Two DOFBOT Synchronized Actions

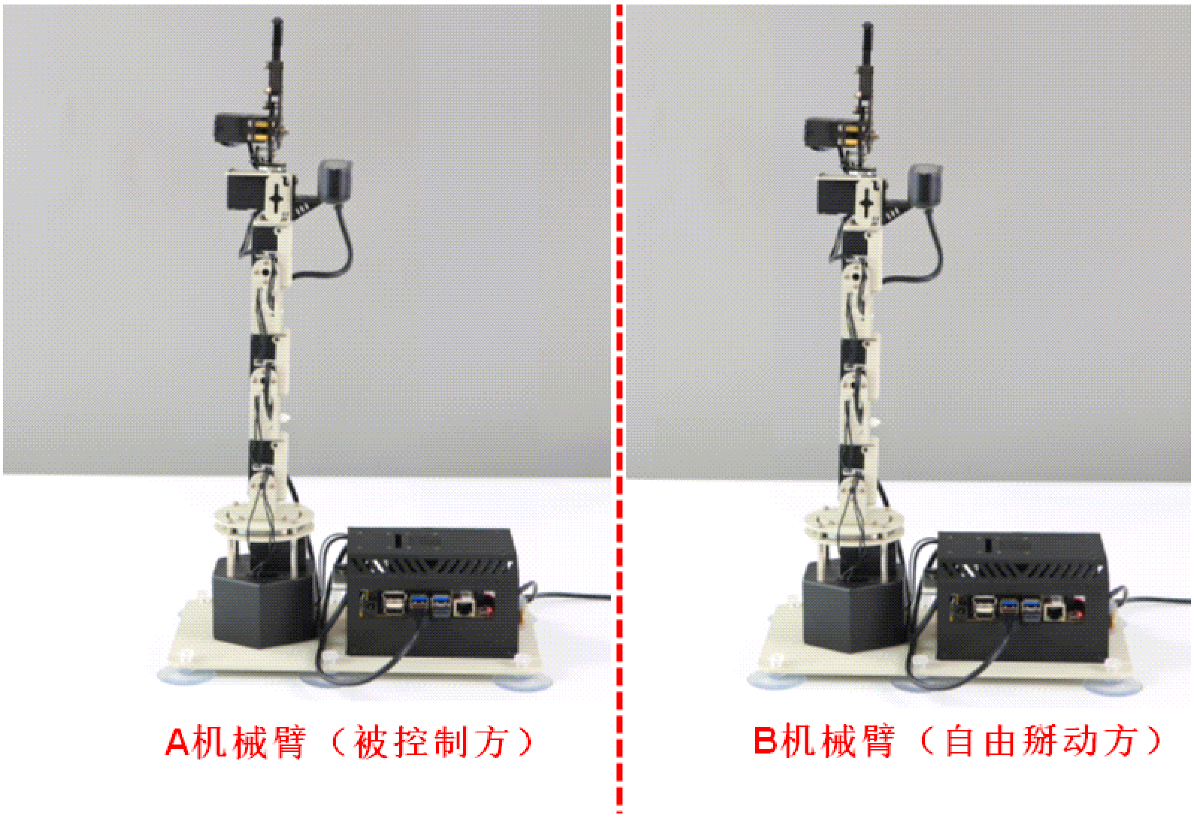
Note: This experiment requires two Dofbot Pro.

12.1 Communication Principle

The principle of this experiment is to use one Dofbot Pro as a server, starting a TCP network service to receive data and execute operations; while the other Dofbot Pro acts as a client, also known as the controller, reading the angle values of all servos sequentially and immediately sending all servo angles to the server via TCP service. At this time, the client continuously reads servo angle values and sends them to the server. The server receives and sets the angles to immediately update all servos, achieving remote synchronized action between the two robotic arms.

12.2 Pre-Experiment Preparation

1. This experiment involves two Dofbot Pro with identical hardware. Except for the different programs running, everything else can be the same. If the Dofbot Pro are using the factory system image, please first stop the auto-starting factory program before conducting this experiment.
2. Ensure both Dofbot Pro are connected to the same local area network, i.e., connected to the same router (wired or wireless connection is acceptable), and can communicate within the local area network.
3. Since this experiment involves two Dofbot Pro, to easily distinguish them, we call the Dofbot Pro acting as the server "Arm A" (the controlled side), and the Dofbot Pro acting as the controller "Arm B" (the freely movable side). Since Arm A's movement depends on Arm B's motion, to avoid excessive movement range of Arm A when starting the program, please adjust both Arm A and Arm B to an upright vertical position before starting the program.



12.3 Program Code

[Server Code: Arm A]

For specific code content, please refer to the following path:

```
~/dofbot_pro/dofbot_ctrl/scripts/12.sync_movement/A_arm_follow.py
```

[Client Code: Arm B]

For specific code content, please refer to the following path:

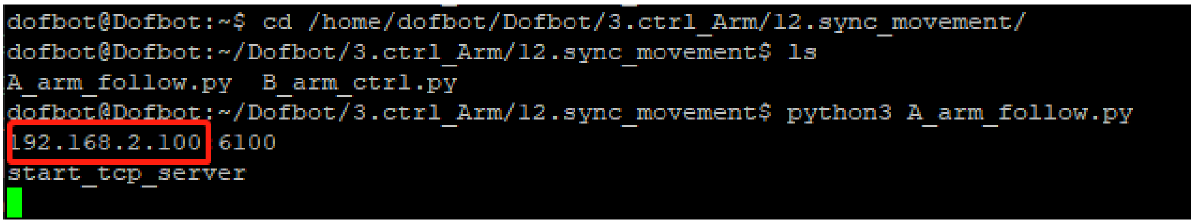
```
~/dofbot_pro/dofbot_ctrl/scripts/12.sync_movement/B_arm_ctrl.py
```

12.4 Experimental Operation and Purpose

Experimental Purpose: Use Arm B to remotely control Arm A.

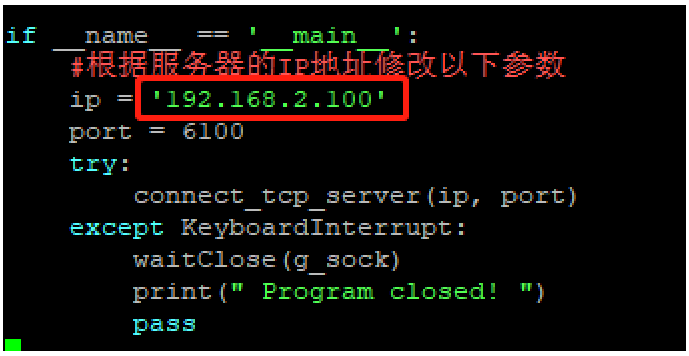
1. Open the terminal of Arm A and run the A_arm_follow.py program. Then note down the printed IP address.

```
cd ~/dofbot_pro/dofbot_ctrl/scripts/12.sync_movement/
python3 A_arm_follow.py
```



```
dofbot@Dofbot:~$ cd /home/dofbot/Dofbot/3.ctrl_Arm/12.sync_movement/
dofbot@Dofbot:~/Dofbot/3.ctrl_Arm/12.sync_movement$ ls
A_arm_follow.py  B_arm_ctrl.py
dofbot@Dofbot:~/Dofbot/3.ctrl_Arm/12.sync_movement$ python3 A_arm_follow.py
192.168.2.100 6100
start_tcp_server
```

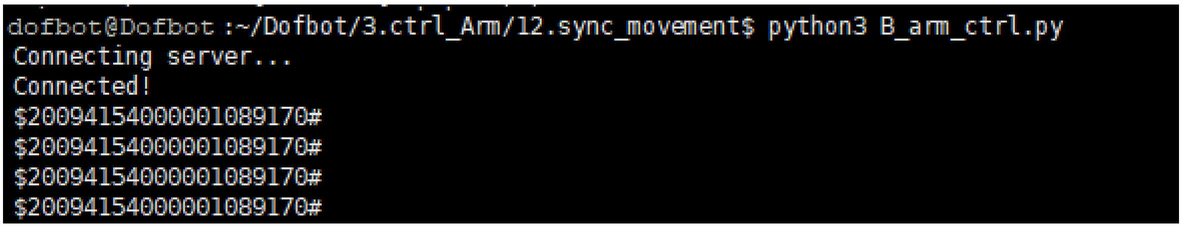
2. Open the terminal of Arm B and replace the IP address recorded in the previous step in the ip address position in B_arm_ctrl.py.



```
if __name__ == '__main__':
    #根据服务器的IP地址修改以下参数
    ip = '192.168.2.100'
    port = 6100
    try:
        connect_tcp_server(ip, port)
    except KeyboardInterrupt:
        waitClose(g_sock)
        print(" Program closed! ")
        pass
```

3. After modification, save and exit, then run the B_arm_ctrl.py program.

```
cd ~/dofbot_pro/dofbot_ctrl/scripts/12.sync_movement/
python3 B_arm_ctrl.py
```



```
dofbot@Dofbot:~/Dofbot/3.ctrl_Arm/12.sync_movement$ python3 B_arm_ctrl.py
Connecting server...
Connected!
$20094154000001089170#
$20094154000001089170#
$20094154000001089170#
$20094154000001089170#
```

Note: You must run Arm A's A_arm_follow.py program first, then run Arm B's B_arm_ctrl.py program. Arm B's terminal will print connection information and sent data, while communication between the two robotic arms officially begins.

At this time, you can see that Arm A's state is the same as Arm B's state, and you can modify Arm A's state by moving Arm B's posture. Through this TCP transmission command, the effect of synchronized action between two robotic arms can be achieved. Arm A's control has angle limitations, while Arm B's free movement has no angle limitations. If Arm B's servo 1 is moved beyond 180 degrees, Arm A will stay at 180 degrees until Arm B's angle is adjusted back to a reasonable range before continuing to move.