

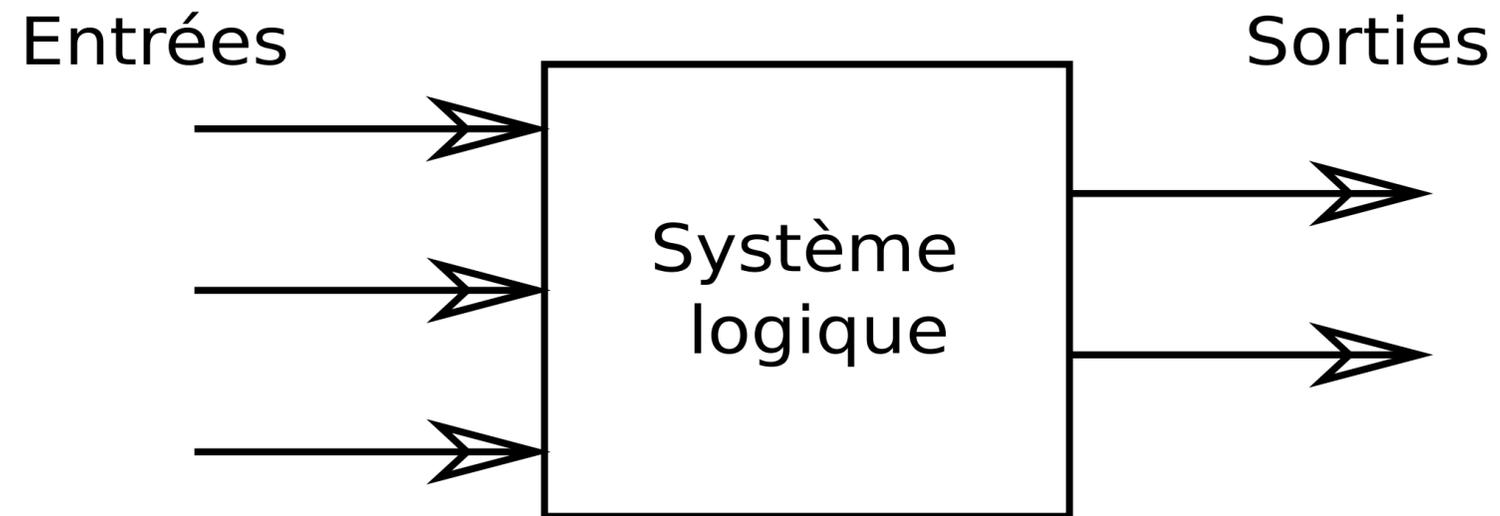
# **Systemes séquentiels**

## **Comprendre les Microcontrôleurs**

Jean-Daniel NICOUD et Pierre-Yves ROCHAT

## Systemes séquentiels

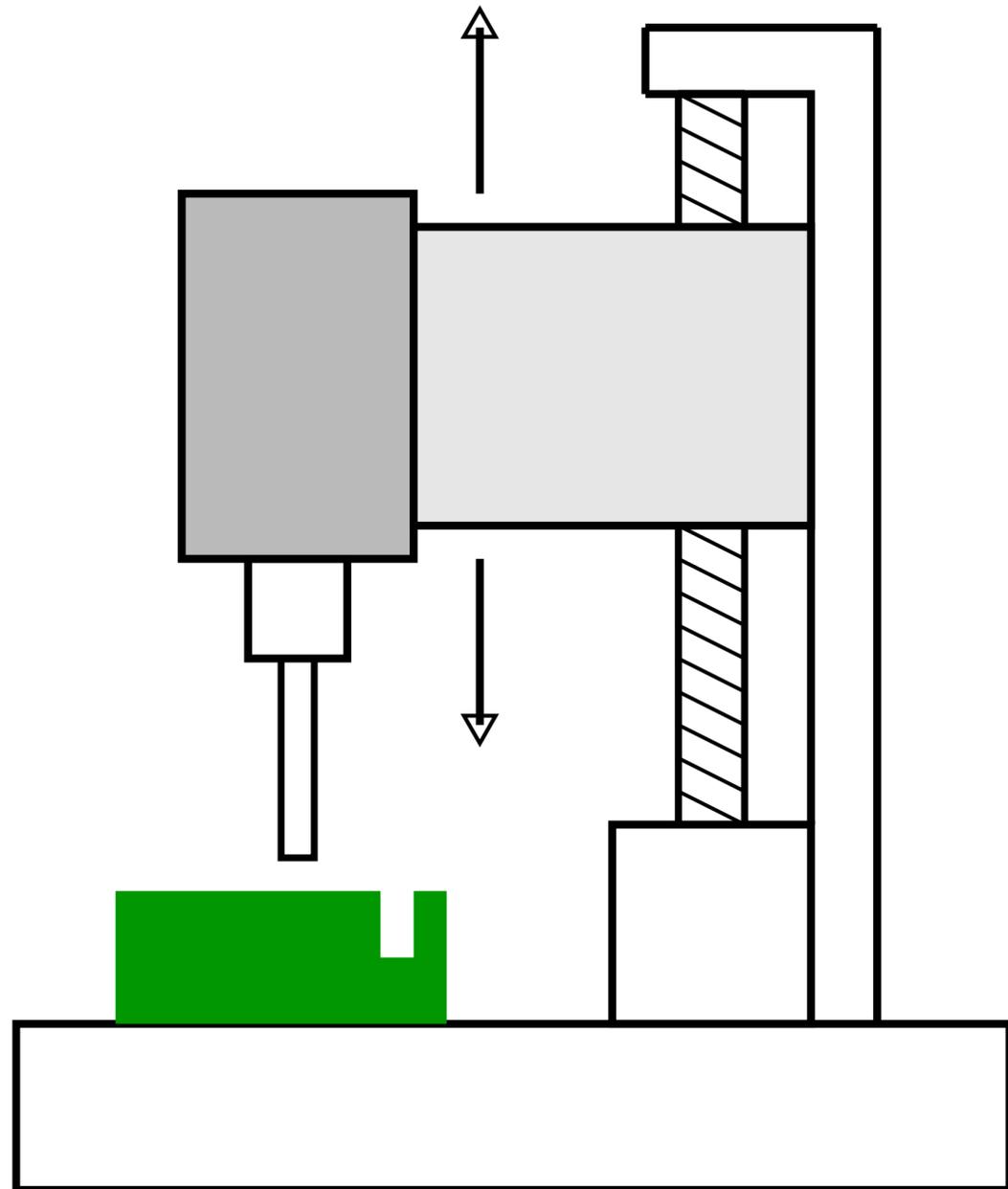
- Description par un graphe d'état
- Conception d'une machine d'état
- Solution avec un microcontrôleur !



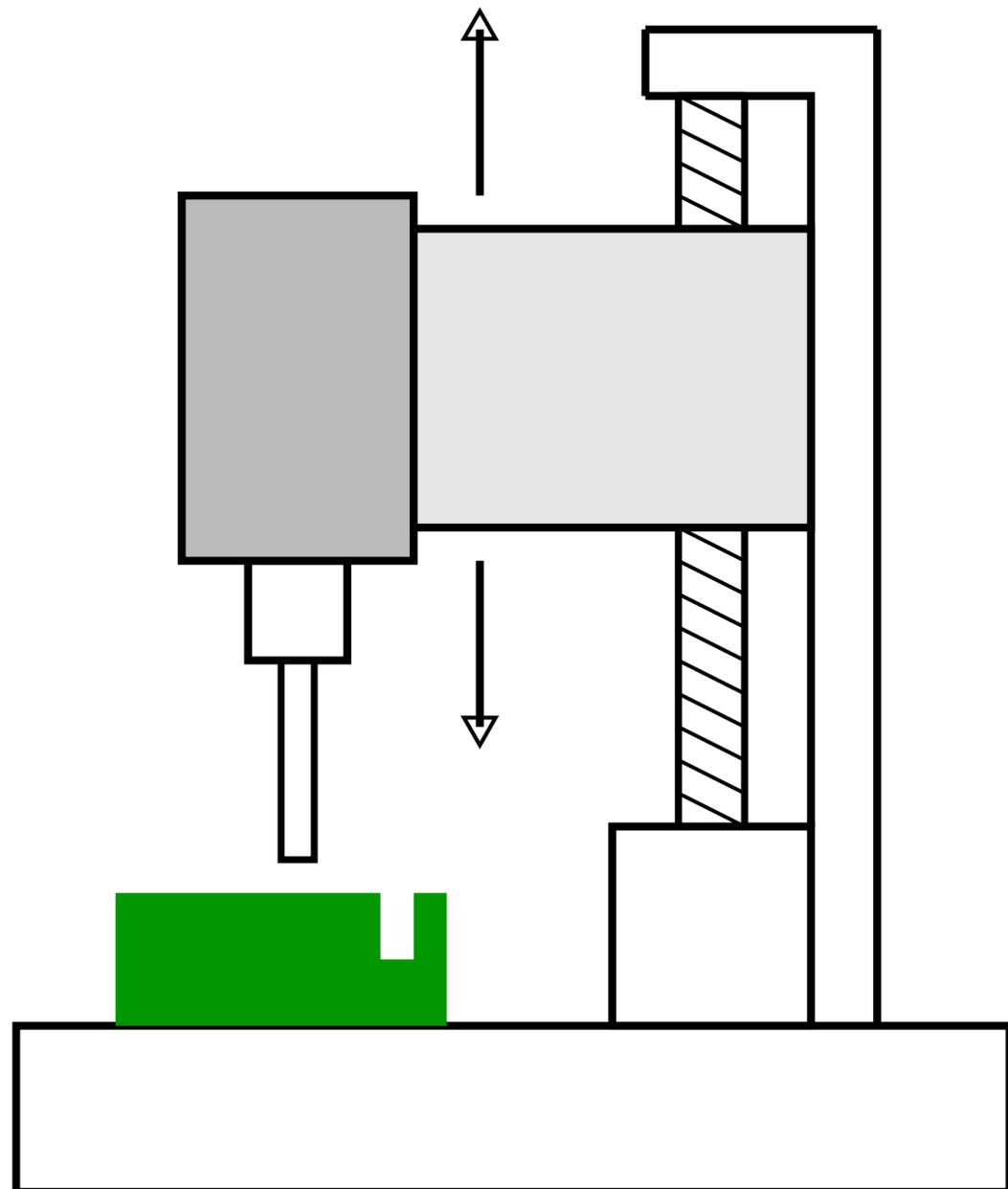
Entrée A	Entrée B	Entrée C	Sortie 1	Sortie 2
0	0	0	1	1
0	0	1	0	0
0	1	0	0	0
0	1	1	1	1
1	0	0	1	1
1	0	1	0	0
1	1	0	1	1
1	1	1	1	1

Table de vérité

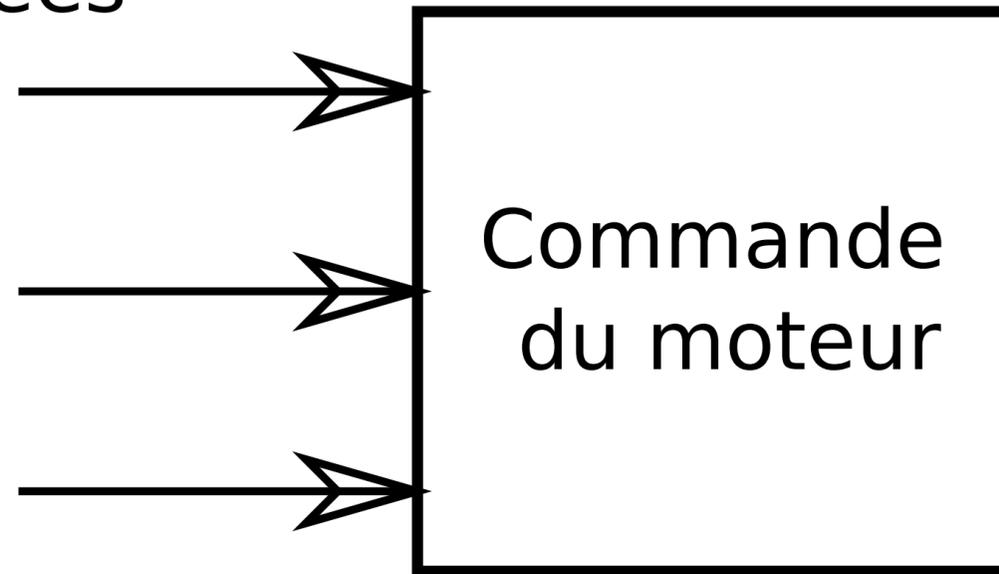
# Un exemple : commande du mouvement d'une perceuse



# Identifier les entrées et les sorties

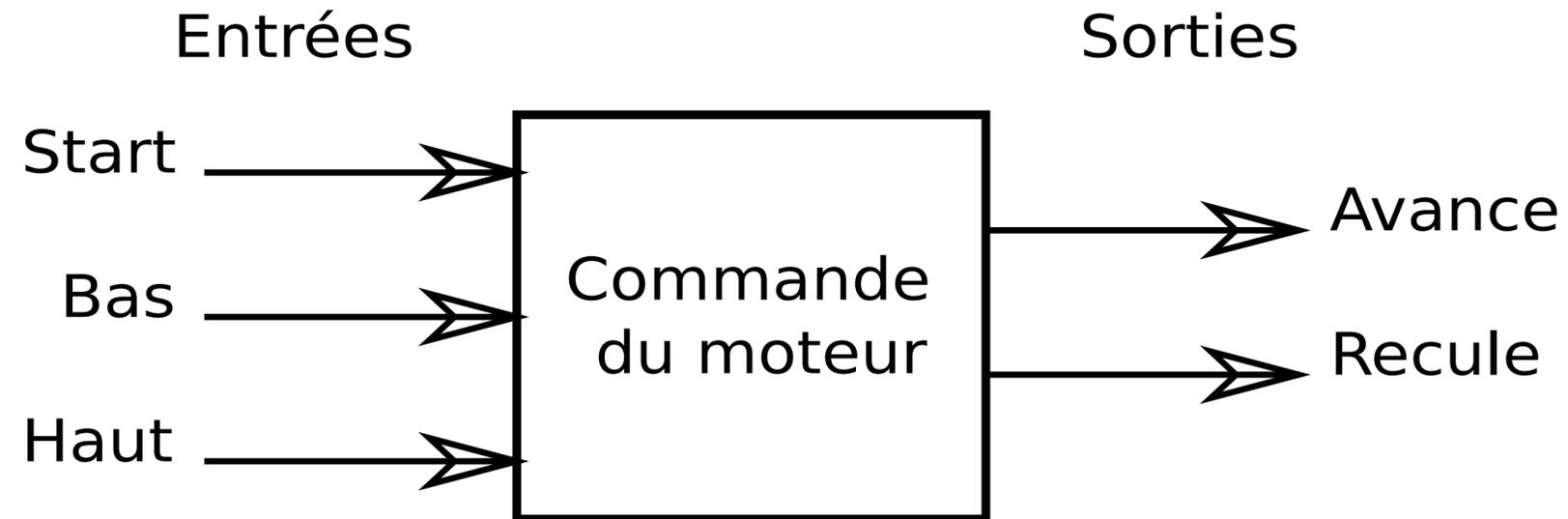


Entrées



Sorties

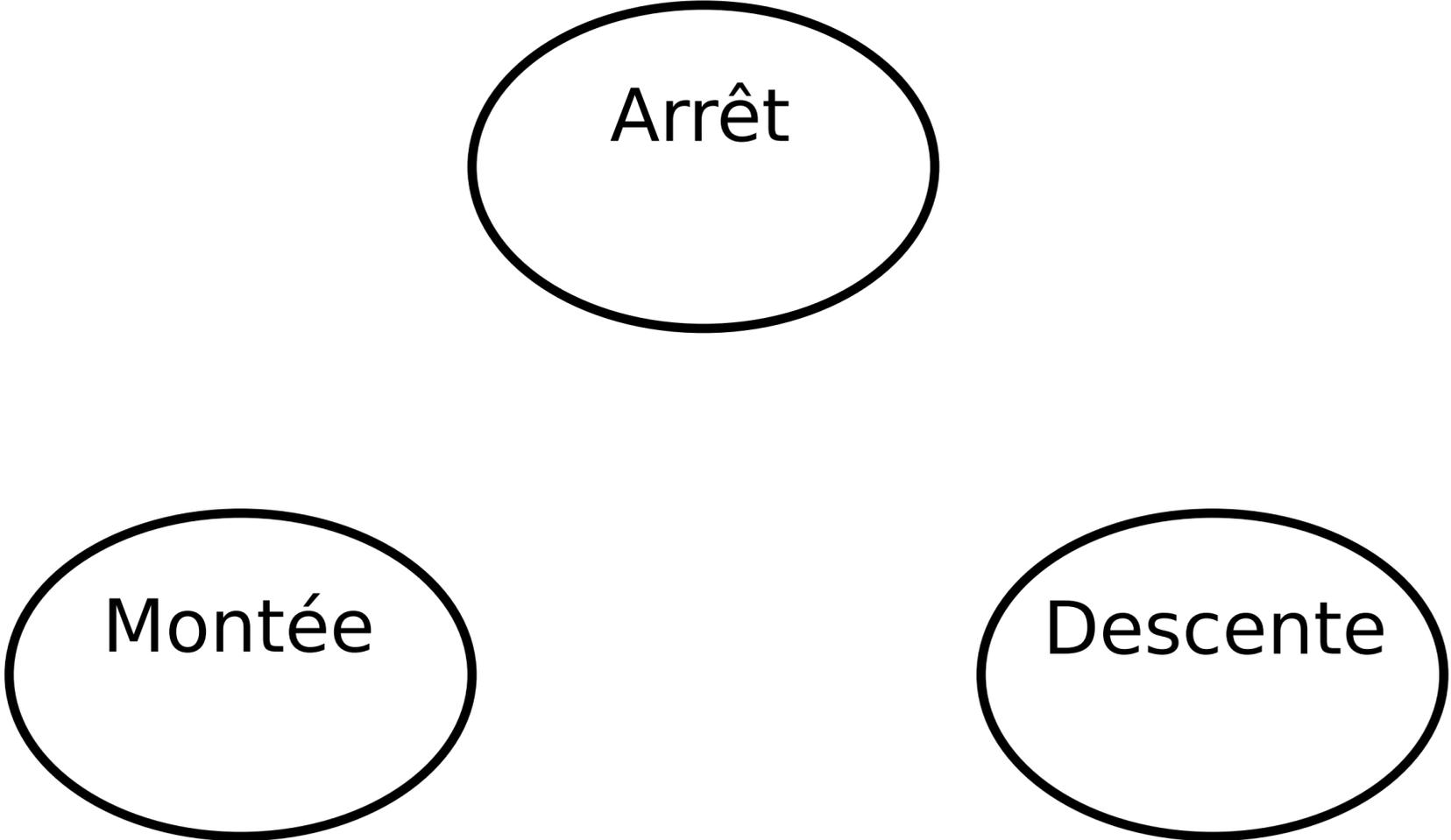
# Systemes logiques combinatoires ?



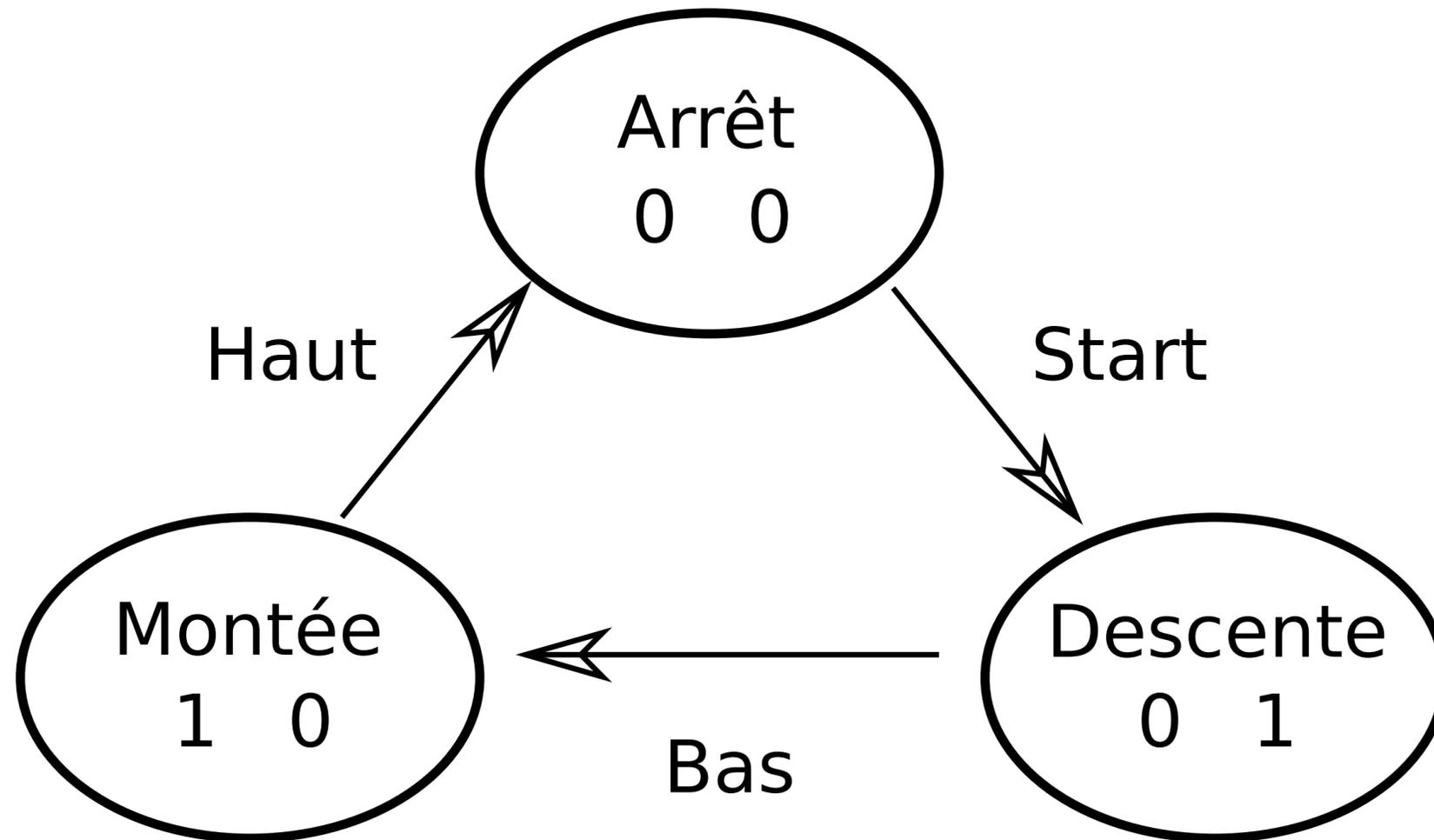
Start	Haut	Bas	Avance	Recule
0	0	0		
0	0	1		
0	1	0		
0	1	1		
1	0	0		
1	0	1		
1	1	0		
1	1	1		

Table de vérité ?

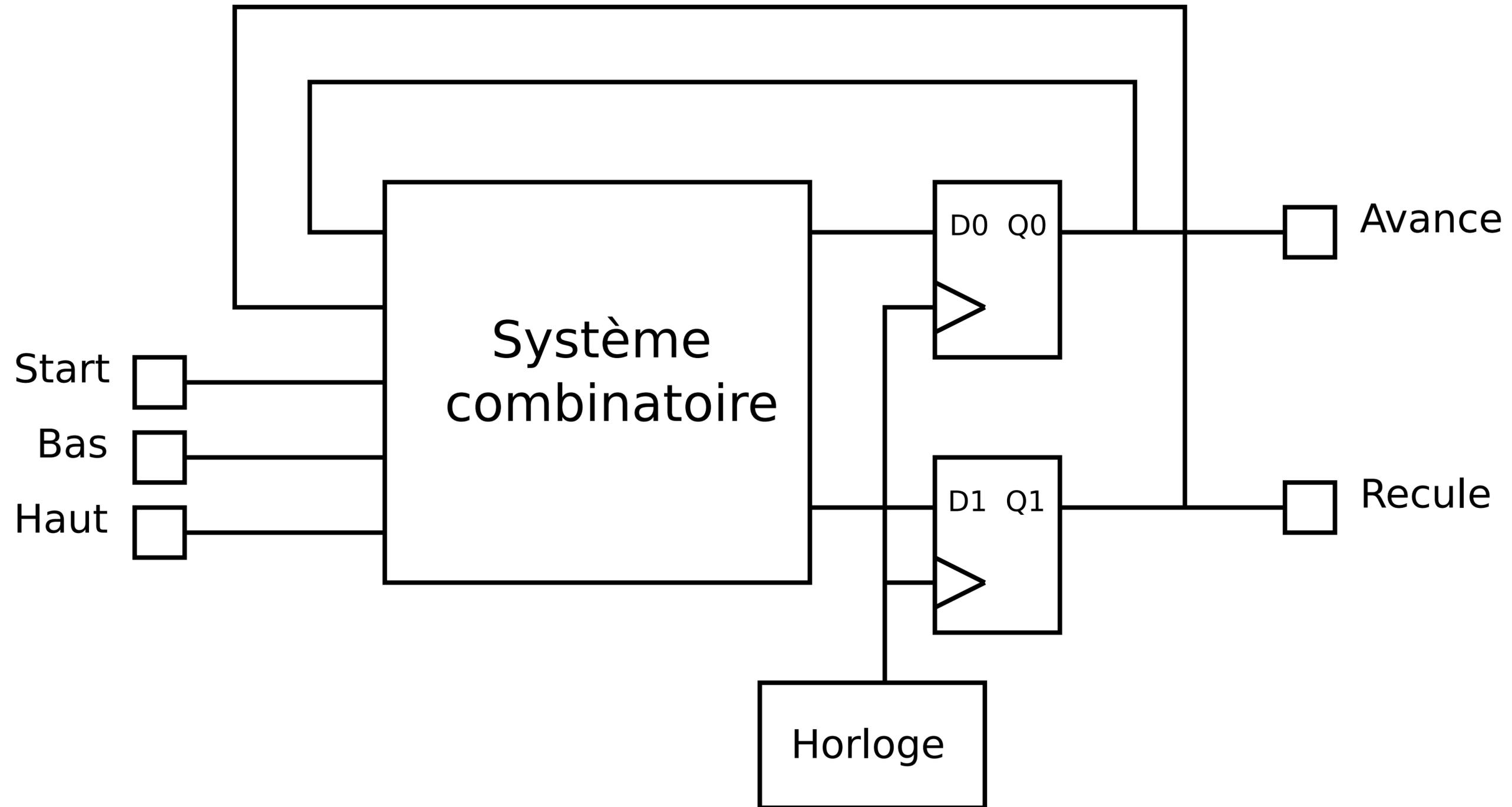
# Graphe d'état



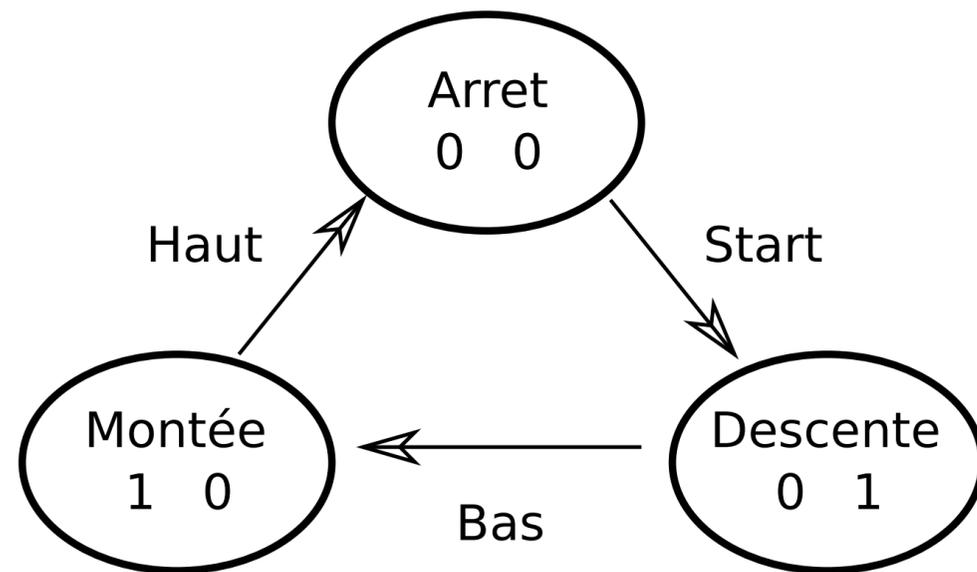
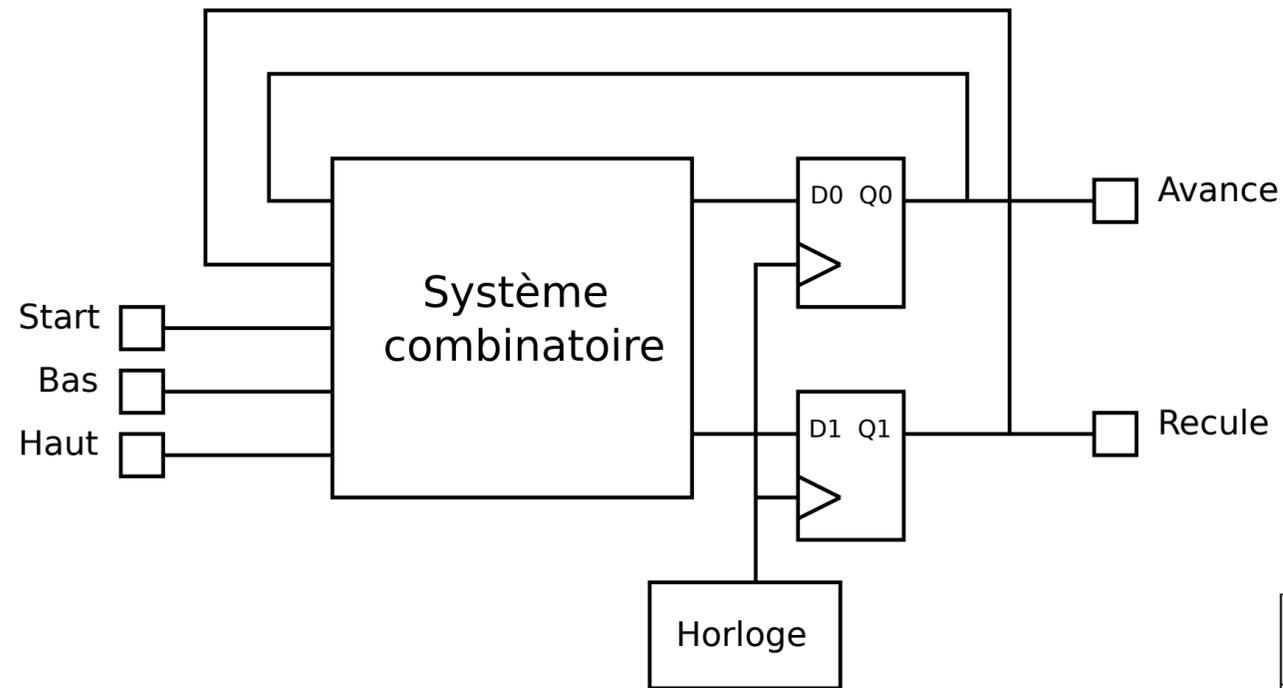
# Graphe d'état



# Machine d'état

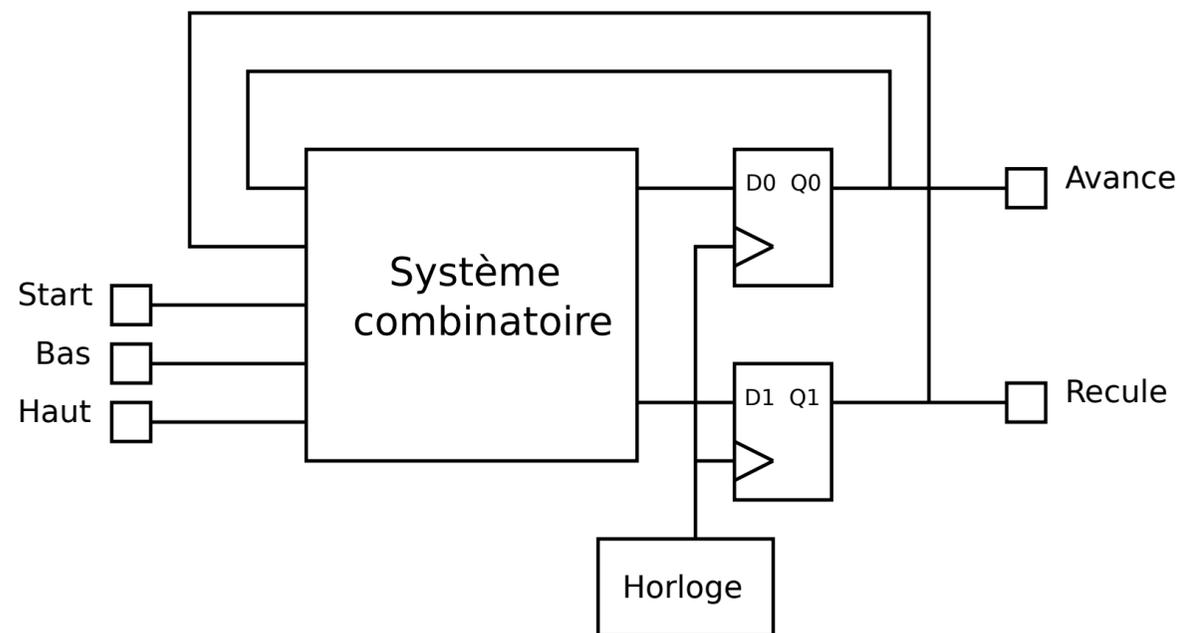


# Machine d'état



Start	Bas	Haut	Reculé	Avance	Reculé+	Avance+

# Machine d'état

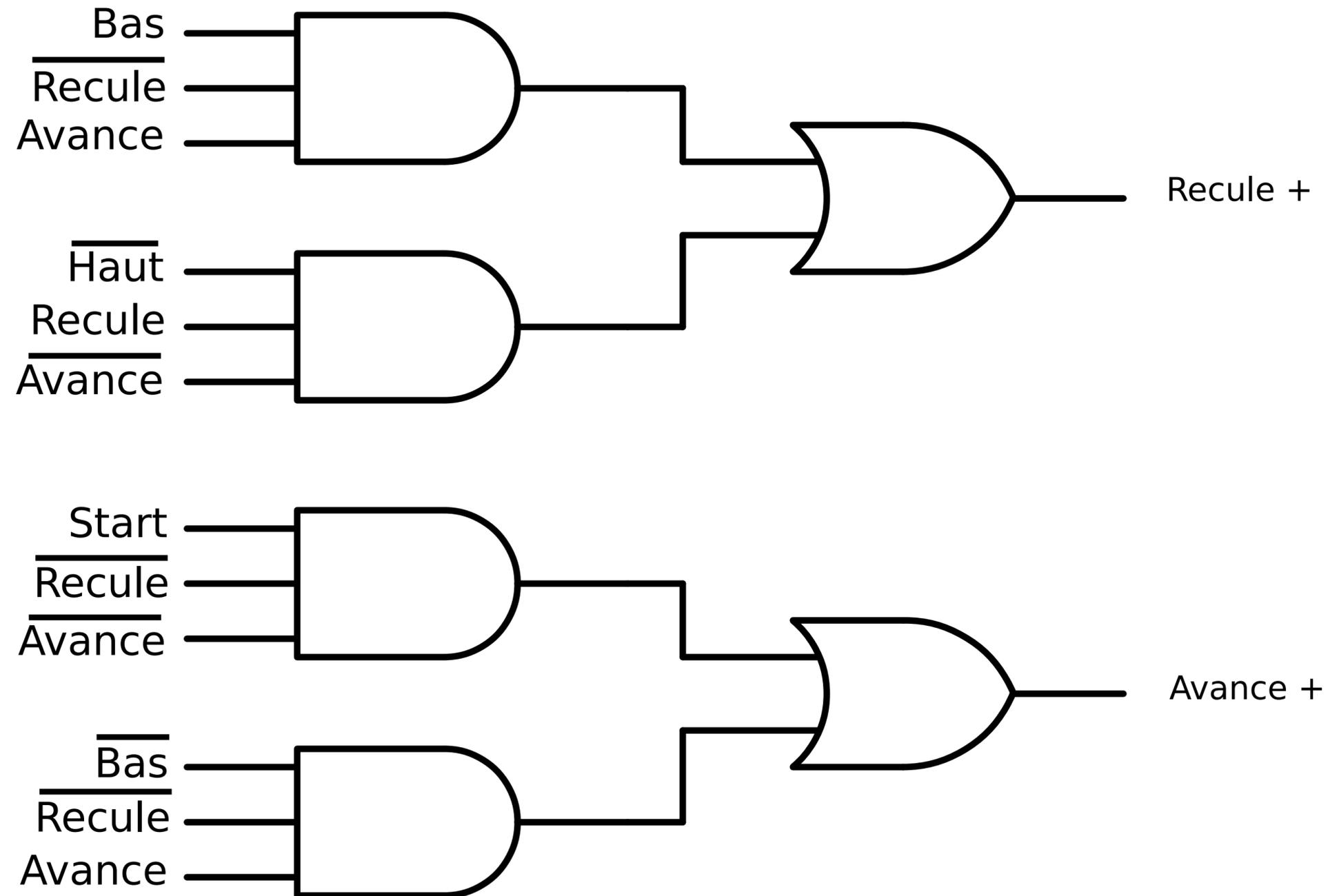


Start	Bas	Haut	Reculé	Avance	Reculé+	Avance+
0	x	x	0	0	0	0
1	x	x	0	0	0	1
x	0	x	0	1	0	1
x	1	x	0	1	1	0
x	x	0	1	0	1	0
x	x	1	1	0	0	0
x	x	x	1	1	0	0

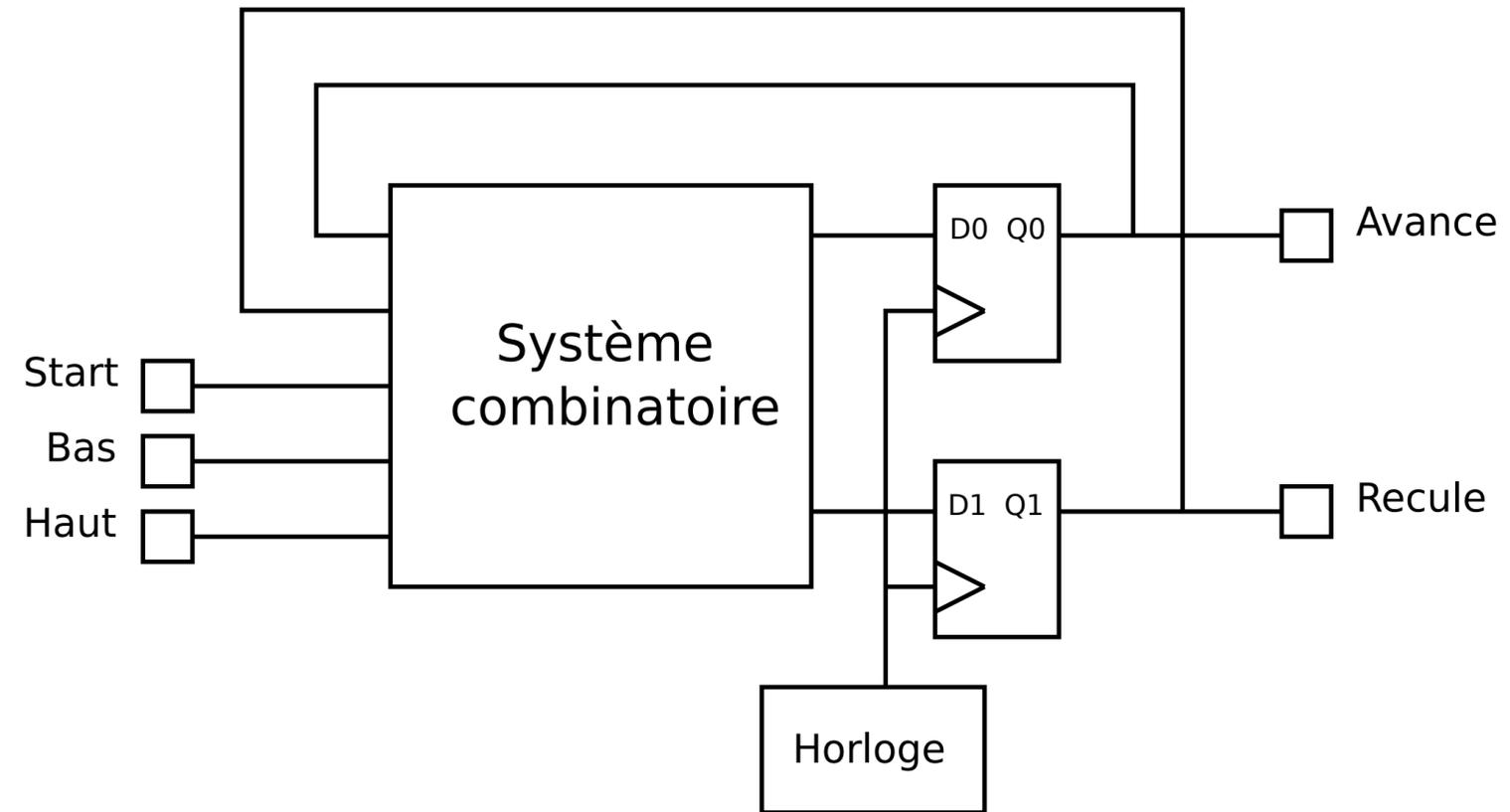
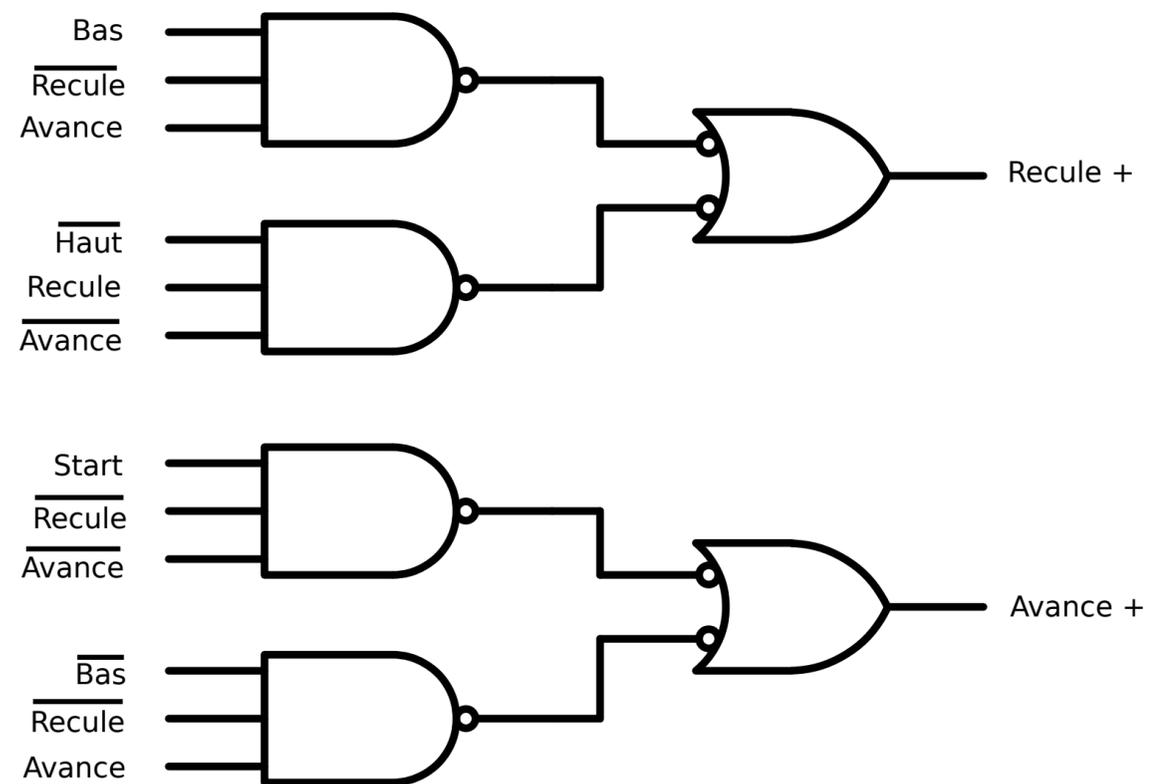
$$\text{Reculé+} = (\text{Bas} \cdot \overline{\text{Reculé}} \cdot \text{Avance}) + (\overline{\text{Haut}} \cdot \text{Reculé} \cdot \overline{\text{Avance}})$$

$$\text{Avance+} = (\text{Start} \cdot \overline{\text{Reculé}} \cdot \overline{\text{Avance}}) + (\overline{\text{Bas}} \cdot \overline{\text{Reculé}} \cdot \text{Avance})$$

# Schéma logique



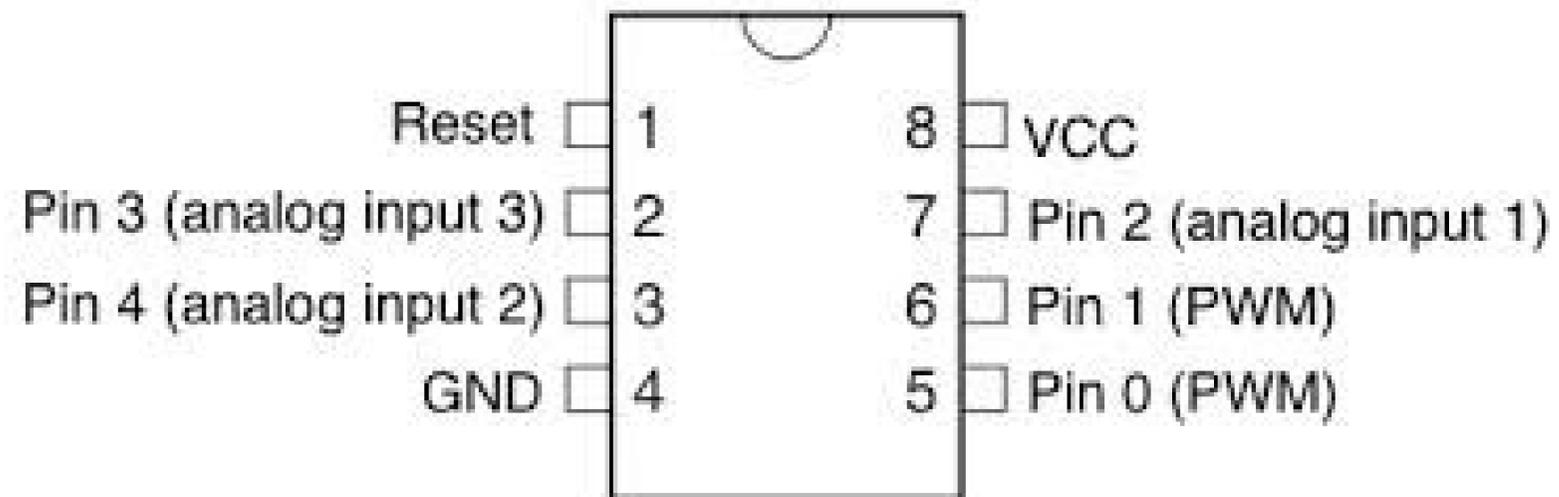
# Réalisation avec des circuits intégrés logiques



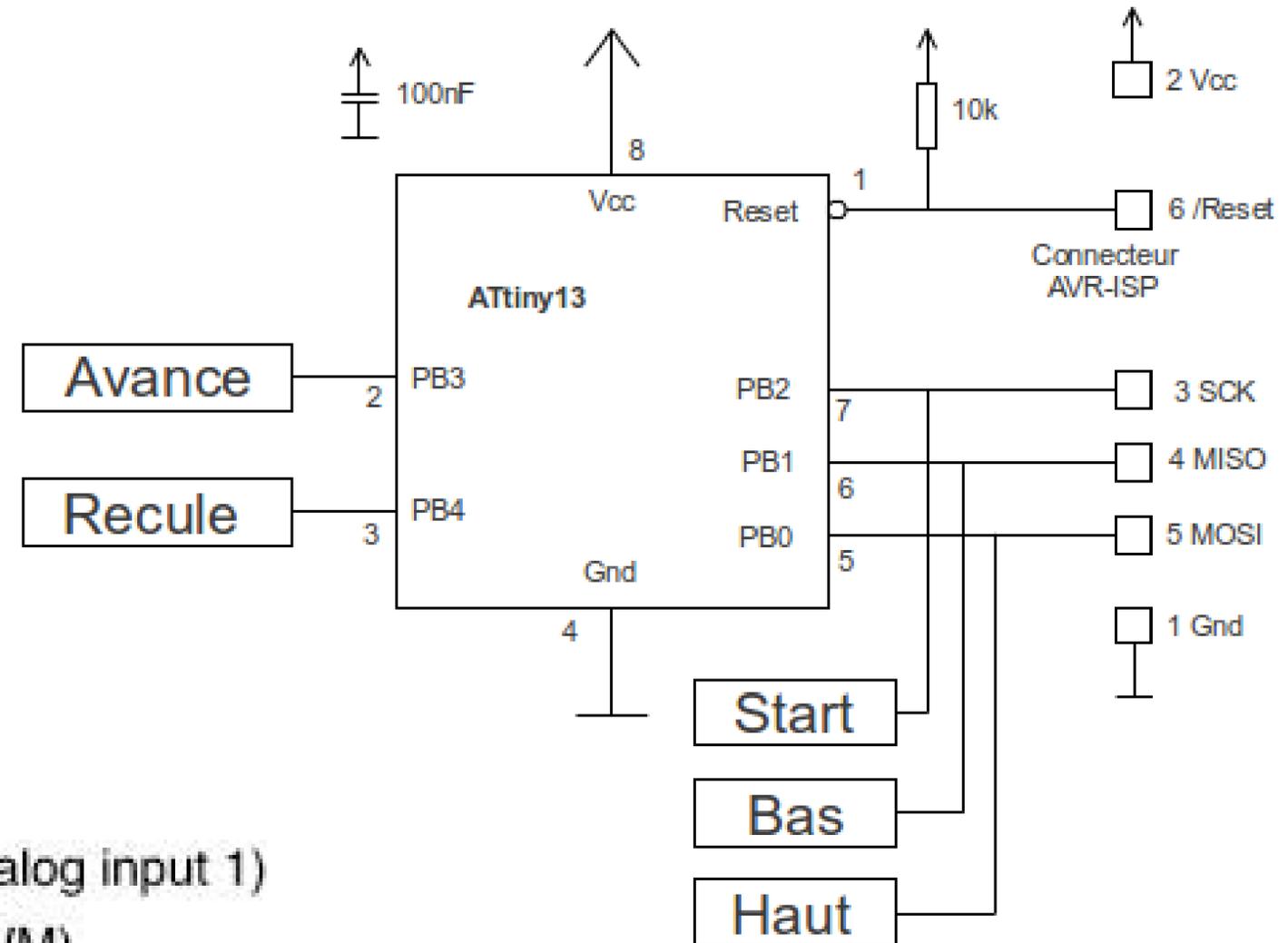
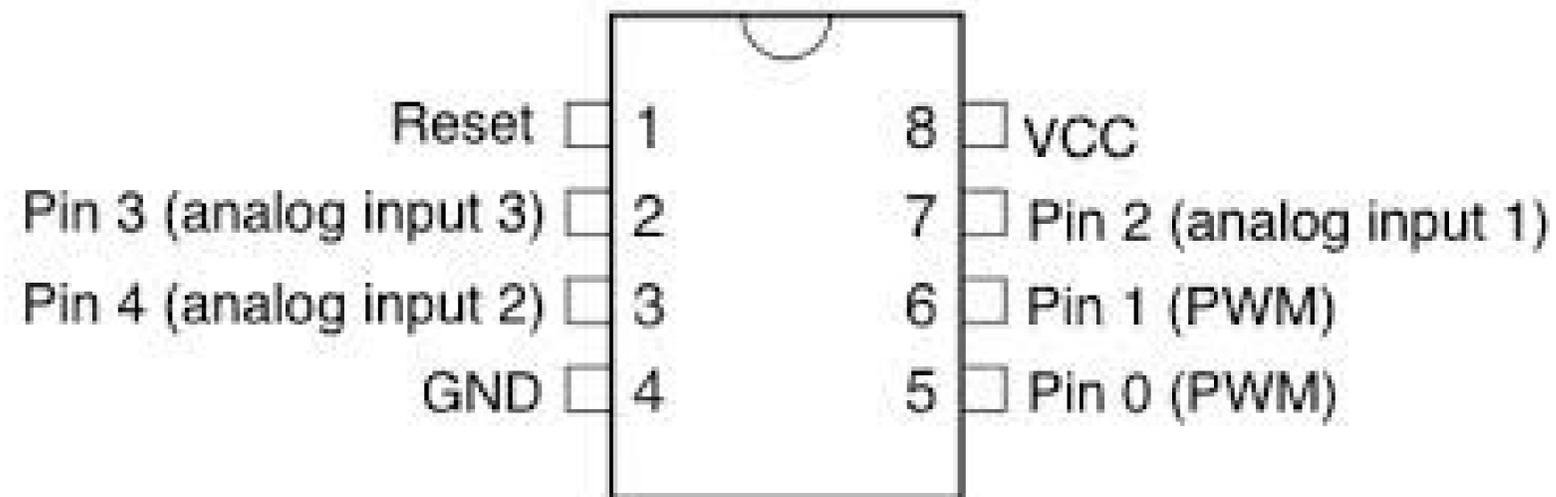
# Solution plus simple ?

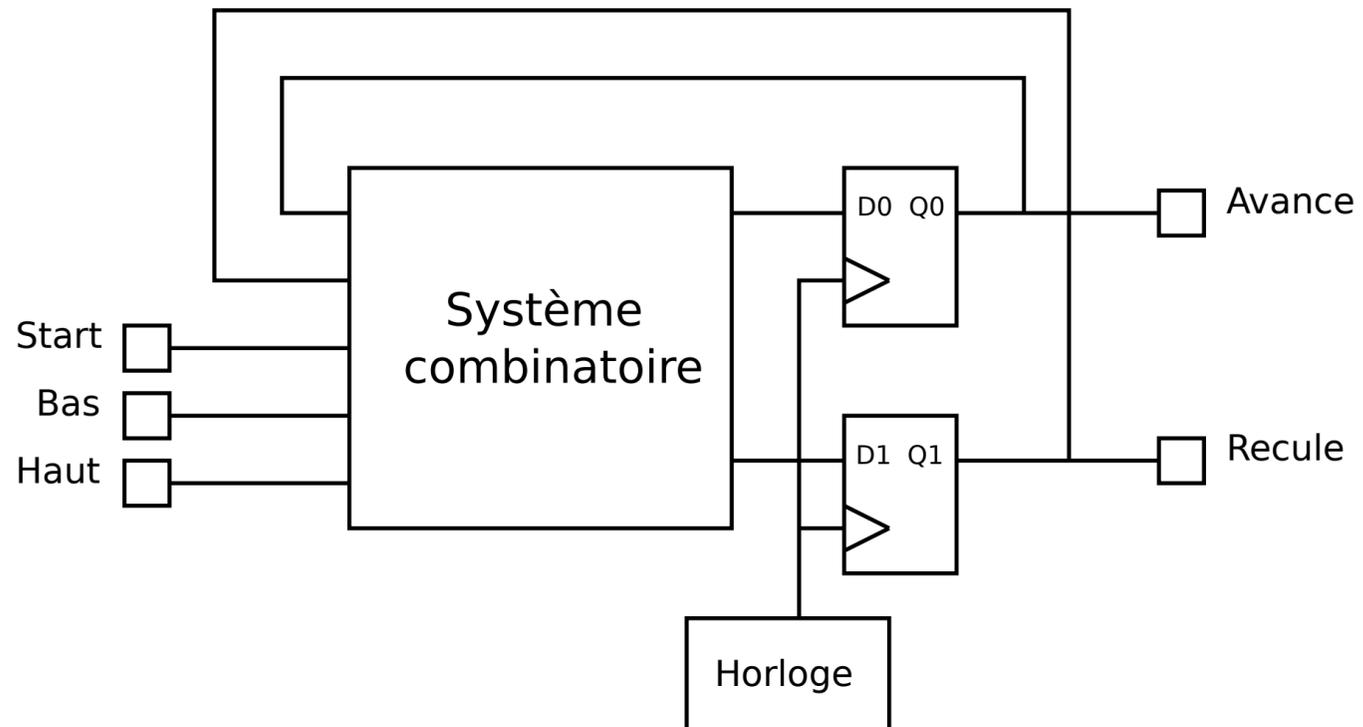


# Avec un microcontrôleur : schéma



# Avec un microcontrôleur : schéma





```
void loop(){  
    // Lit les valeurs des entrées :  
    Start = digitalRead(StartPin);  
    Bas = digitalRead(BasPin);  
    Haut = digitalRead(HautPin);  
  
    // Calcule le nouvel état :  
    AvancePlus = (Start && !Reculé && !Avance) ||  
                (!Bas && !Reculé && Avance);  
    ReculéPlus = (Bas && !Reculé && Avance) ||  
                (!Haut && Reculé && !Avance);  
  
    // Enregistre le nouvel état :  
    Avance = AvancePlus;  
    Reculé = ReculéPlus;  
  
    // Active les sorties :  
    digitalWrite(AvancePin, Avance);  
    digitalWrite(ReculéPin, Reculé);  
}
```

## Systemes séquentiels

- Description par un graphe d'état
- Conception d'une machine d'état
- Solution avec un microcontrôleur !