StackedArhat

1. Introduction to gameplay

The purpose of this experiment is exactly the opposite of the previous lesson "Nature Porter". It is to pick up the building blocks from different sides in the order of yellow, red, green and blue and stack them into the gray area in the middle.

The way to place the building blocks is as shown in the figure below:

After executing the code, the robot arm will stack the building blocks, and the final effect is as shown in the figure below:

2. Code content

Code path:/root/Dofbot/3.ctrl_Arm/11.heap_up.ipynb

```python
#!/usr/bin/env python3
#coding=utf-8
import time
from Arm_Lib import Arm_Device
#Create a robotic arm object
Arm = Arm_Device()
time.sleep(.1)
```

```python
# Define the function of clamping building blocks, enable=1: clamp, =0: release
def arm_clamp_block(enable):
if enable == 0:
Arm.Arm_serial_servo_write(6, 60, 400)
else:
Arm.Arm_serial_servo_write(6, 130, 400)
time.sleep(.5)

# Define the mobile robot arm function and control the movement of servos No. 1-5
at the same time, p=[S1, S2, S3, S4, S5]
def arm_move(p, s_time = 500):
    for i in range(5):
id=i+1
    if id == 5:
    time.sleep(.1)
    Arm.Arm_serial_servo_write(id, p[i], int(s_time*1.2))
```

```python
    elif id == 1 :
      Arm.Arm_serial_servo_write(id, p[i], int(3*s_time/4))
      else:
      Arm.Arm_serial_servo_write(id, p[i], int(s_time))
      time.sleep(.01)
      time.sleep(s_time/1000)
```

```python
# Define variable parameters at different locations
p_mould = [90, 130, 0, 0, 90]
p_top = [90, 80, 50, 50, 270]
p_layer_4 = [90, 76, 40, 17, 270]
p_layer_3 = [90, 65, 44, 17, 270]
p_layer_2 = [90, 65, 25, 36, 270]
p_layer_1 = [90, 48, 35, 30, 270]
p_Yellow = [65, 22, 64, 56, 270]
p_Red = [118, 19, 66, 56, 270]
p_Green = [136, 66, 20, 29, 270]
p_Blue = [44, 66, 20, 28, 270]
```

```python
# Let the robotic arm move to a position ready to grab
arm_clamp_block(0)
arm_move(p_mould, 1000)
time.sleep(1)
```

```python
# Clamp the blocks in the yellow area and stack them to the bottom position in
the middle.
arm_move(p_top, 1000)
arm_move(p_Yellow, 1000)
arm_clamp_block(1)
arm_move(p_top, 1000)
arm_move(p_layer_1, 1000)
arm_clamp_block(0)
time.sleep(.1)
arm_move(p_mould, 1100)

# time.sleep(1)
```

```python
# Clamp the blocks in the red area and stack them to the second layer in the
middle.
arm_move(p_top, 1000)
arm_move(p_Red, 1000)
arm_clamp_block(1)
arm_move(p_top, 1000)
arm_move(p_layer_2, 1000)
arm_clamp_block(0)
time.sleep(.1)
arm_move(p_mould, 1100)
# time.sleep(1)
```

```python
# Clamp the blocks in the green area and stack them to the third layer in the
middle.
arm_move(p_top, 1000)
arm_move(p_Green, 1000)
arm_clamp_block(1)
arm_move(p_top, 1000)
arm_move(p_layer_3, 1000)
arm_clamp_block(0)
time.sleep(.1)
arm_move(p_mould, 1100)

# time.sleep(1)
```

```python
# Clamp the blocks in the blue area and stack them to the fourth layer in the
middle.
arm_move(p_top, 1000)
arm_move(p_Blue, 1000)
arm_clamp_block(1)
arm_move(p_top, 1000)
arm_move(p_layer_4, 1000)
arm_clamp_block(0)
time.sleep(.1)
arm_move(p_mould, 1100)

# time.sleep(1)
```

```python
del Arm # Release the Arm object
```