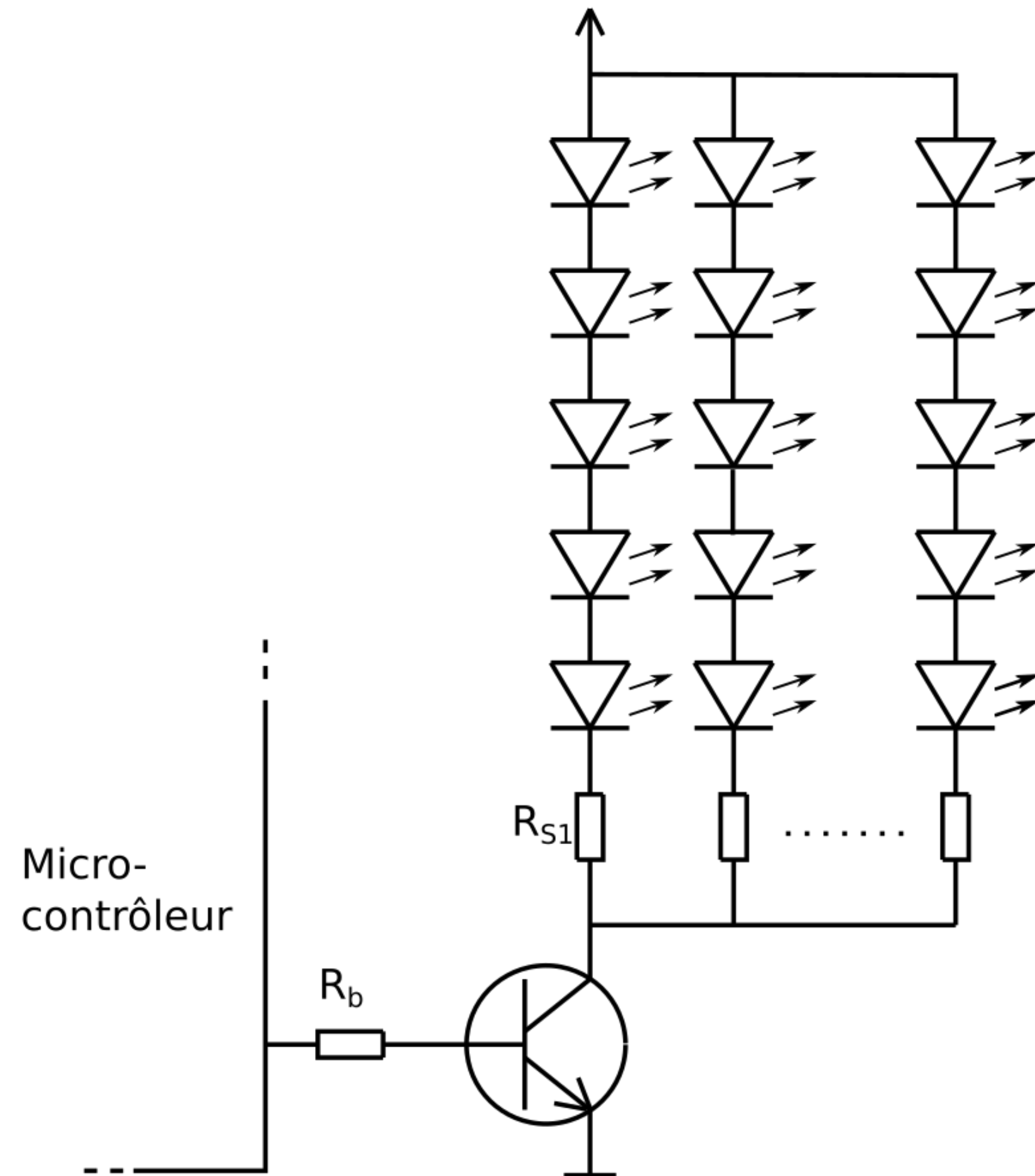
- Une LED consomme 10 mA
- Un microcontrôleur peut fournir environ 10 à 20 mA

# Commande des LED

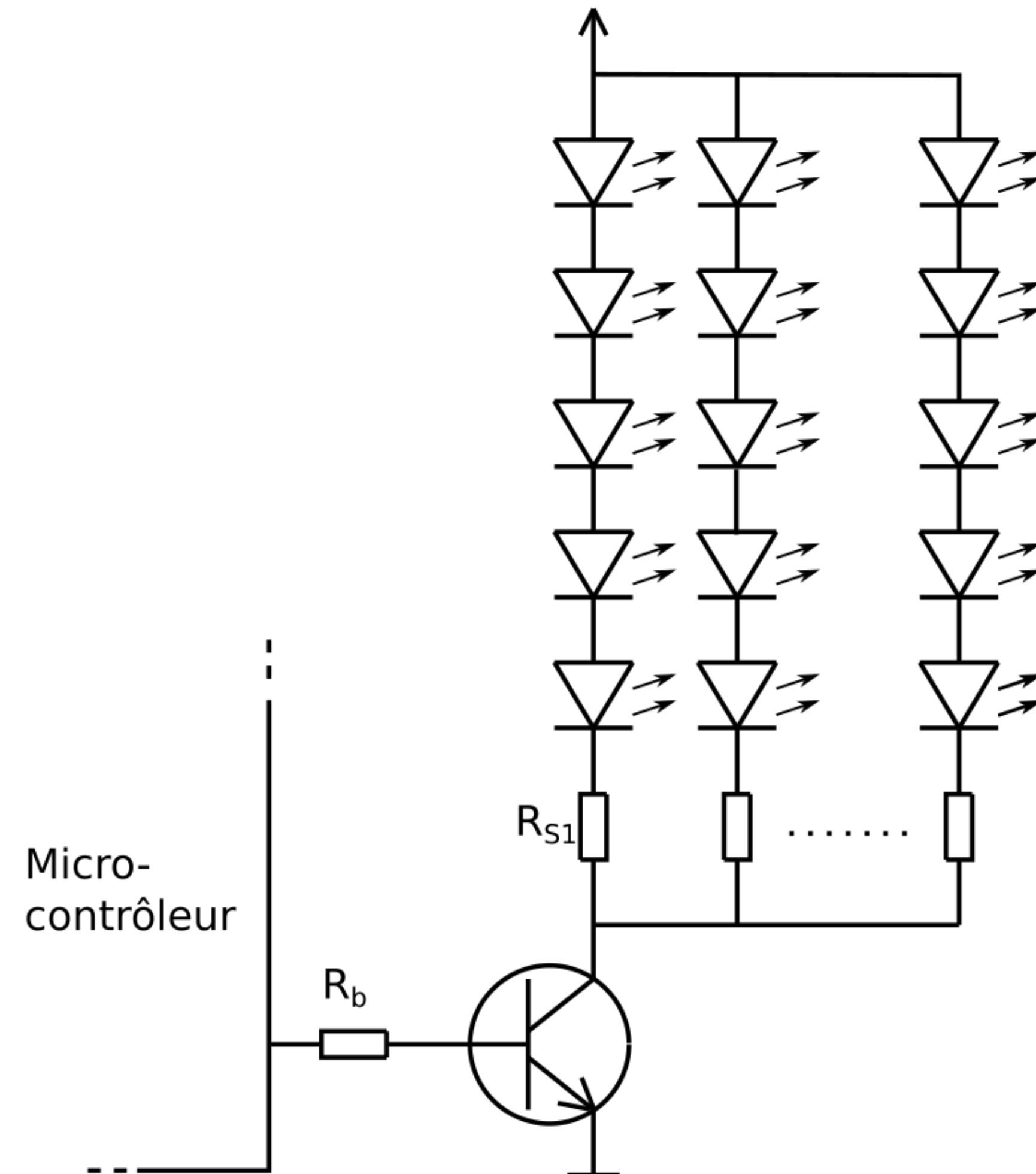
- Une LED consomme 10 mA
- Un microcontrôleur peut fournir environ 10 à 20 mA
- Utilisation d'un transistor



# Montage en série

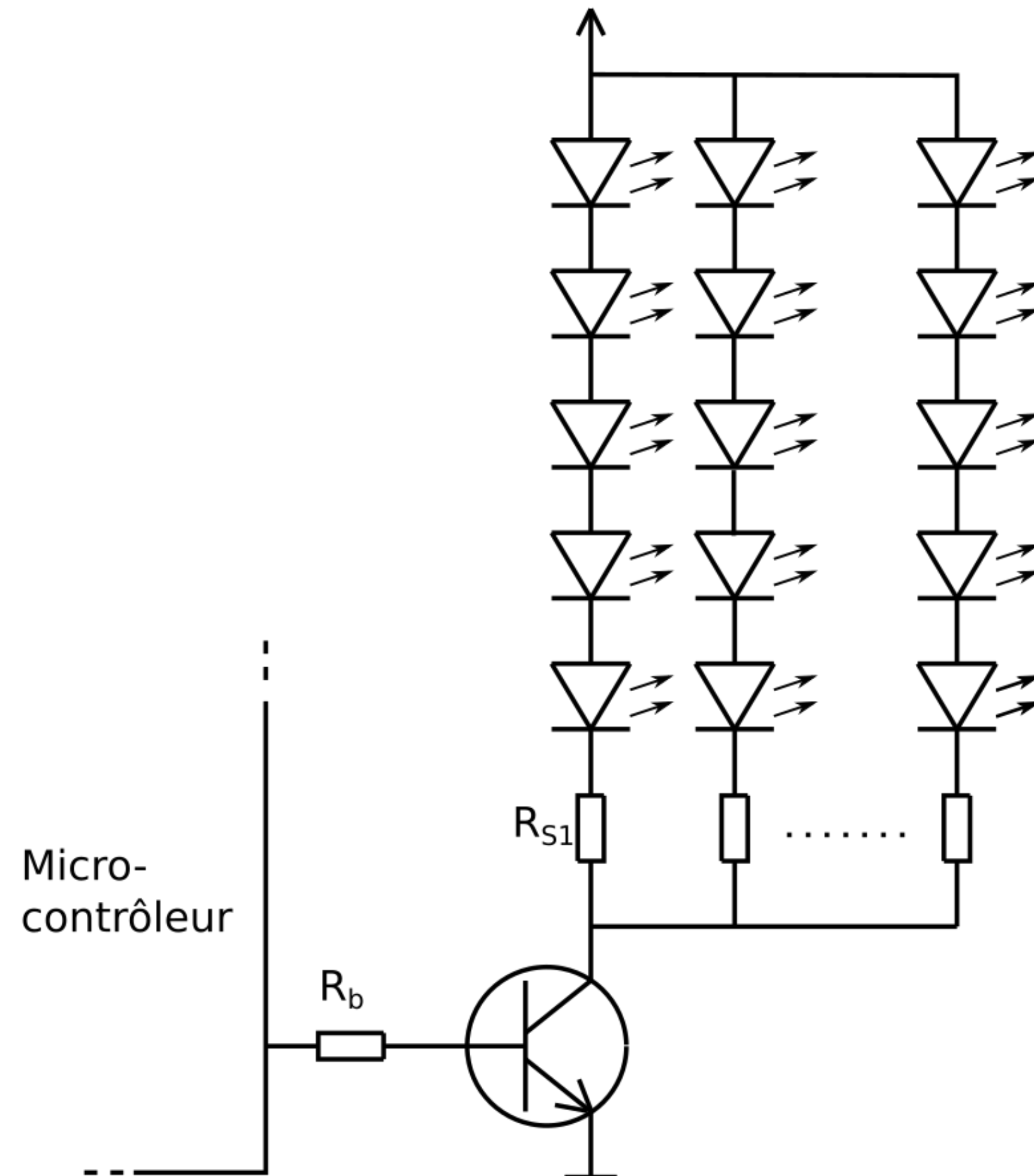


# Montage en série



BC337 : 500 mA

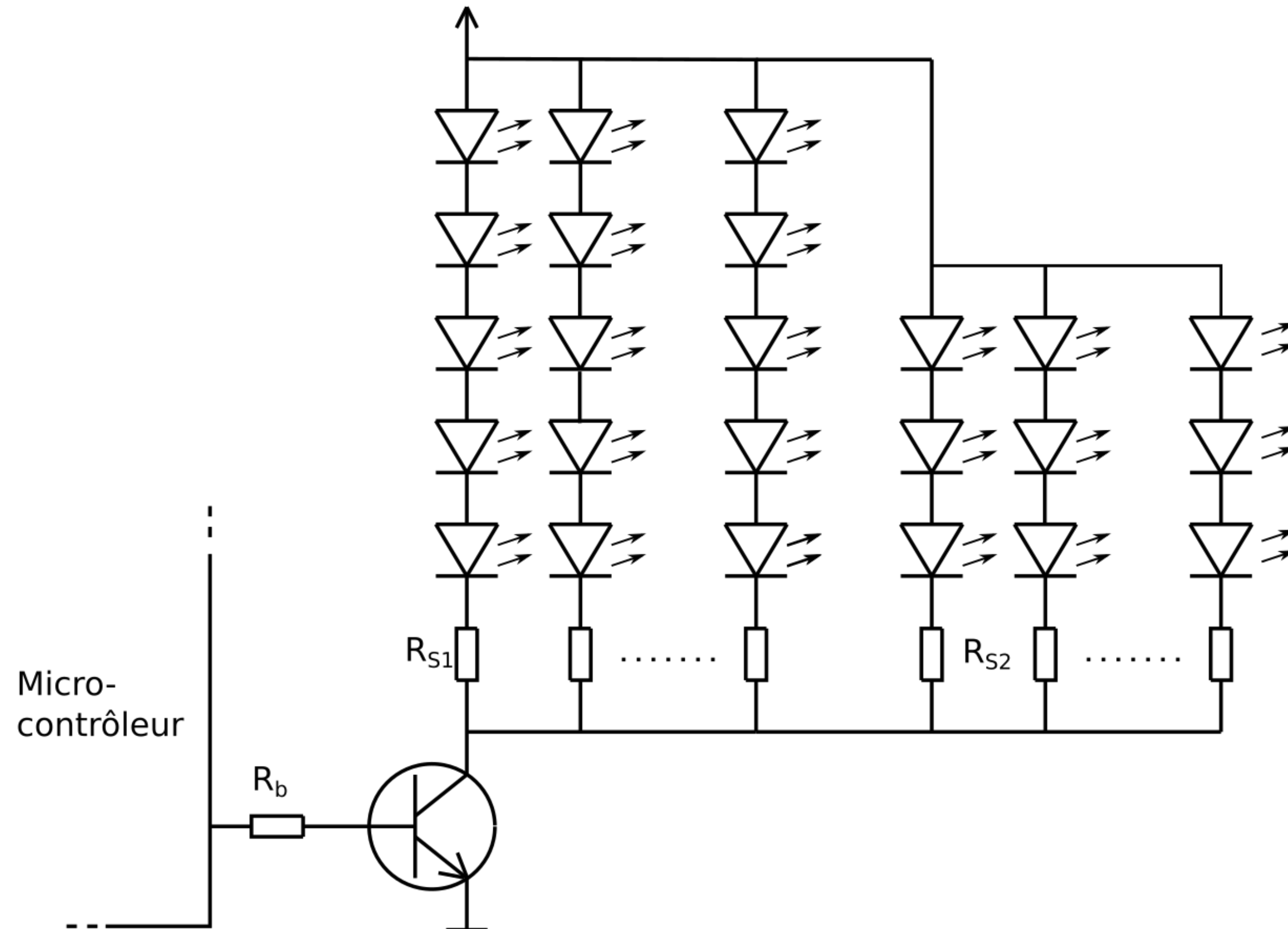
# Montage en série



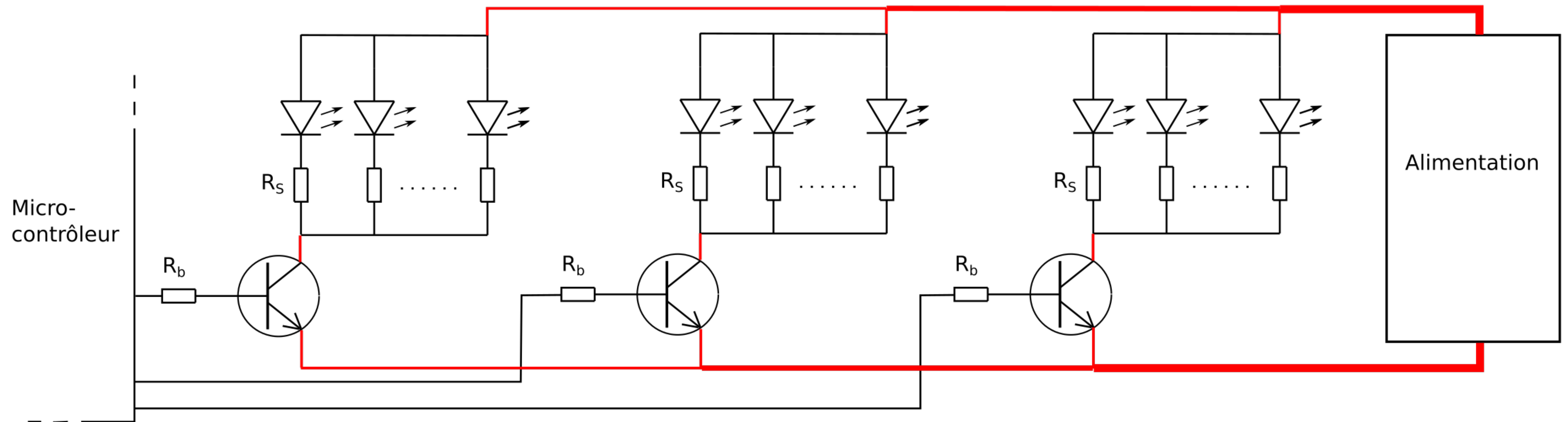
BC337 : 500 mA

50 × 5 LED

# Montage en série



# Courant importants



# Programmation des animations

```
1 void setup() {
2   P2DIR |= 0xFF; // P2.0 à P2.7 en sortie
3 }
4
5 void loop() { // Boucle infinie, correspond à toute l'animation
6   P2OUT = 0; // éteint toutes les LED
7   P2OUT |= (1<<0); delay(200); // allume la première LED
8   P2OUT |= (1<<1); delay(200); // allume successivement chaque LED...
9   P2OUT |= (1<<2); delay(200);
10  P2OUT |= (1<<3); delay(200);
11  P2OUT |= (1<<4); delay(200);
12  P2OUT |= (1<<5); delay(200);
13  P2OUT |= (1<<6); delay(200);
14  P2OUT |= (1<<7); delay(1000); // attend un peu après la dernière LED
15  P2OUT = 0; delay(500); // éteint toutes les LED pendant 1/2 seconde
16  P2OUT = 0xFF; delay(500); // allume toutes les LED
17  P2OUT = 0; delay(500); // répète...
18  P2OUT = 0xFF; delay(1000);
19  P2OUT = 0; delay(500); // pause avant la reprise de l'animation
20 }
```



# Programmation des animations

```
1 void ChenillardAjoute(uint16_t attente) {
2     uint16_t i;
3     for (i=0; i<8; i++) {
4         P2OUT |= (1<<i);
5         delay(attente);
6     }
7 }
8 ...
9 void loop() { // Boucle infinie, correspond à toute l'animation
10     ChenillardAjoute(200);
11     delay(800); // attend un peu après la dernière LED
12     P2OUT = 0; delay(500); // éteint toutes les LED pendant 1/2 seconde
13     ...
14 }
```

# Programmation des animations

- Les programmes peuvent devenir très longs, peu lisibles

- Les programmes peuvent devenir très longs, peu lisibles  
... et dépasser même la taille mémoire du microcontrôleur

- Les programmes peuvent devenir très longs, peu lisibles  
... et dépasser même la taille mémoire du microcontrôleur
- Langages spécifiques, qu'il faudra interpréter

- Les programmes peuvent devenir très longs, peu lisibles  
... et dépasser même la taille mémoire du microcontrôleur
- Langages spécifiques, qu'il faudra interpréter
- Intensité variable des LED : PWM, BCM

- Principe des enseignes à motifs fixes
- Découpage en segments
- Schémas de commande à transistor
- Montages série et parallèle
- Programmation des animations