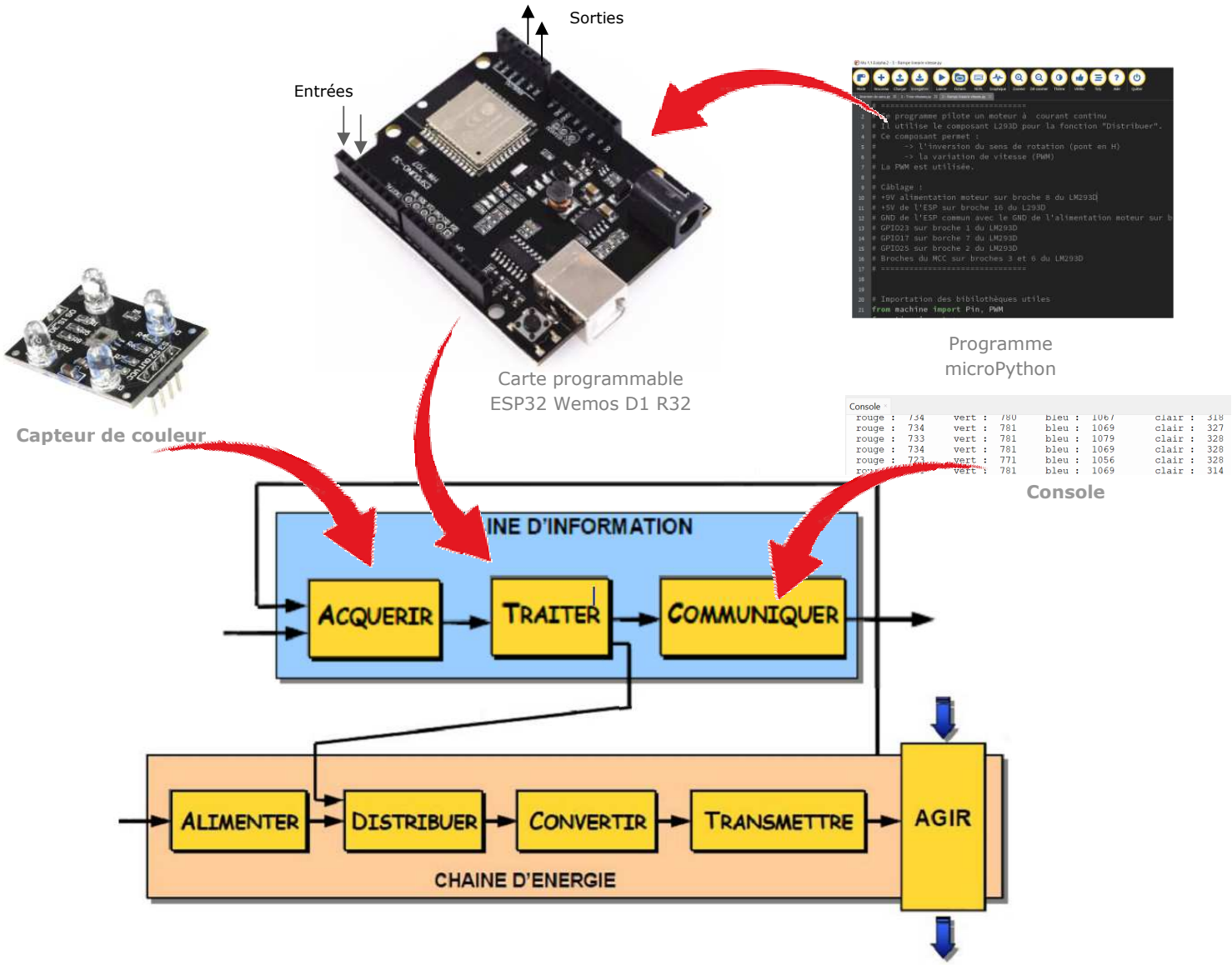
MISE EN ŒUVRE

→ **TRAITER : ESP32 WEMOS (EDI MU)**

→ **ACQUERIR : Capteur de couleur HW-067 - TCS230**

1 – Mise en situation



2 – Câblage / Montage

Raccorder la carte ESP au capteur de couleur en respectant :

ESP32	Capteur	Elément contrôlé
GPIO17	S0	Détermine la fréquence du signal renvoyé
GPIO16	S1	
GPIO27	S2	Détermine la couleur mesurée
GPIO14	S3	
NC	/OE	Non connectée
GPIO19	OUT	Renvoi un signal carré de largeur variable en fonction de l'intensité de la couleur

3 – Principe de fonctionnement de la télécommande IR et du récepteur IR

Le capteur est utilisé pour détecter la couleur d'un objet à une distance maximale de 10 mm.
Il est équipé de quatre diodes électroluminescentes (DEL) qui éclairent l'emplacement de la mesure.

Le module comporte 8 connecteurs :
VCC : +3,3V
GND : 0V

S0 et S1 : sélection de l'échelle de fréquence

S0 =1 et S1 = 1: 100% de la fréquence (environ 600 kHz)
S0 = 1 et S1 = 0: 20% de la fréquence (environ 120 kHz)
S0 = 0 et S1 = 1: 2% de la fréquence (environ 12 kHz)
S0 = 0 et S1 = 0: capteur inactif

S2 et S3: sélection des photodiodes actives

S2 = 0 et S3 = 0 : Rouge
S2 = 0 et S3 = 1: Bleu
S2 = 1 et S3 = 1: Vert
S2 = 1 et S3 = 0: Sans filtre

OUT: signal de sortie
Le signal de sortie est une onde carrée dont la fréquence est proportionnelle à l'intensité lumineuse désirée.

4 – Programme

ESP32 Micropython programme qui renvoie dans la console les 3 valeurs mesurées pour chaque couleur et une valeur pour la clareté.

```
1  ##############################################################
2  #             Programme de gestion du capteur de couleur     #
3  #             S0 sur broche 17 - S1 sur borche 16            #
4  #             S2 sur broche 27 - S3 sur borche 14            #
5  #             OUT sur broche 19 - EO non câblé               #
6  #      Renvoie une valeur. Plus la valeur est importante     #
7  #            moins la couleur mesurée est présente           #
8  #     Voir : https://www.youtube.com/watch?v=9NToIzM6cVE     #
9  #      CG 9/3/23                                             #
10 ##############################################################
11
12 # importation des bibliothèques ESP32
13 from machine import Pin
14 import time
15
16 # définition des broches utilisées
17 broche_S0=17
18 broche_S1=16
19 broche_S2=27
20 broche_S3=14
21 broche_OUT=19 # Attention on l'utilise pour une interruption
22
23 pS0=Pin(broche_S0,Pin.OUT)
24 pS1=Pin(broche_S1,Pin.OUT)
25 pS2=Pin(broche_S2,Pin.OUT)
26 pS3=Pin(broche_S3,Pin.OUT)
27 pSOUT=Pin(broche_OUT,Pin.IN)
28
29 #réglage de la fréquence de sortie du capteur
30 pS0.off()
31 pS1.on()
32
33 # définition de la fonction qui mesure les couleur
34 def mesure(): # cherche la durée entre 2 fronts montants
35     delta=0
36     while pSOUT.value()==1: #attend que l'entrée SOUT passa à 0
37         pass
38     while pSOUT.value()==0: # attend que l'entrée SOUT passe à 1 => Front montant
39         pass
40     start=time.ticks_us() # prise de temps machine (en µs)
41     while pSOUT.value()==1: # attend que l'entrée SOUT passe à 0
42         pass
43     while pSOUT.value()==0: # attend que l'entrée SOUT passe de nouveau à 1 => Front montant
44         pass
45     stop = time.ticks_us() # prise de temps machine
46     delta = time.ticks_diff(stop,start) # Calcule la différence de temps entre les 2 fronts montants de SOUT
47     return delta # renvoie le temps entre les 2 fronts montants
48
49
50 while(True):
51     pS2.off()
52     pS3.off()
53     rouge=mesure()
54     time.sleep_ms(100)
55
56     pS2.off()
57     pS3.on()
58     bleu=mesure()
59     time.sleep_ms(100)
60
61     pS2.on()
62     pS3.on()
63     vert=mesure()
64     time.sleep_ms(100)
65
66     pS2.on()
67     pS3.off()
68     clair=mesure()
69
70     print("rouge : ",rouge, "    vert : ",vert, "    bleu : ", bleu, "      clair : ",clair)
71     time.sleep_ms(700)
```