

Module 7 : Registre à décalage

Applications mobiles et objets connectés 420-W48-SF

Objectifs

- Circuits intégrés
 - Fonction, bornes
 - Programmation
- Registre à décalage
 - Principe de fonctionnement
 - Branchement sur circuit
 - Animation
- Programmation - Exemples de codes
 - Envoi de données au registre à décalage
 - Décomposition d'un entier en bits

Circuits intégrés

- Fonctions de base implantées dans le matériel
 - Exemples: additionneurs binaires, décalage de bits, portes logiques, ...
- Propriétés :
 - Puce enfermé dans un boîtier miniature
 - Bornes en nombre varié, de chaque côté
 - Spécifications dans les fiches techniques (« data sheet »)



Registre
décalage



Additionneur
4 bits



Optocoupleur

Circuits intégrés

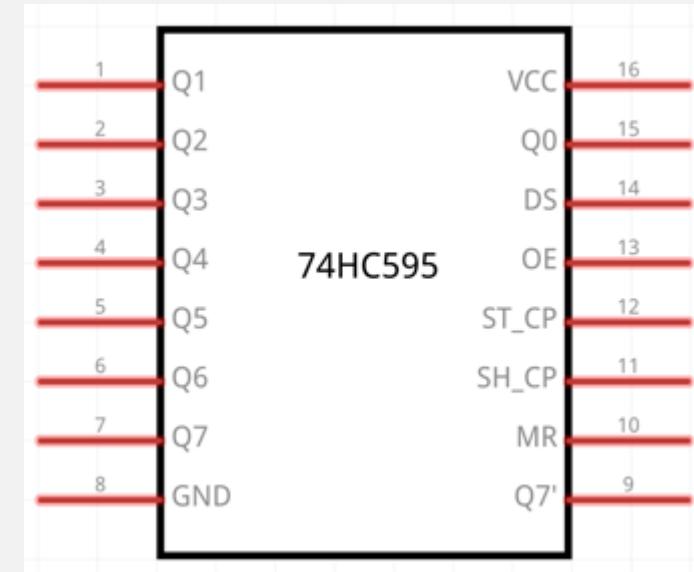
- Fonctions de base implantées dans le matériel
 - Pour
 - Performant
 - Uniformité de l'implantation des fonctions
 - Réduction du code de programmation
 - Espace mémoire libérée pour les applications
 - Contre
 - Câblage et de montage préparatoire
 - Consommation d'électricité !
 - Évolution de versions difficile



Registre à décalage



- Fonctions :
 - Recevoir des bits, un à un, sur la borne d'entrée DS
 - Accumuler les bits dans 8 mémoires internes
 - Transférer tous les bits simultanément sur 8 bornes de sortie Q0 à Q7
 - Q7* : permet de chainer plusieurs registres (non utilisé dans le cours)
- Application
 - **Économie de bornes Arduino**
 - 3 bornes suffisent pour contrôler un grand nombre de DELs en sortie
 - Borne « cascade » : permet de chainer plusieurs groupes de 8 DELs en séquence par exemple

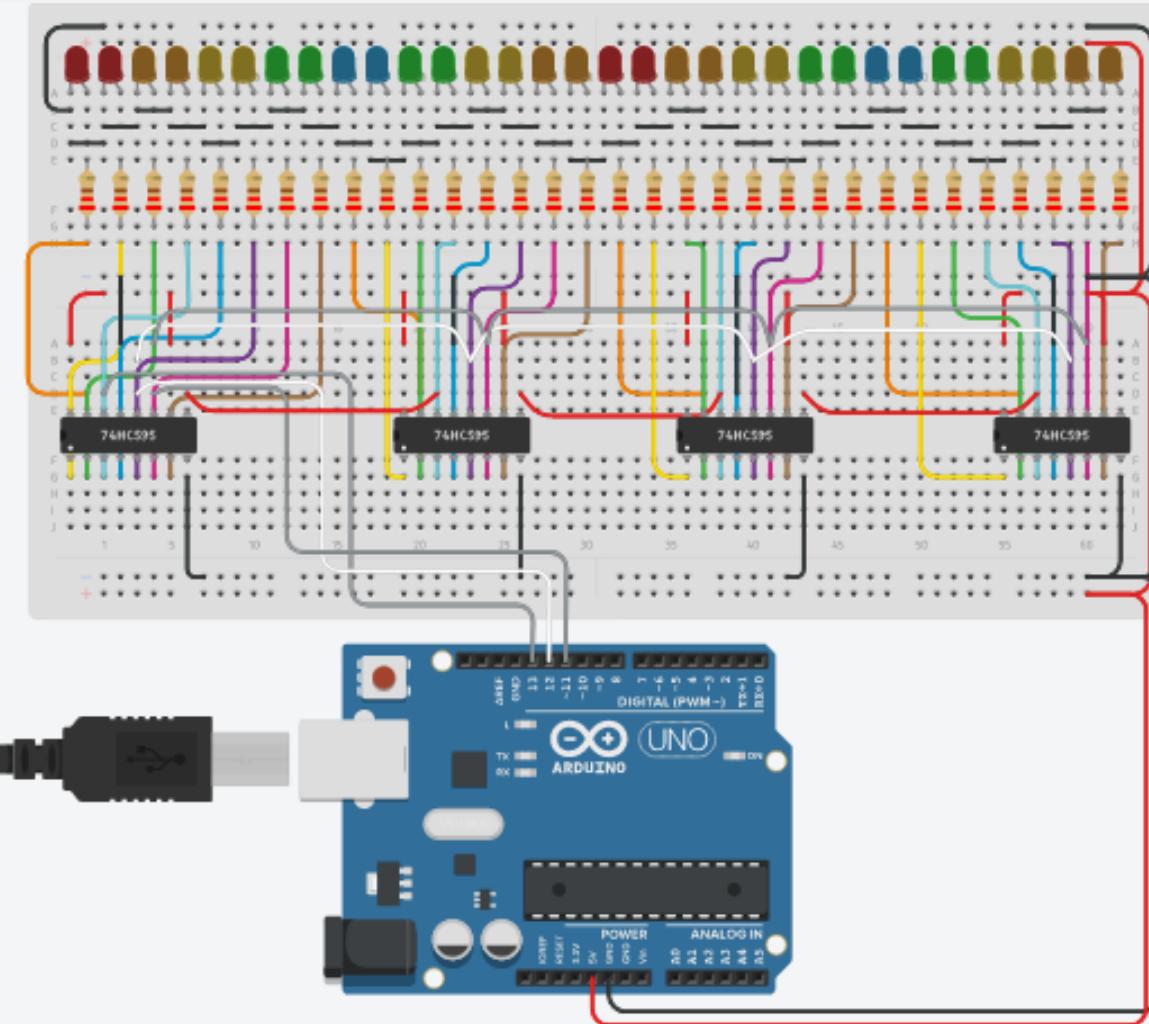


Registre à décalage – Exemple « 32 DELs »

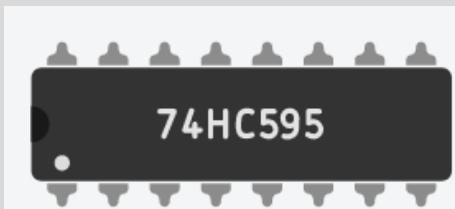
- TinkerCad : 32 DELs alimentées par seulement 3 bornes Arduino !

**Ne pas reproduire sur un Arduino physique
Trop de courant tiré !**

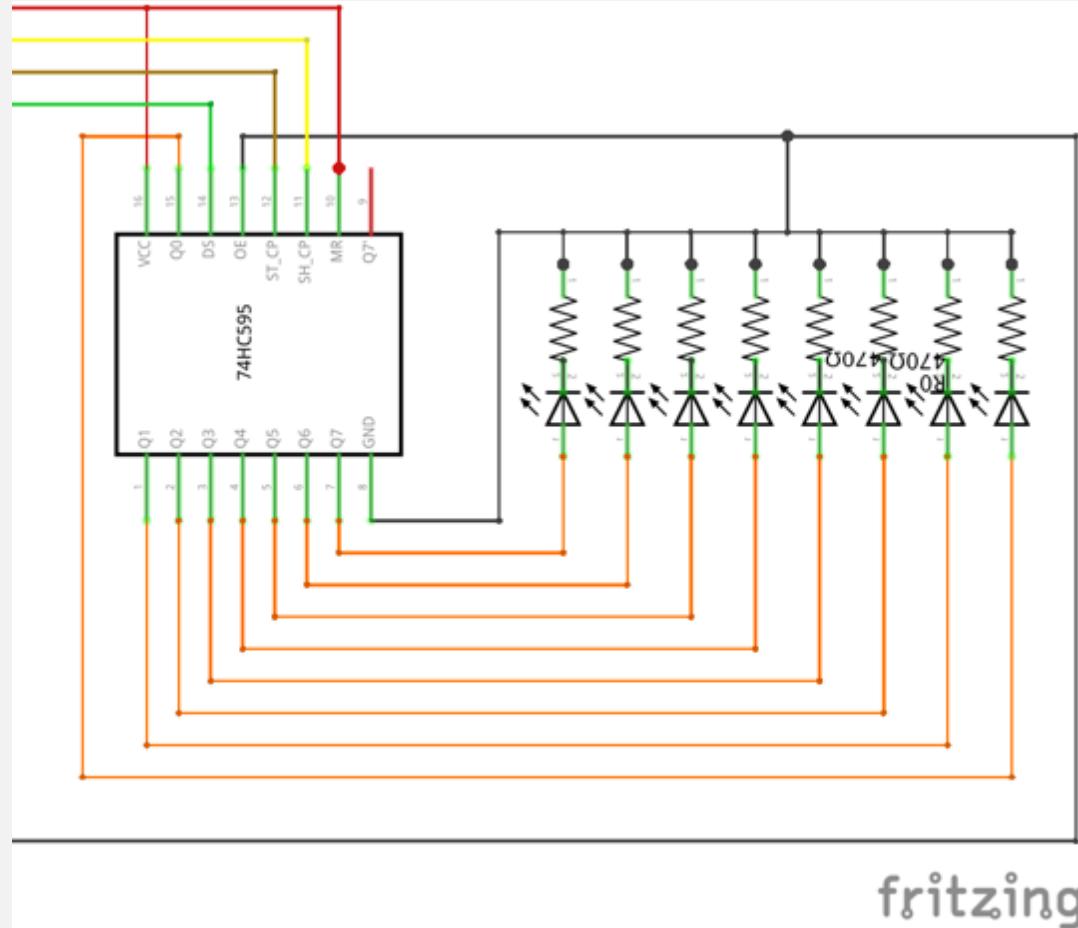
- Référence : <http://valentin.cassayre.me>



Registre à décalage – Branchement

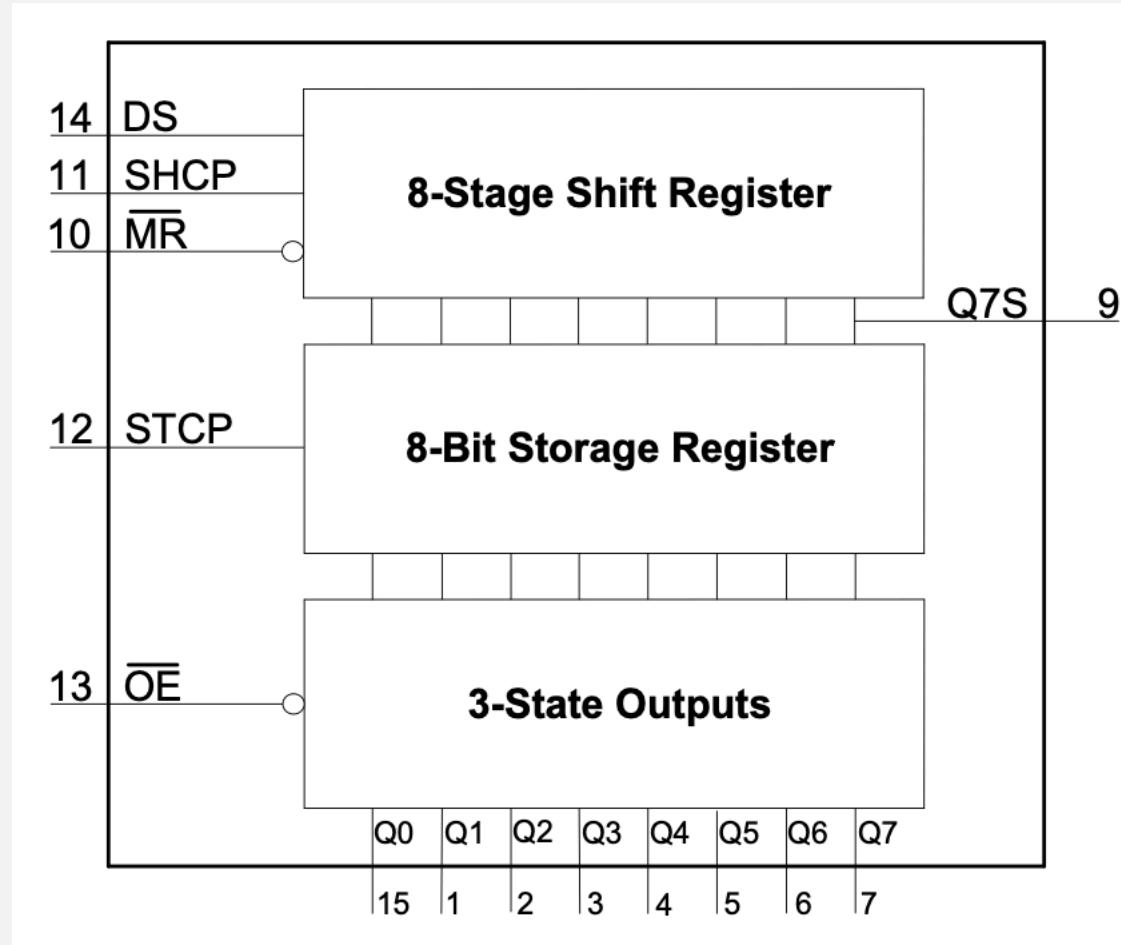


- Alimentation : VCC (+ : 16) / GND (- : 8)
- DS (14) : entrée (bit à insérer)
- SH_CP (11) : Shift clock
- ST_CP (12) : Storage clock
- Q0 à Q7 (1-7, 15) : sorties 8 bits
- Q7' (9) : cascade de registres
- MR (+ : 10) : reset function
- OE (- : 13) : output enable



!! Attention : il ne supporte qu'un courant de 70 mA !!

Registre à décalage – Architecture interne



Registre à décalage – Algorithme envoi données

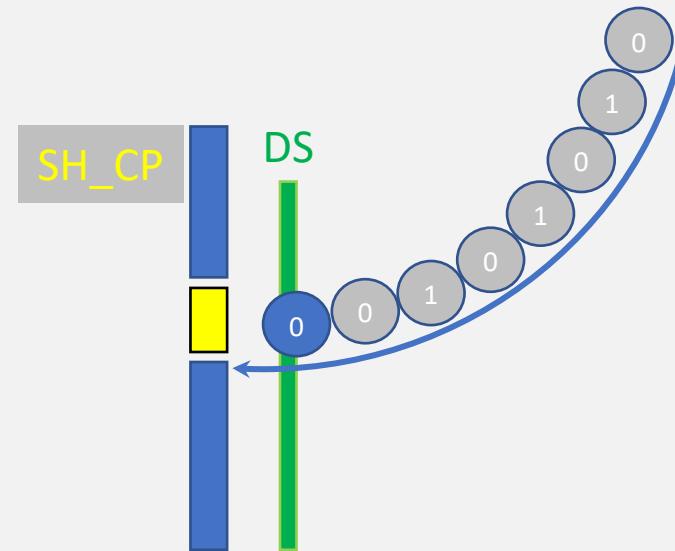
- Au départ $ST_CP \leftarrow LOW$
- Pour chaque bit extrait de gauche à droite avec un décalage et un masque :
 - $SH_CP \leftarrow LOW$
 - $DS \leftarrow$ La valeur du bit
 - $SH_CP \leftarrow HIGH$
- À chaque fois que la borne SH_CP passe de LOW à $HIGH$,
 - Les bits du décalage se déplacent vers la gauche
 - Le bit le plus à gauche est perdu
 - Le bit situé sur la borne DS est copié à la position 0 du registre décalage
- Une fois les bits envoyés, passer la borne ST_CP à $HIGH$ pour durant quelques millisecondes, le temps de recopie des bits sur les bornes $Q0$ à $Q7$

Registre à décalage – Exemple

- Envoi du nombre 42 (0b00101010)

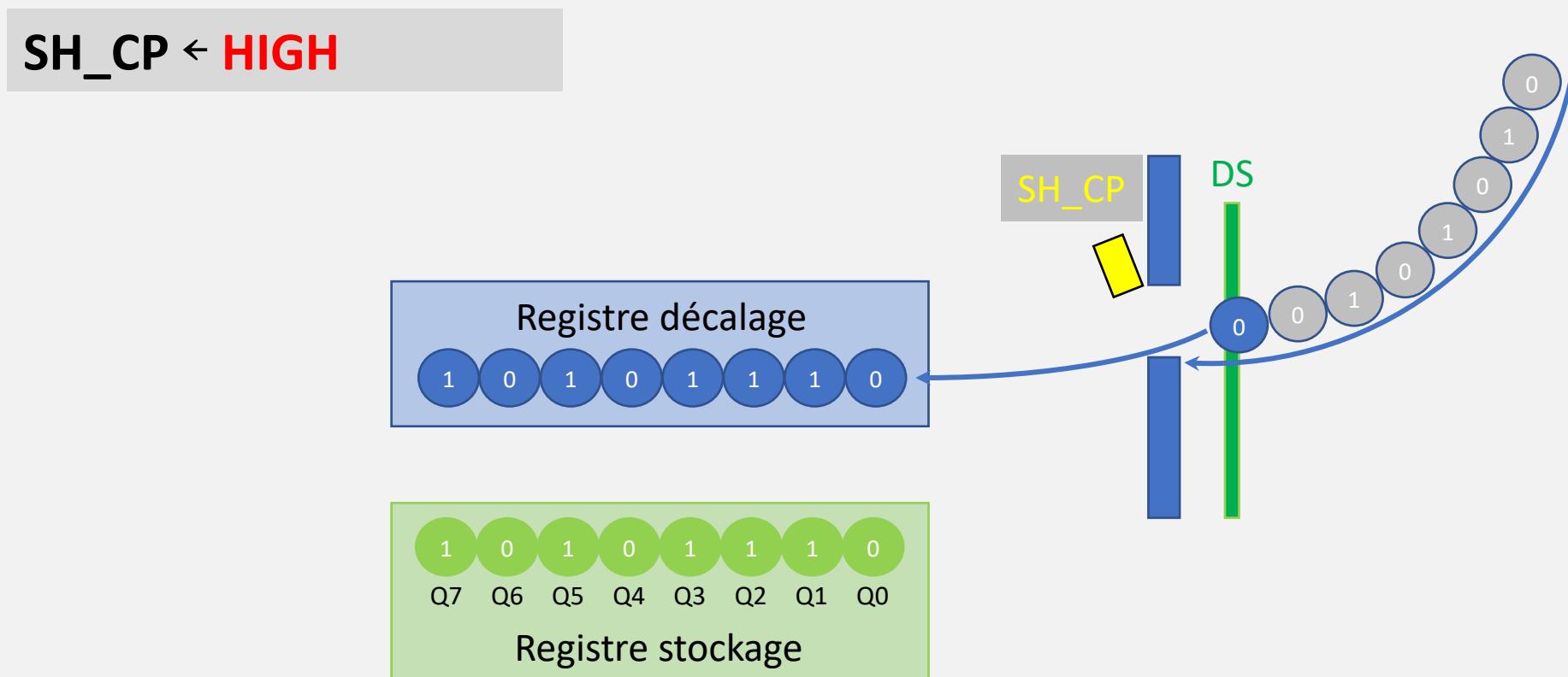
ST_CP \leftarrow LOW
SH_CP \leftarrow LOW
DS \leftarrow LOW

Anciennes valeurs
dans le registre



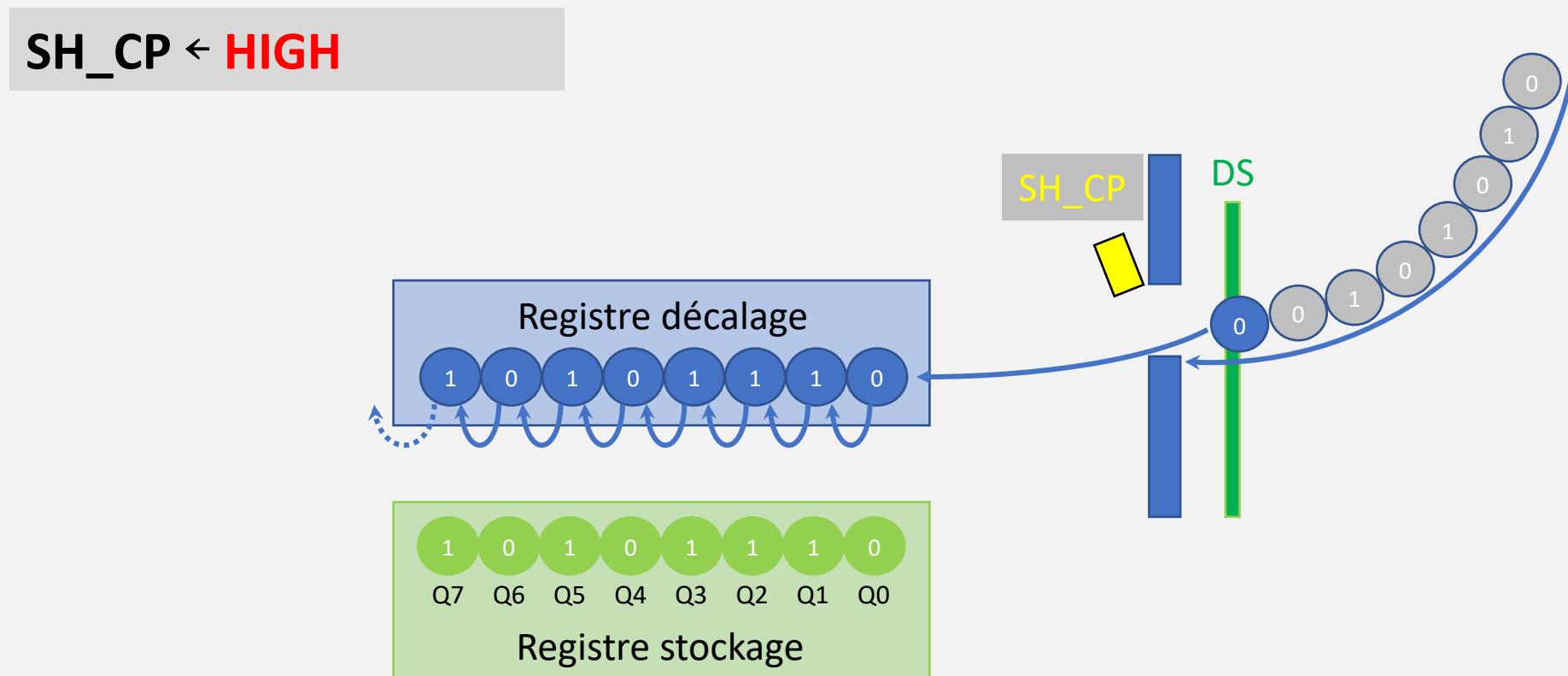
Registre à décalage – Exemple

- Envoi du nombre 42 (0b00101010)



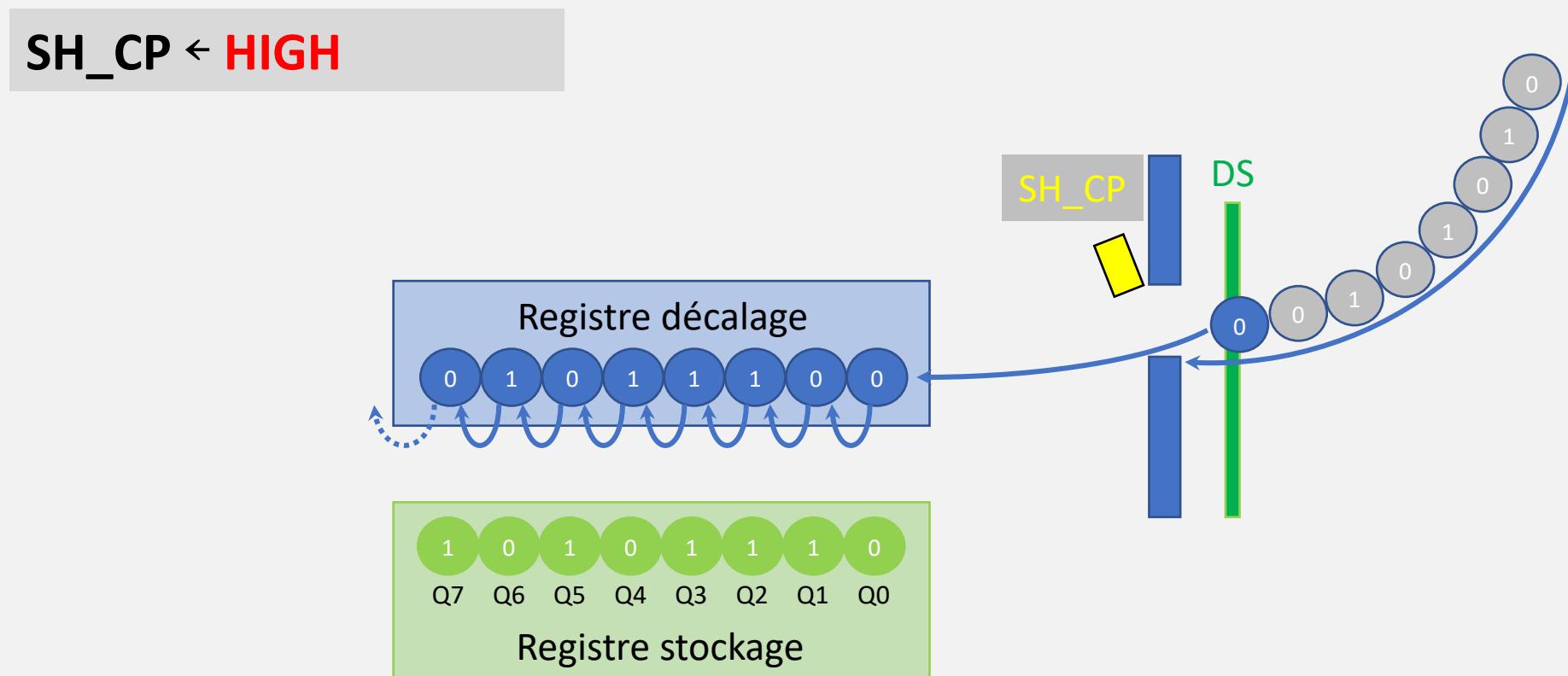
Registre à décalage – Exemple

- Envoi du nombre 42 (0b00101010)



Registre à décalage – Exemple

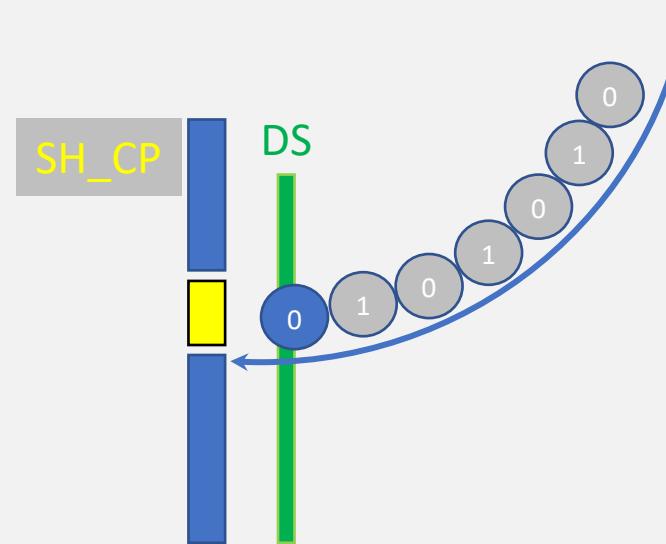
- Envoi du nombre 42 (0b00101010)



Registre à décalage – Exemple

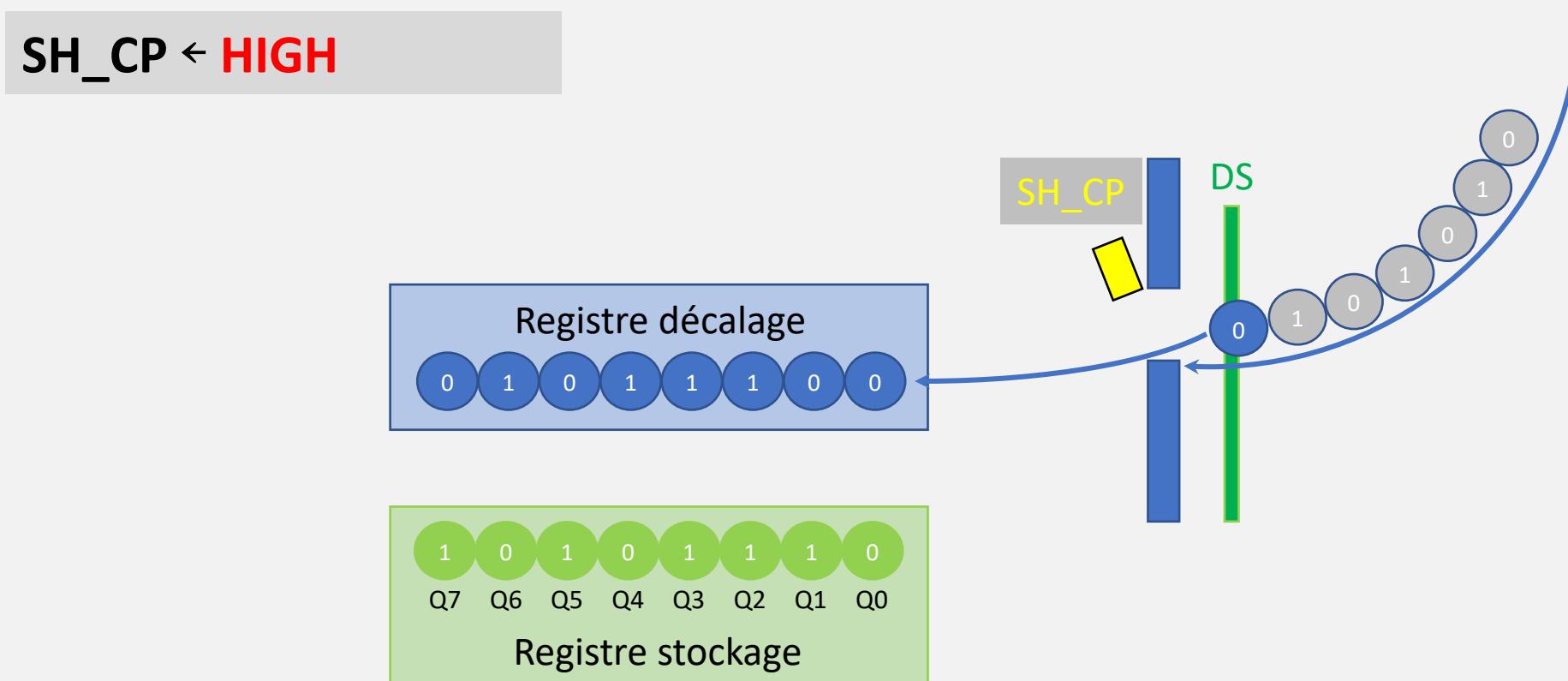
- Envoi du nombre 42 (0b00101010)

SH_CP \leftarrow LOW
DS \leftarrow LOW



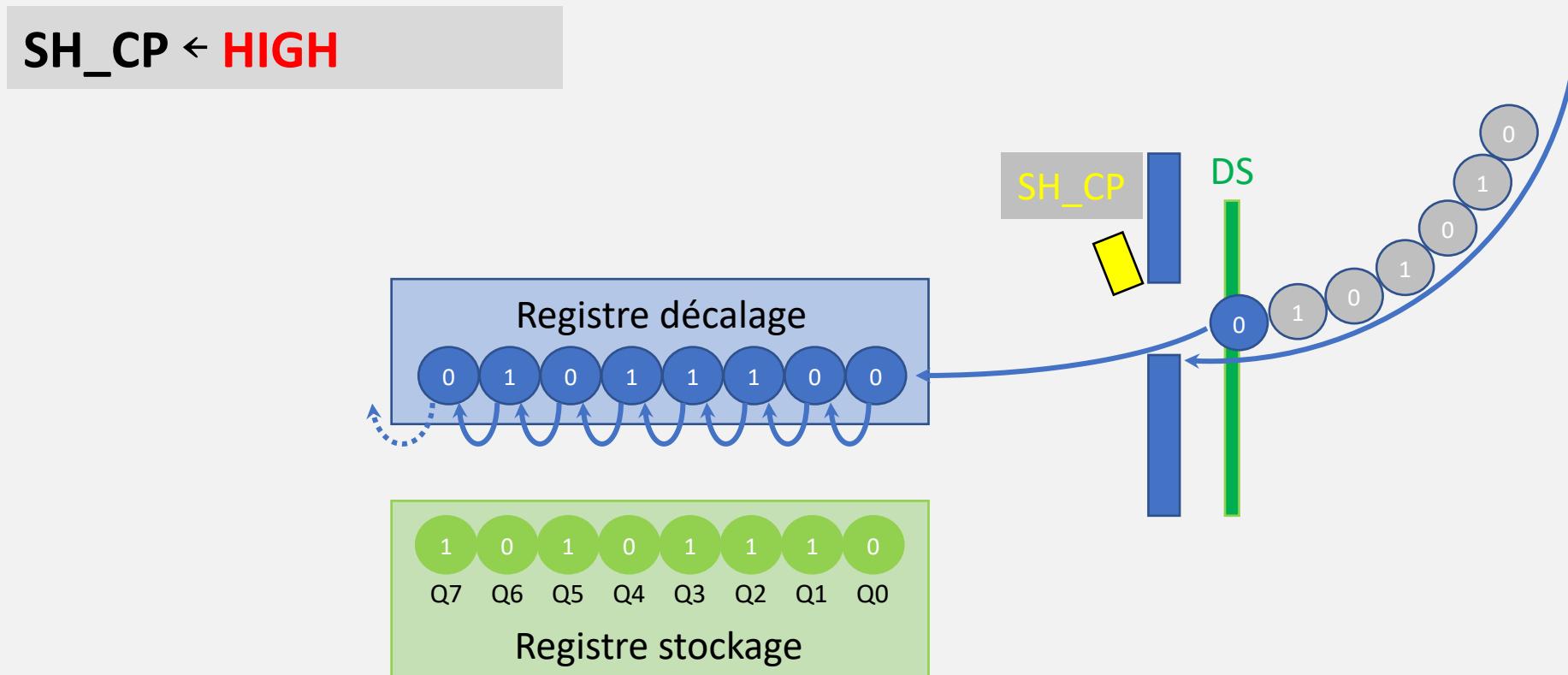
Registre à décalage – Exemple

- Envoi du nombre 42 (0b00101010)



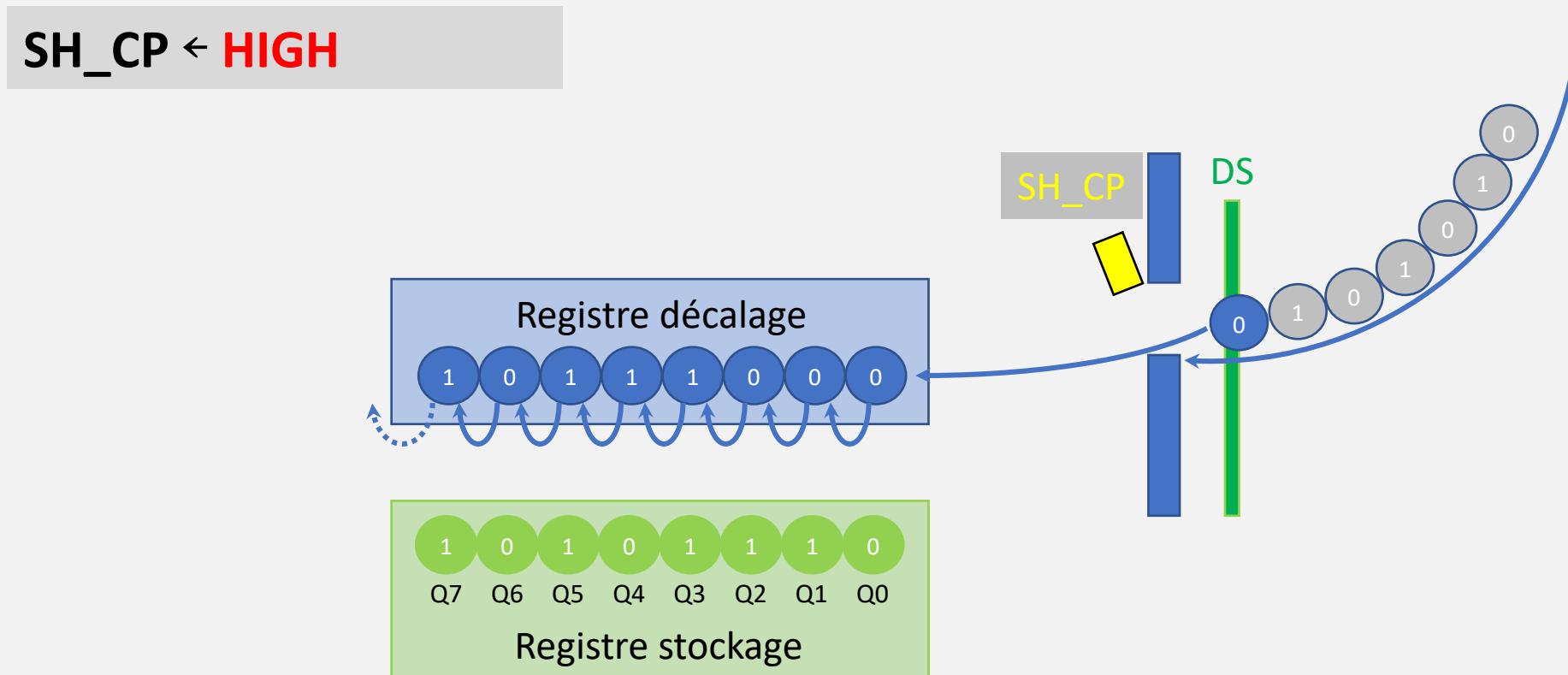
Registre à décalage – Exemple

- Envoi du nombre 42 (0b00101010)



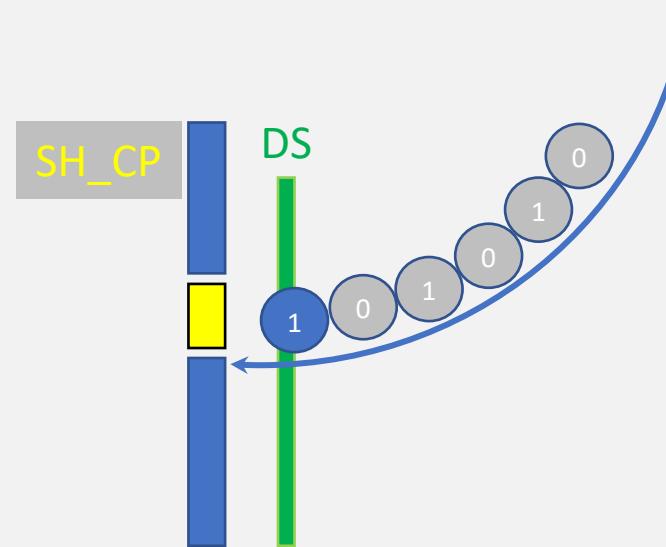
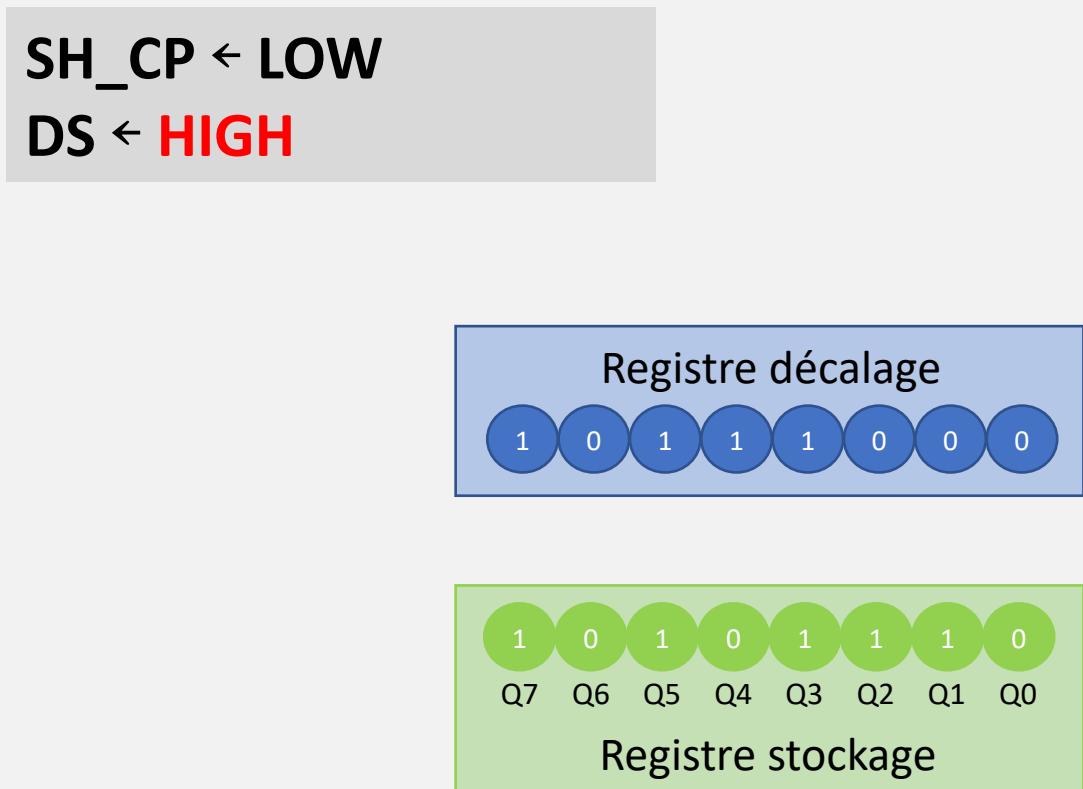
Registre à décalage – Exemple

- Envoi du nombre 42 (0b00101010)



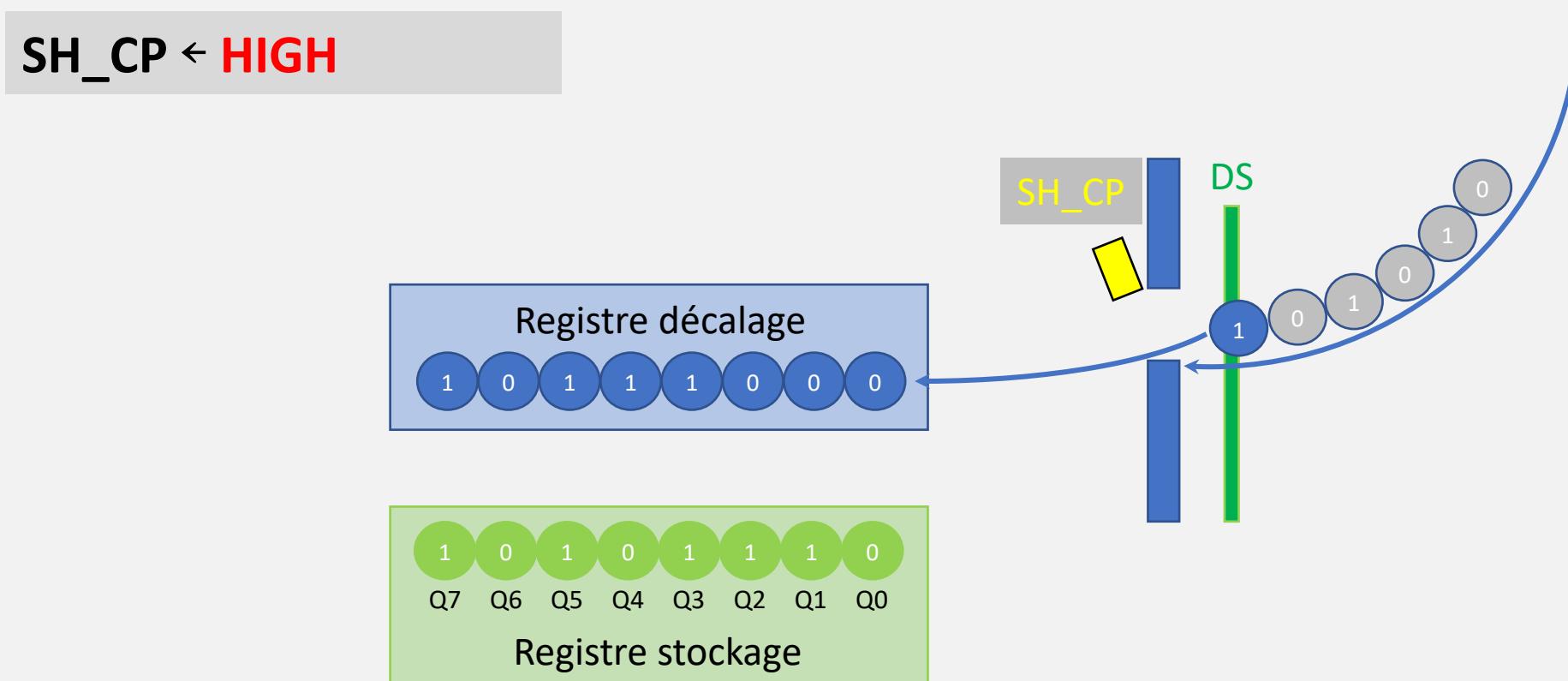
Registre à décalage – Exemple

- Envoi du nombre 42 (0b00101010)



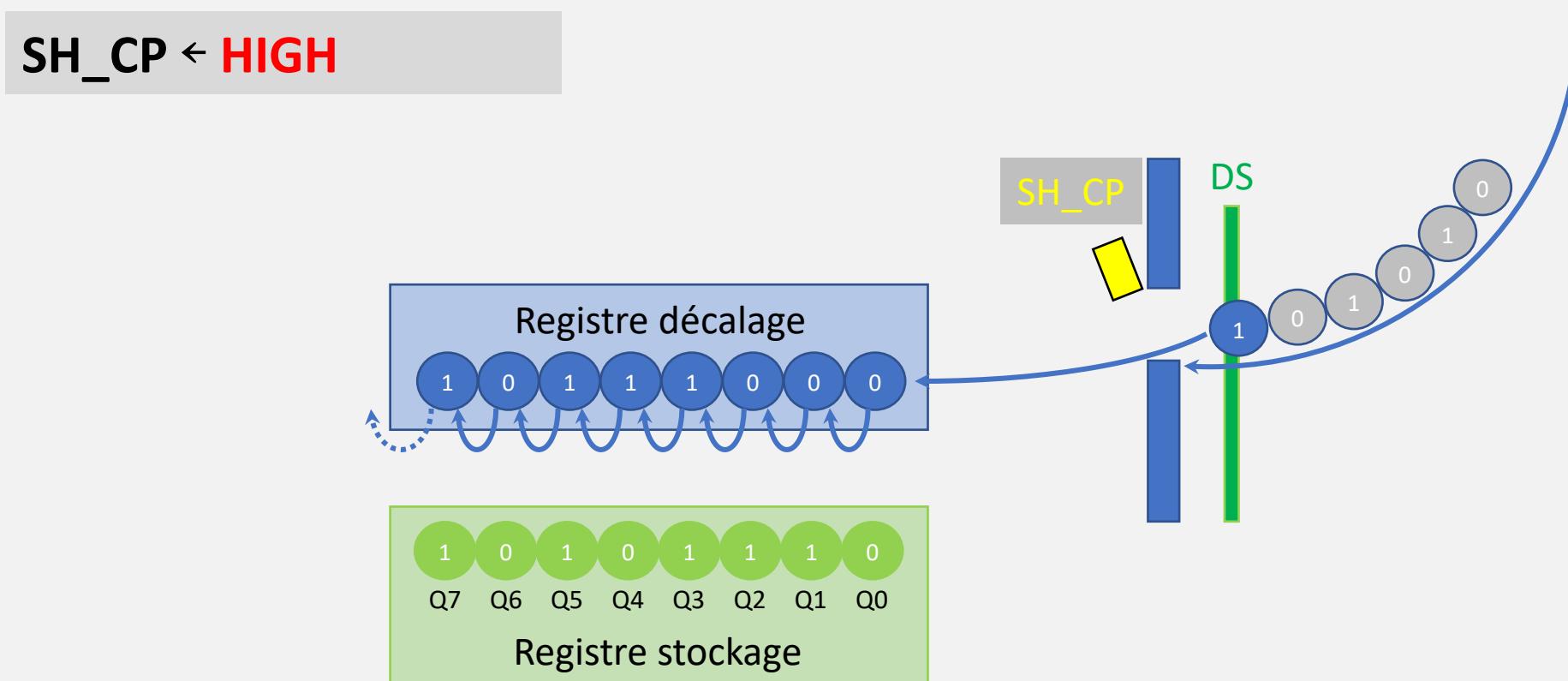
Registre à décalage – Exemple

- Envoi du nombre 42 (0b00101010)



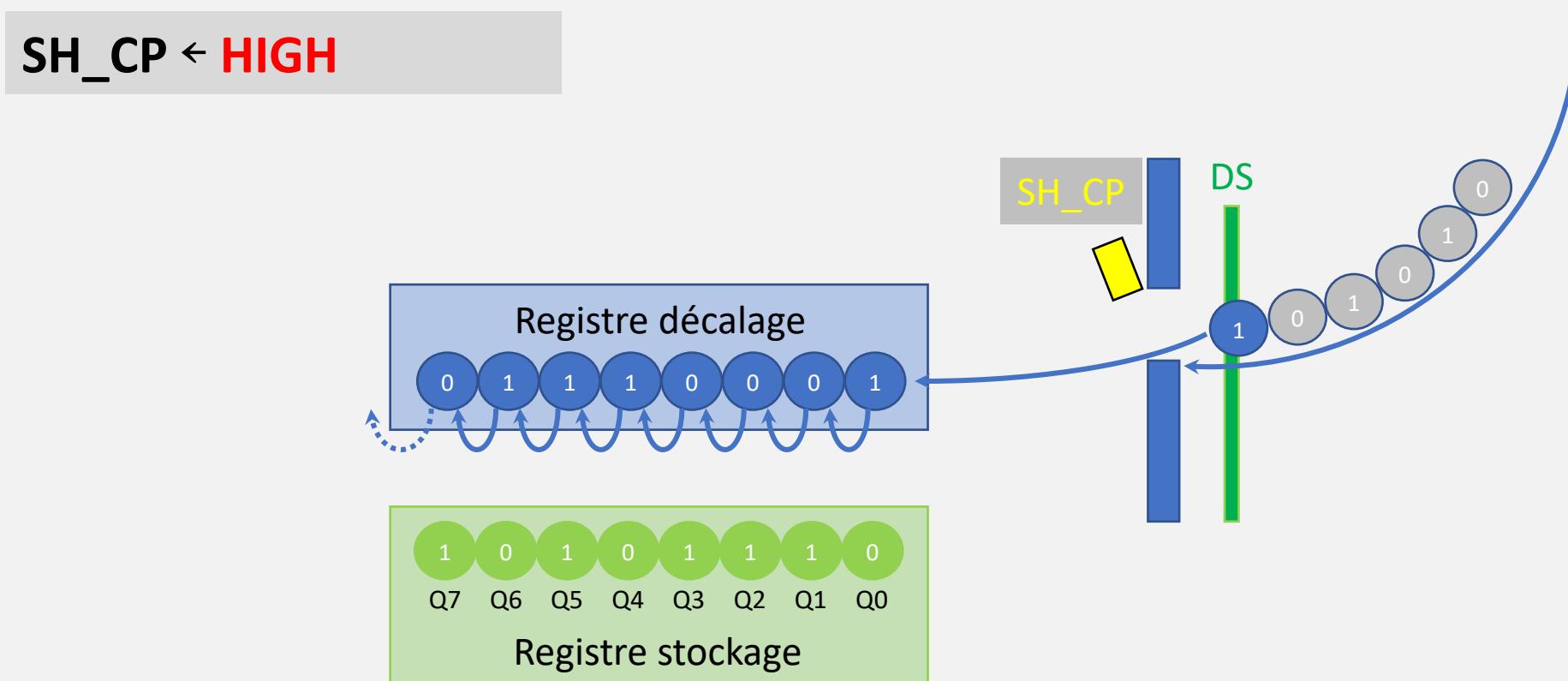
Registre à décalage – Exemple

- Envoi du nombre 42 (0b00101010)



Registre à décalage – Exemple

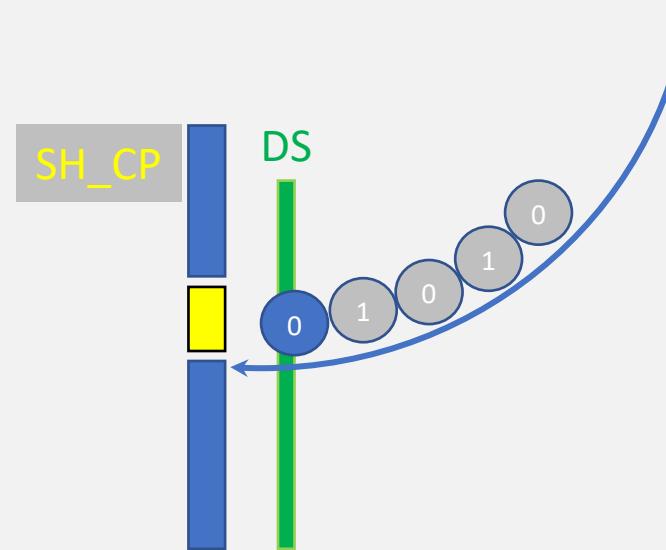
- Envoi du nombre 42 (0b00101010)



Registre à décalage – Exemple

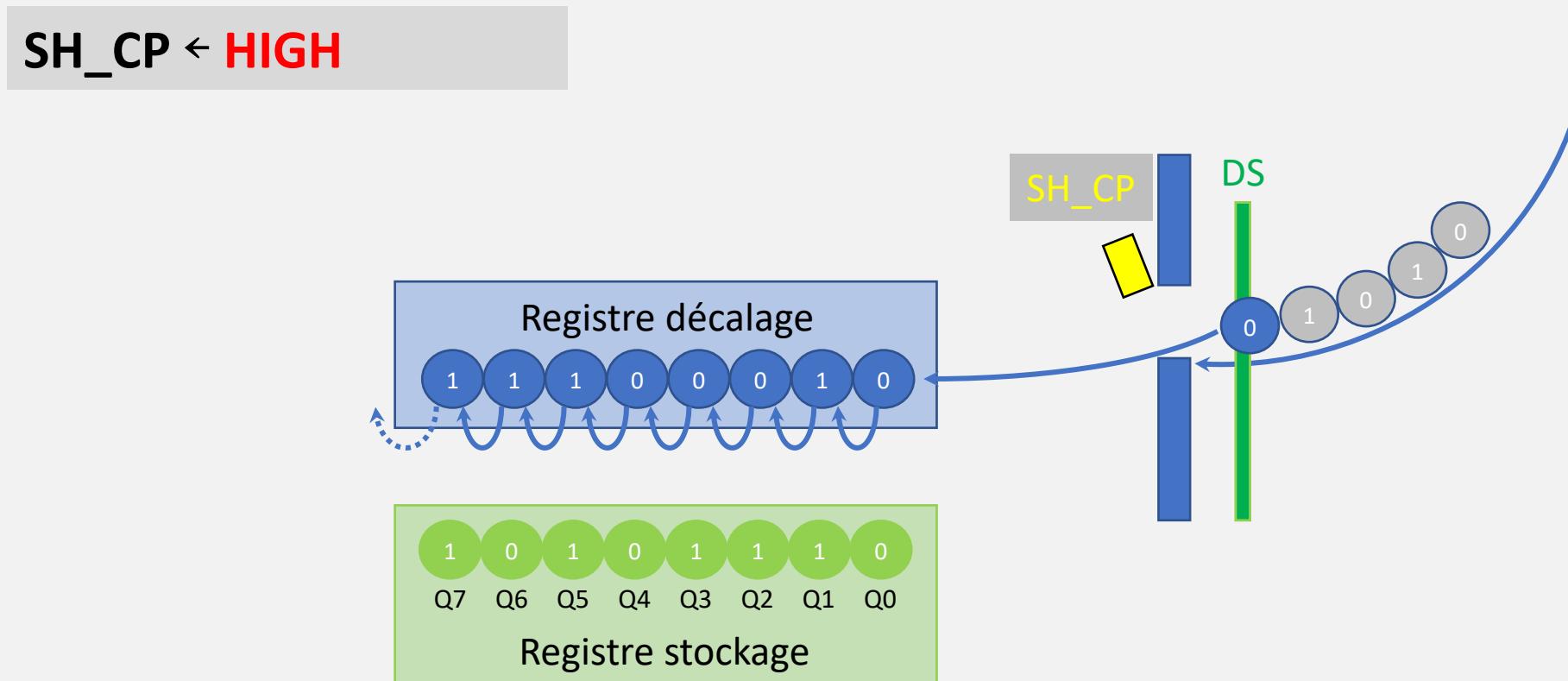
- Envoi du nombre 42 (0b00101010)

SH_CP \leftarrow LOW
DS \leftarrow LOW



Registre à décalage – Exemple

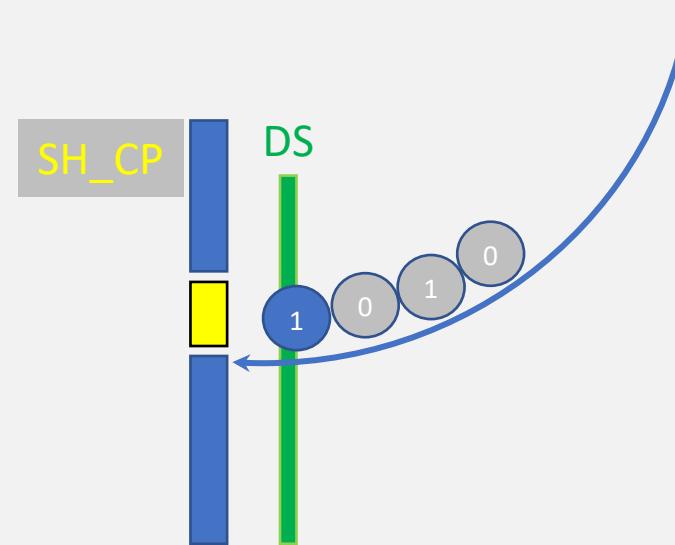
- Envoi du nombre 42 (0b00101010)



Registre à décalage – Exemple

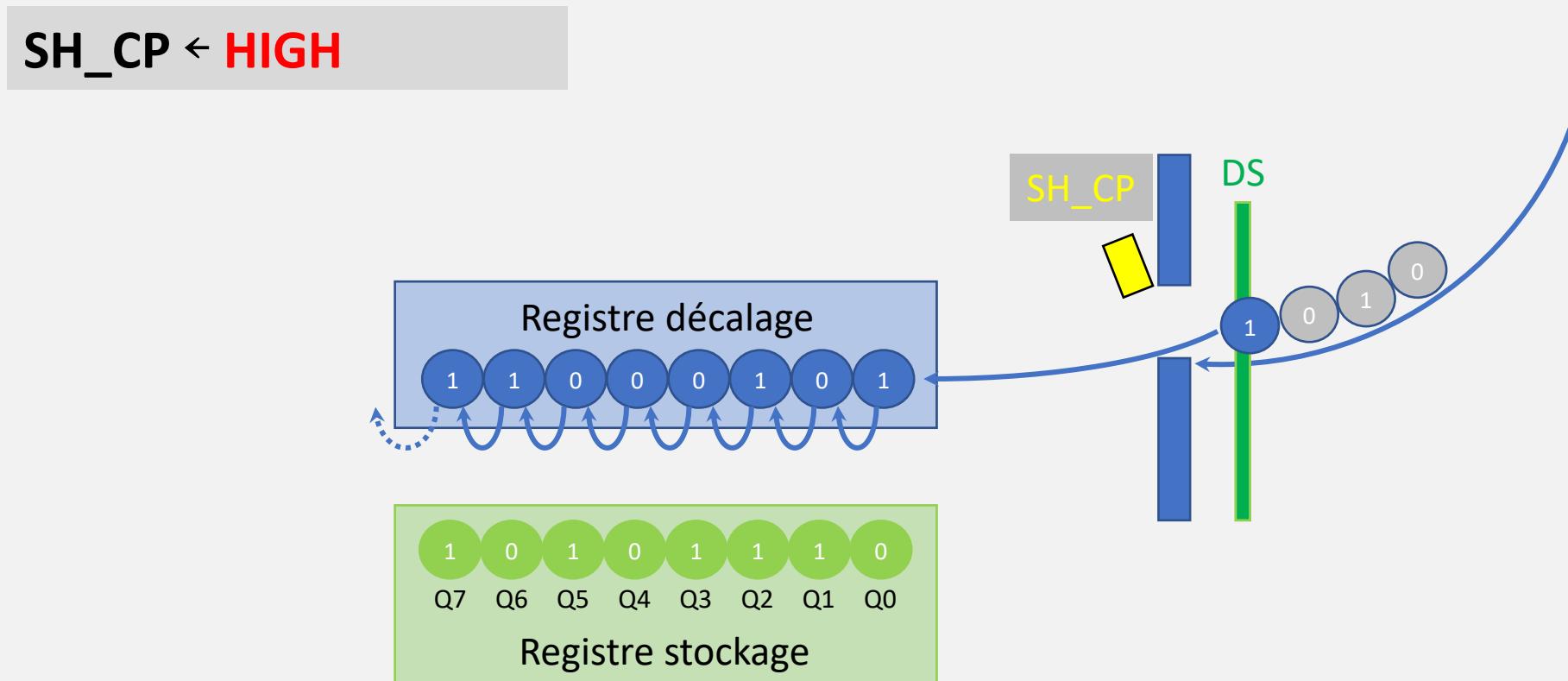
- Envoi du nombre 42 (0b00101010)

SH_CP \leftarrow **LOW**
DS \leftarrow **HIGH**



Registre à décalage – Exemple

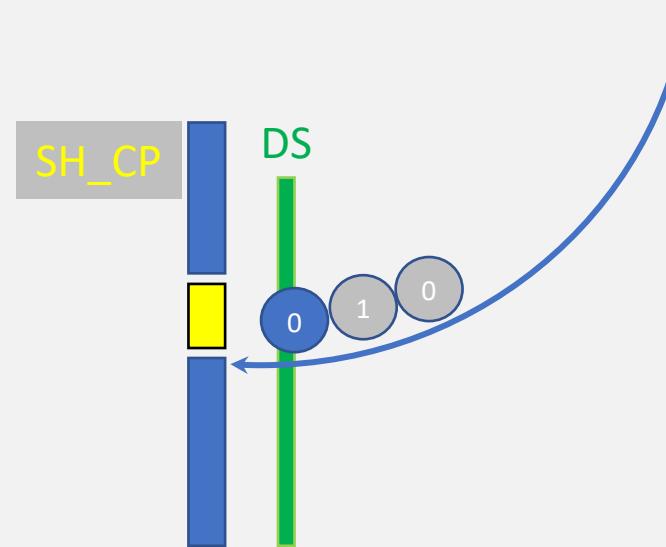
- Envoi du nombre 42 (0b00101010)



Registre à décalage – Exemple

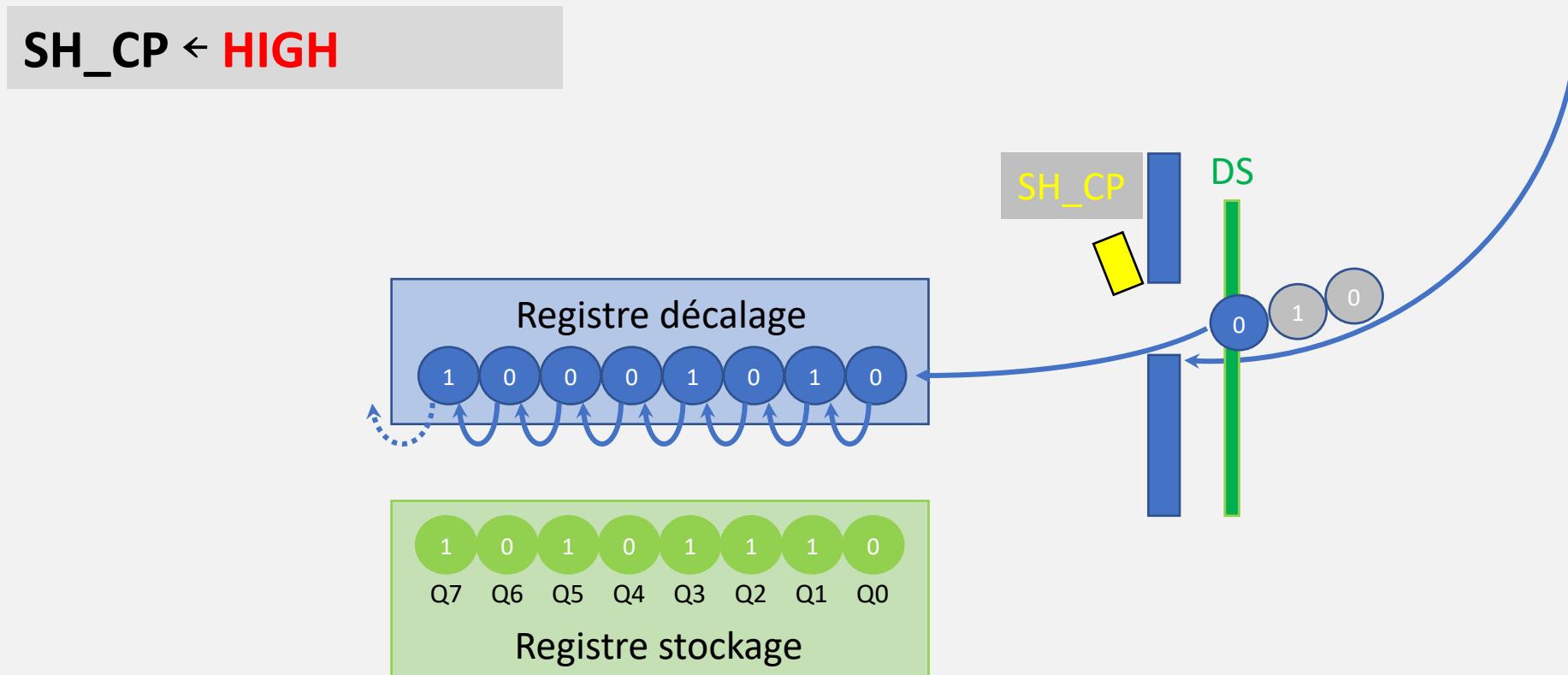
- Envoi du nombre 42 (0b00101010)

SH_CP \leftarrow LOW
DS \leftarrow LOW



Registre à décalage – Exemple

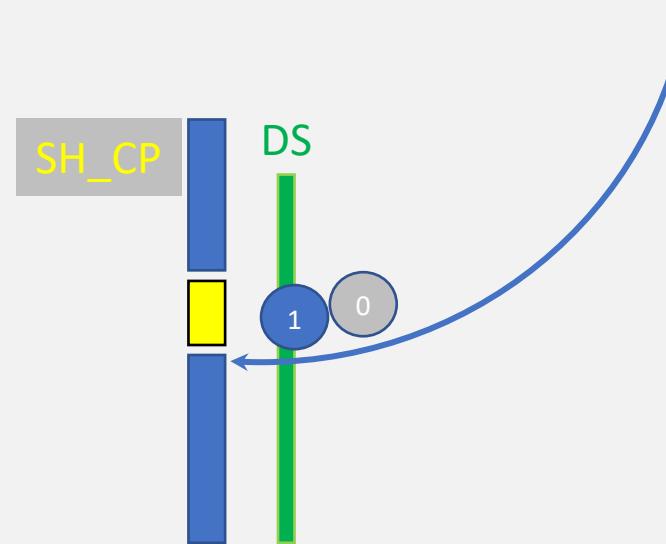
- Envoi du nombre 42 (0b00101010)



Registre à décalage – Exemple

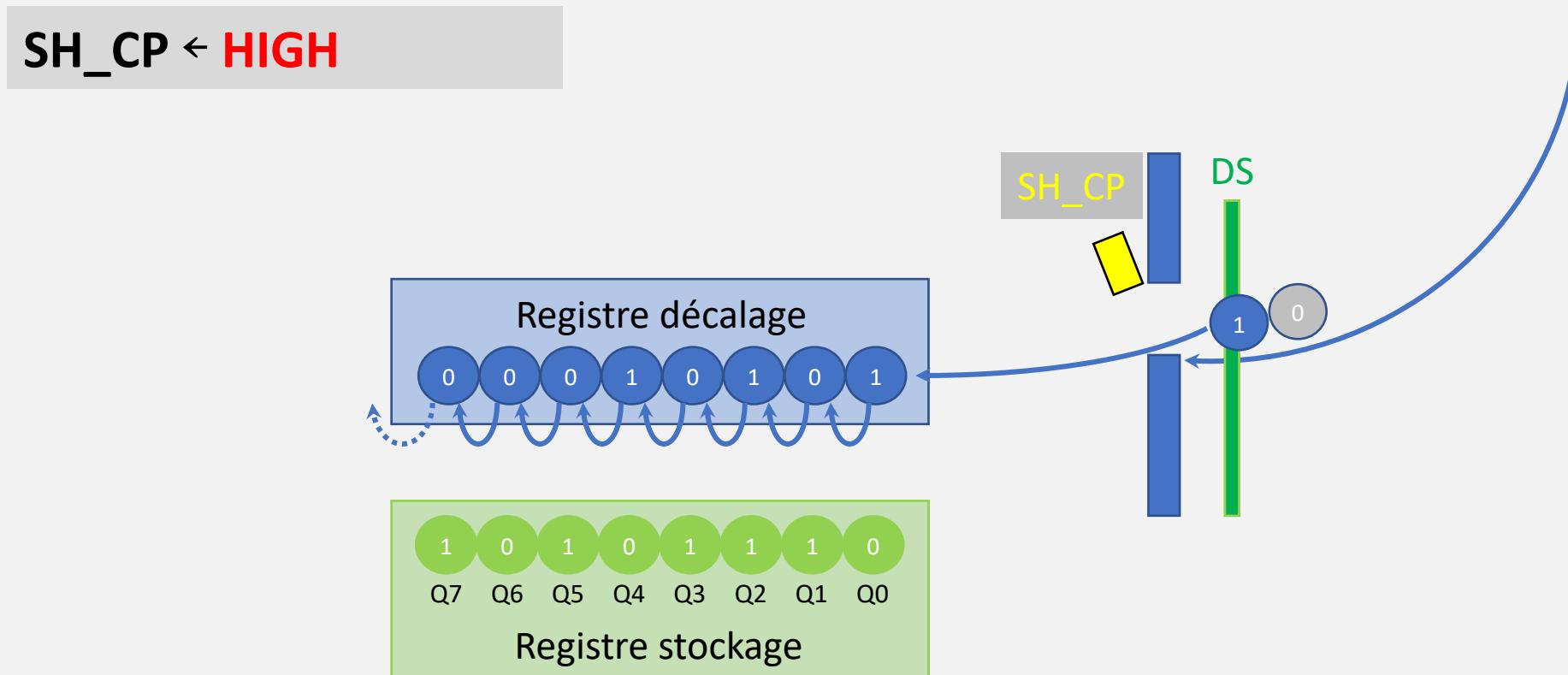
- Envoi du nombre 42 (0b00101010)

SH_CP \leftarrow **LOW**
DS \leftarrow **HIGH**



Registre à décalage – Exemple

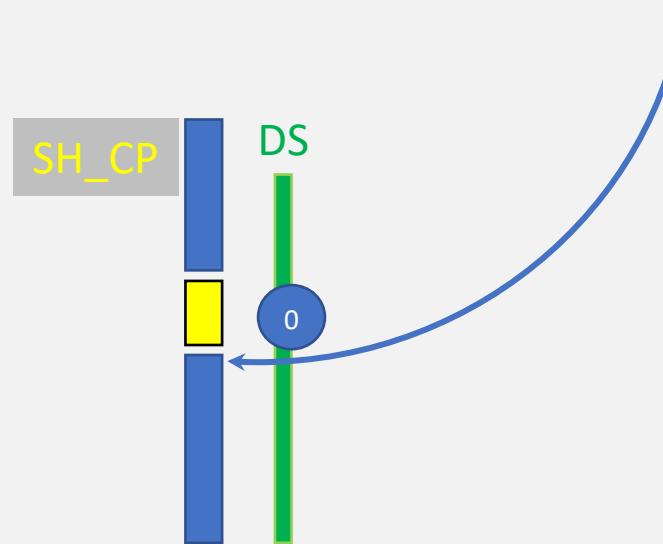
- Envoi du nombre 42 (0b00101010)



Registre à décalage – Exemple

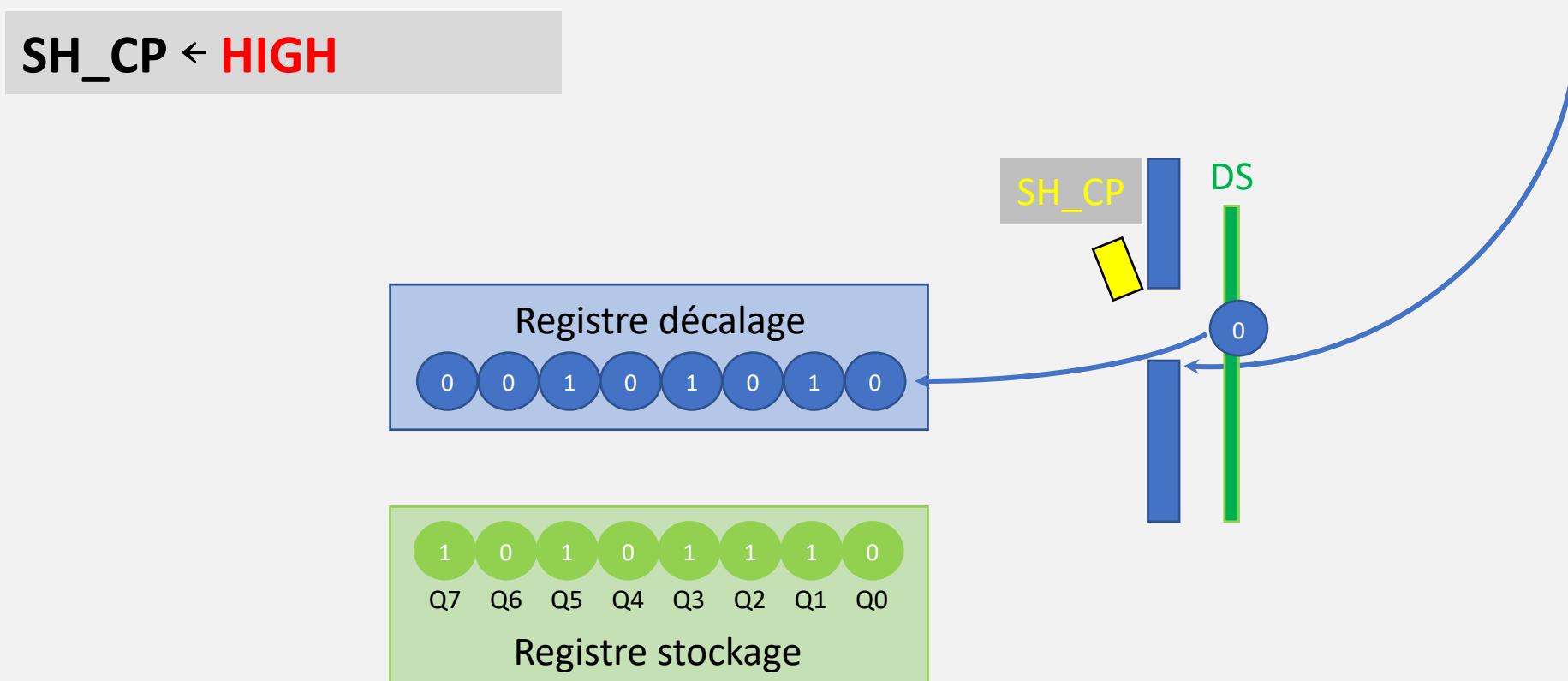
- Envoi du nombre 42 (0b00101010)

SH_CP \leftarrow LOW
DS \leftarrow LOW



Registre à décalage – Exemple

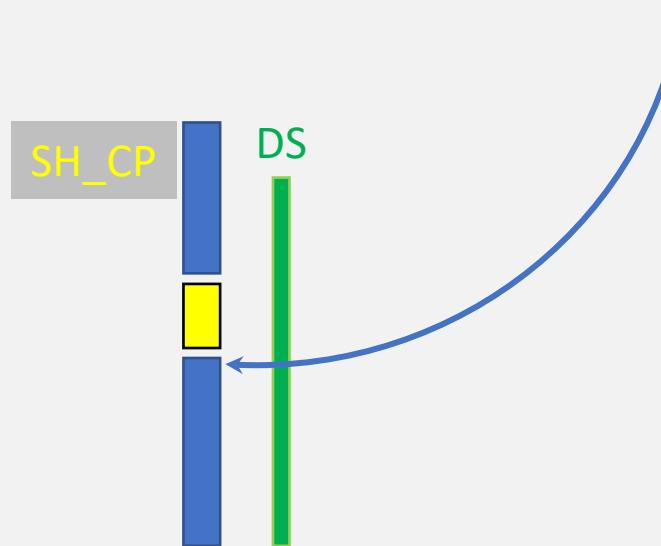
- Envoi du nombre 42 (0b00101010)



Registre à décalage – Exemple

- Envoi du nombre 42 (0b00101010)

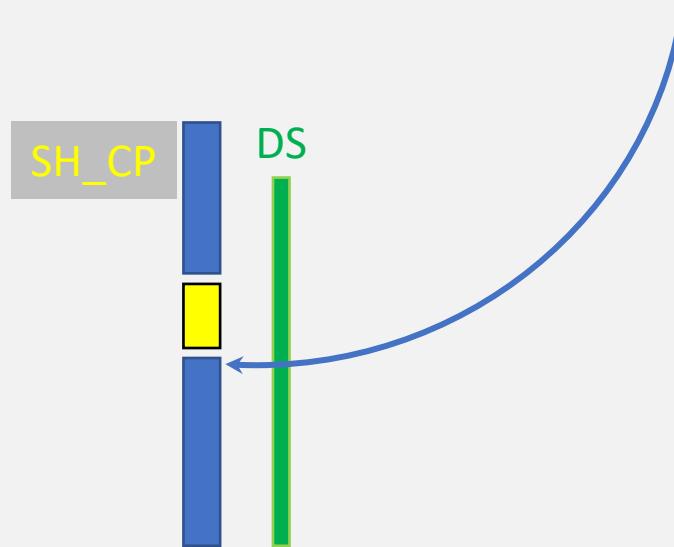
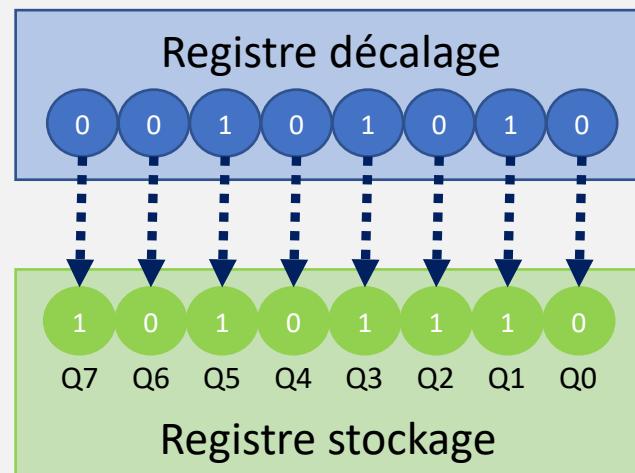
SH_CP \leftarrow LOW



Registre à décalage – Exemple

- Envoi du nombre 42 (0b00101010)

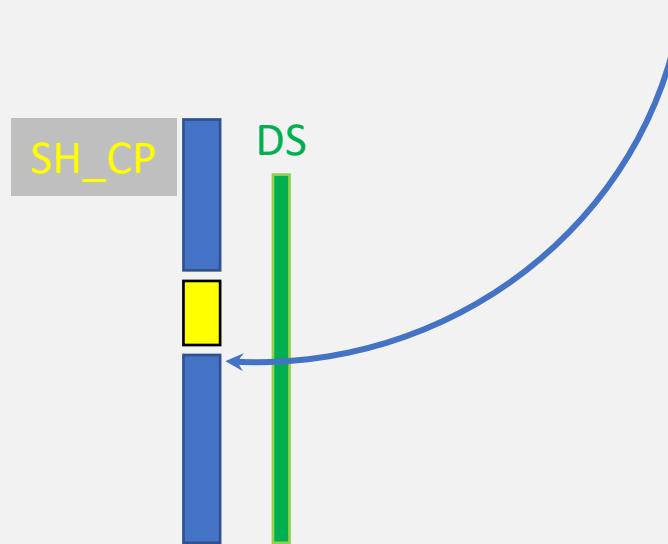
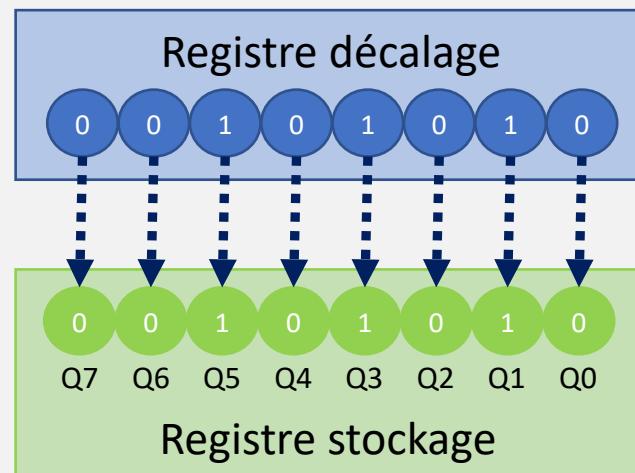
ST_CP ← HIGH



Registre à décalage – Exemple

- Envoi du nombre 42 (0b00101010)

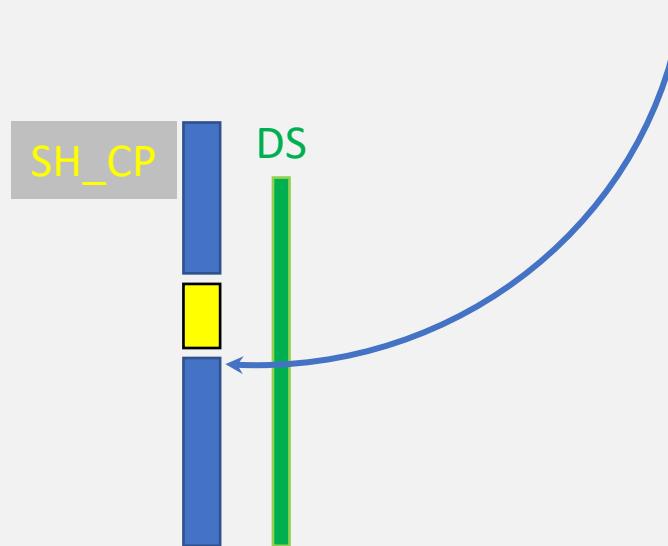
ST_CP ← HIGH



Registre à décalage – Exemple

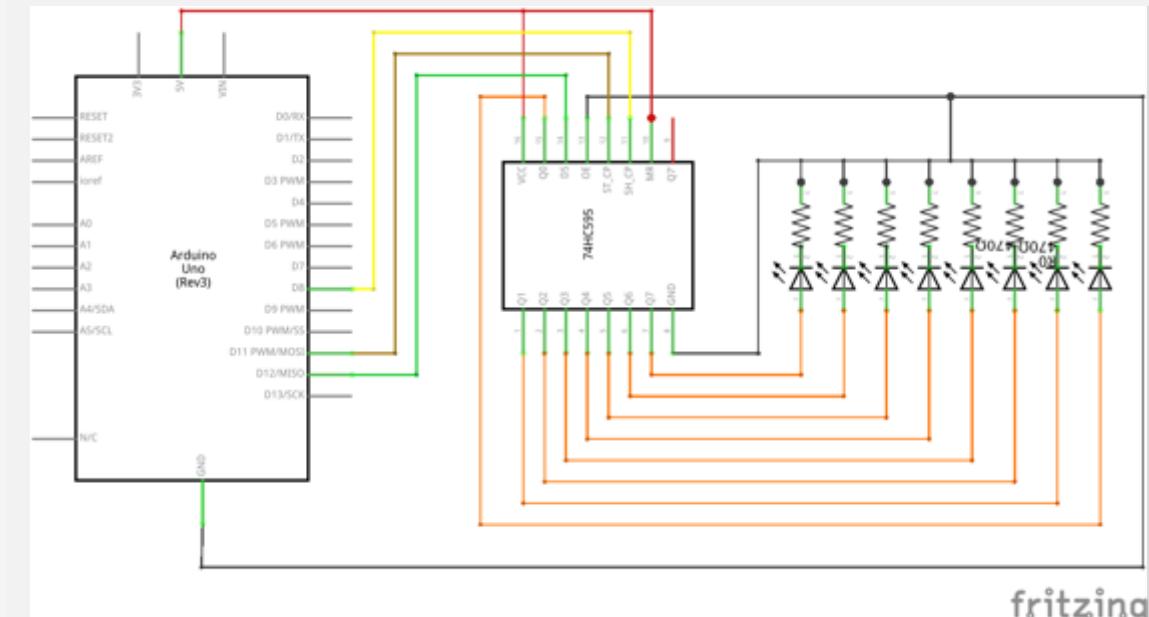
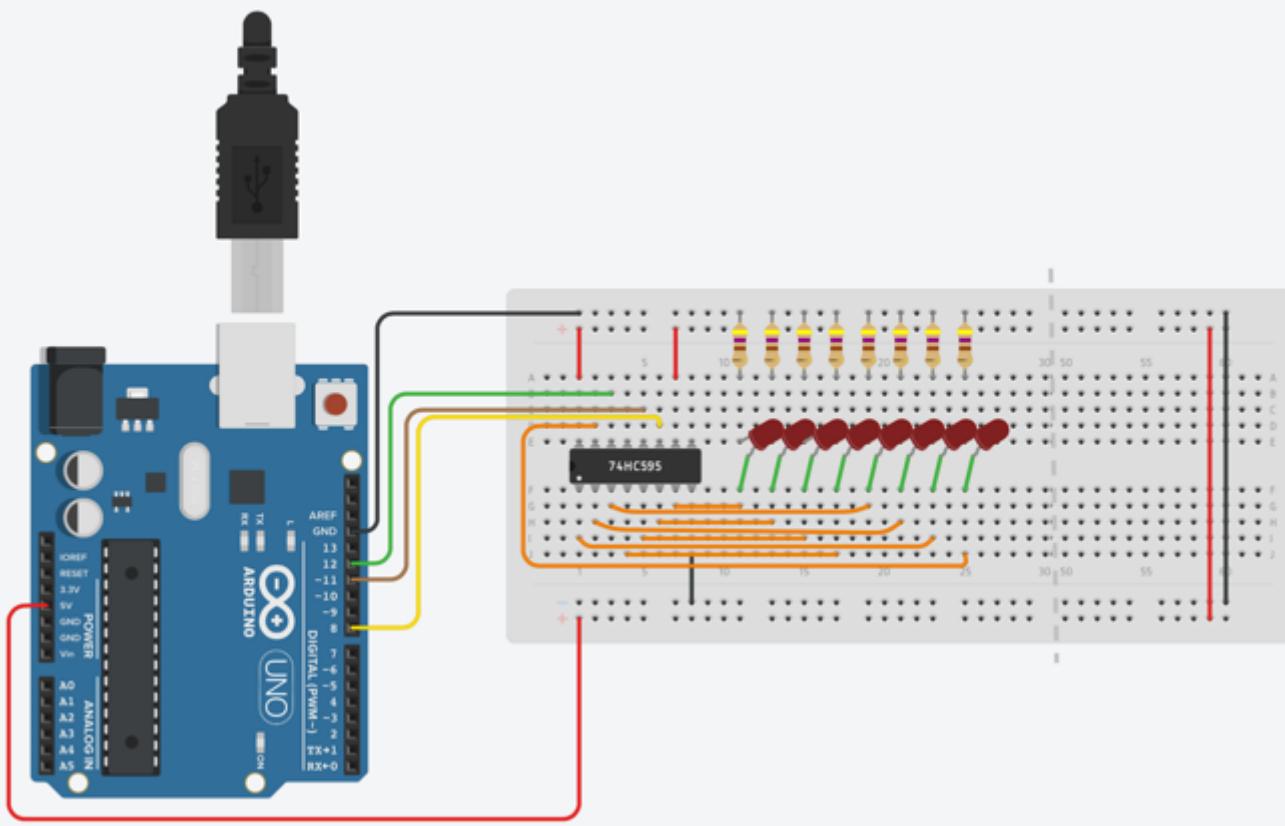
- Envoi du nombre 42 (0b00101010)

ST_CP \leftarrow LOW



Registre à décalage – Schéma 8 DELs

- Schéma du montage



fritzing

Registre à décalage – Exemple

- Envoyer 1 à Q1 et 0 à Q0
- Les précédents bits sont décalés de deux positions vers la gauche
- Les deux bits de poids fort sont perdus

```
const int borneST_CP = 11;
const int borneSH_CP = 8;
const int borneDS = 12;

void setup()
{
    pinMode(borneST_CP, OUTPUT);
    pinMode(borneSH_CP, OUTPUT);
    pinMode(borneDS, OUTPUT);

    digitalWrite(borneST_CP, LOW);

    // Envoyer 1
    digitalWrite(borneSH_CP, LOW);
    digitalWrite(borneDS, HIGH);
    digitalWrite(borneSH_CP, HIGH);

    // Envoyer 0
    digitalWrite(borneSH_CP, LOW);
    digitalWrite(borneDS, LOW);
    digitalWrite(borneSH_CP, HIGH);

    // Envoyer données du registre
    // décalage à registre stockage
    digitalWrite(borneST_CP, HIGH);
    delay(100);
    digitalWrite(borneST_CP, LOW);
}
```

Rappels – Décomposition d'un entier en bits

- M1 : la valeur reste intacte
 - Le masque se déplace de gauche à droite
- M2 : la valeur est détruite
 - La valeur se déplace de droite à gauche

```
// M1
int valeur = 42;
for (int i = 7; i >= 0; --i) {
    Serial.print((valeur >> i) & 0x01);
}
Serial.println();
```

```
// M2
for (int i = 0; i < 8; ++i) {
    Serial.print((valeur & 0x80) != 0);
    valeur <<= 1;
}
Serial.println();
```