Manual Book

for Transfer Stage

Produced by: Ce Zheng

czheng563@outlook.com

If you come across any problems, please contact me at czheng563@outlook.com

Happy Researching!

Table of Contents

1 **Brief Introduction to Transfer Stage** 3

2 **Safety Instructions** 4
- 2.1 Basic Notes 4
- 2.2 Warnings 4
- 2.3 Suggestions 5

3 **Components and Functions** 6
- 3.1 Main Components 6
- 3.2 Microscope 7
  - 3.2.1 Nikon microscope 8
  - 3.2.2 Canon DSLR camera(2b) 8
  - 3.2.3 Light controller(2a) 8
  - 3.2.4 Objective lens and automatically controlled objective disk(2c) 8
- 3.3 Heating system 9
- 3.4 Motors 9
- 3.5 Computers and joystick 10
- 3.6 Others 10

4 **Assembly Guidelines** 11
- 4.1 Preparation 11
- 4.2 Exact process 11
  - 4.2.1 Assembling of stage 11
  - 4.2.2 Assembling of microscope 12
  - 4.2.3 Assembling of heating device 12
  - 4.2.4 Assembling of motors 12
  - 4.2.5 Assembling of other devices 14
- 4.3 Advanced settings: In glove box 14

5 **Guidance for use** 16
- 5.1 Process for scanning samples 16
- 5.2 Process for transferring samples 16
- 5.3 Special Process 17

6 **Common Problems and Solutions** 18
- 6.1 Related to Stage & Motors 18
- 6.2 Related to Microscope 19
- 6.3 Related to Heating device 19
- 6.4 Related to software 19

7 Appendix **21**
7.1 Matlab Code 21
7.2 Shopping List 27

About This File

This document was prepared by Peter Ce ZHENG for use by all iLab for Quantum Matters and Devices members.
It is confidential and should not be distributed.

1

SECTION

Brief Introduction to Transfer Stage

In recent years, with the rise of van der Waals materials research, the study and characterization of 2D materials has gradually become a focus of attention. Generally speaking, it is difficult to fabricate 2D materials with different thicknesses, and we usually tear single-crystal samples by tape, and then find their locations through microscope to grasp and transfer them. Typically, this process is accomplished using a transfer stage, which is a very expensive piece of equipment (www. hqgraphene.com), often costing $25,000. By modifying the original structure, we were able to achieve both fully automated operation and a lower price.

Motor-controlled 2D material transfer platforms offer several advantages. These platforms provide precise control, ensuring accurate positioning during the transfer process and minimizing material damage and errors. Additionally, they enable automation, allowing for the autonomous completion of transfers, increasing efficiency, reducing labor costs, and minimizing human errors. The multi-axis control capability of these systems enables precise transfers in various orientations to meet different application requirements. Furthermore, their high programmability allows for easy adjustment of motion and parameter settings, enhancing system intelligence and customization. In summary, motor-controlled 2D material transfer platforms provide an efficient, accurate, and reliable solution for transferring 2D materials.

SECTION 2

Safety Instructions

When operating the full platform, please read the following instructions to ensure the safety of the lab equipment and the operator.

2.1 Basic Notes

1. This product is not for consumption
2. The platform should be placed on a level surface and secured to prevent unnecessary injury or death to persons and property due to overturning.
3. Please purchase the plug-in board according to the appropriate power supply.

2.2 Warnings

1. Objective lenses are very valuable equipment and must not be touched directly during installation.
2. Do not attempt to turn the motor with your bare hands when the motor power is on to prevent damage to motor related parts.
3. When it is not possible to use the motor to move the rotary table, be sure to test the position of the rotary table to prevent causing the screw to slip.
4. When using a heater (24V power supply), increase the voltage in **steps of 5V** to a maximum of 24V. Also, monitor the temperature detected by the thermocouple in real time to prevent a fire.

5. Pay attention to the wiring connections to prevent short-circuiting by touching between metal wires.

6. If the vacuum pump function is used in the glove box, care must be taken to stabilize the air pressure to prevent air from being sucked back into the glove box.

7. When all experiments are completed, be sure to turn off all power to prevent the motor from overheating and burning.

2.3 Suggestions

1. Check the wiring connections and the stability of the platform before turning it on.

2. Turn off the power promptly after each use.

3. Save the operation code and do not modify the code easily.

4. Perform regular device testing and maintenance to prevent accidents.

5. When observing samples, use low magnification first and then high magnification.

SECTION 3

Components and Functions

The following provides you with overall view of the whole parts of device. It can be divided into Five parts: Main operating stages, Microscope, Heating device, Computer& Joystick, Motors.

3.1 Main Components

Here we get two parts, Left and Right1. It can be separately shown as below. Components are presented in the following list

(a) LEFT (b) RIGHT

Figure 1: MAIN PART

1. Glass slide holder: To clamp the glass with PDMS&PC to complete the transferring
2. Angle displacement stage: To adjust the table level for glass slide holder
3. Z-direction stage: To control the raising and falling of the glass slide holder

4. Motor holder: To be used for placing the motor
5. XY displacement stage: To control the moving of holder
6. Rotating holder: To take out of glass from whole device by rotating it
7. Heating table: Connecting with thermal couples and heating rod to heat the sample
8. Tractor connecting table: Connecting with tractor and pump
9. Supporting table: Increasing the height
10. Rotating stage: To complete the twisting
11. Motor holder:: To be used for placing the motor
12. Supporting Table: Increasing the height
13. Angle displacement stage: To adjust the table level for glass slide holder
14. Motor holder: To be used for placing the motor
15. Angle displacement stage: To adjust the table level for glass slide holder
16. Motor holder: To be used for placing the motor
17. Y displacement stage: Moving back and forth to search the location of target
18. Motor holder: To be used for placing the motor
19. X displacement stage: Moving left and right to search the location of target
20. Motor Holder: To be used for placing the motor

The above system can be divided into left and right parts, in which the left part mainly accomplishes the transfer-related work, while the right side is mainly responsible for the function of retrieving and labeling the target samples. This part can be fully controlled by the machine, and we will install motors in the reserved table position to realize fully automatic control.
In addition, in order to ensure the height of the machine, we have added a support device under the left and right side of the machine, as shown in the figure. The choice of the device depends on our height requirements. Afterwards, we will fix all the devices to the optical breadboard.

3.2 Microscope

The microscope consists of four parts: a Nikon microscope, a Canon DSLR camera, a light control system for the microscope, an objective lens and an automatically controlled objective disk.(2)

3.2.1 Nikon microscope

The microscope can be purchased by referring to its Nikon website, and since this machine was purchased as a used microscope, some revisions and repairs will be required. Its basic composition can be divided into the microscope support bar (not shown in the figure) and the main part of the microscope, including (22) the connection stage, (23) the light source, (24) the light source controller, (25) the objective lens and objective lens disk. Since we used a Canon DSLR camera to monitor the whole system, we will not discuss the eyepiece section for the time being.
The main function of the microscope is to detect and control our samples in real time, so its quality must be guaranteed, and we choose the microscope made by Nikon Corporation of Japan. Later we will also choose more microscope brands for control and testing.

3.2.2 Canon DSLR camera(2b)

We have chosen a Canon 1300D DSLR camera here for real-time image monitoring and capture. Its advantages compared to CCD, with auto-focus and real-time capture images and beautiful APP and other characteristics. The camera also incorporates Wi-Fi wireless networking capabilities, making it easy to connect to a smartphone for remote shooting, browsing and storing photos.

3.2.3 Light controller(2a)

The controller is a Nikon Model TE2 ps100w. now discontinued. Its interface is simple and clear, and after making the connection. The buttons on the right side are the power and power indicator. The knob on the left side adjusts the intensity of the light.

3.2.4 Objective lens and automatically controlled objective disk(2c)

In order to guarantee imaging, we need to use higher quality objective lenses, and we have also chosen to use objective lenses manufactured by Nikon, including: $10\times, 20\times and 50\times$.
For disk we chose the motorized nosepiece controller model LV-NCNT2, which is still available. The interface consists of two buttons; pressing the left button turns the objective disk clockwise and vice versa.

(a) light controller (b) canon camera (c) disk

Figure 2

3.3 Heating system

Unlike the conventional direct use of a heater with integrated temperature regulation and control, we purchased a separate set of adjustable power supplies and a smart thermocouple with USB3b in order to realize intelligent control and display. Adjustable power3a supply range of voltage 0-24V can be. The temperature range of the thermocouple needs to be greater than 200řC. At the same time, we need to choose two heating rods3c with a rated voltage of 24V and a power of 20W. As shown in the figure3.

The purpose of this heater is to allow the PDMS to slowly diffuse over the surface of the sample and then cool down to allow the PDMS to pick up the sample and complete the process when the PDMS is brought close to the surface of the sample during the transfer operation. Therefore, a visual and precise temperature control system is very important.

(a) Adjustable power

(b) Thermalcouple

(c) heating rod

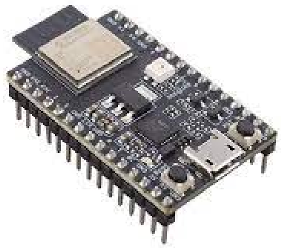
Figure 3

3.4 Motors

The motor system is the highlight of the set, we realize the controlled interaction of human-machine operation through the motor system, which is undoubtedly a more convenient and practical solution. The motor system is divided into three parts: Arduino (ESP32)4b, motor controller4c, and motor4a. The three parts are connected by wires, and the ESP32 is connected to the computer. Through the computer (host computer) to send commands, the electrical signal will be transmitted to the motor controller, through the controller passed to the motor to rotate. The rotation of the motor will drive the stage of the rotary micrometer rotation and thus realize the automatic movement and lifting of the table.

Its advantage perfectly solves the problem of difficult operation in the glove box. We can control everything that can be rotated through the computer terminal. These include 6 motors connected to the stage (Z-direction stage, XY displacement stage×2, Angle displacement stage×2, Rotating Stage), 1 motor connected to the focusing knob of the microscope, 1 motor connected to the heating controller, 1 motor connected to the microscope, and 1 motor connected to the microscope. motor connected to the microscope brightness regulator. For the connection of the motors and knobs, we can either use shaft connectors or our own 3D printed devices.

(a) Motor



(b) ESP32

(c) Motor controller

Figure 4

3.5 Computers and joystick

After we connect the ESP32 to the computer, we use Matlab's program control, so we need to download the genuine matlab software on the computer side. In addition, we need to install the software for Canon cameras (Canon EOS) for real-time detection of the image, and the software for thermocouples (Dagasensor) for temperature control. For the computer, we can choose a Mac computer with high quality and clear image quality. We also need a USB docking station and a long USB cable for the cable connection. If we operate from the glove box, we need to put a controller (laptop or steam deck) in the glove box and use remote control software (team viewer or sunflower remote app) to control it.
The main purpose of the joystick is to move the motor, we use the joytokey app to map the keyboard keys to the joystick for subsequent operation.

3.6 Others

- 12V DC power supply : to provide voltage for the motor
- USB lines
- A motor fan: To cool down the system rapidly
- Shock absorption device
- Optical Breadboard
- Pump and tube
- Smart Switch

SECTION 4

Assembly Guidelines

4.1 Preparation

Besides the necessary components we suported in the last session, we also have a list for additional tools

- Several metric M4 screws (length: 5-20mm)
- Several metric M6 screws (length: 8-15mm)
- hexagonal screwdriver (Metric)
- one-piece screwdriver
- superglue

4.2 Exact process

Next we can assemble it according to the following steps.

4.2.1 Assembling of stage

1. The stage is divided into two parts, i.e. left and right, into the TRANSFER area and the RESEARCH area. Build it layer by layer according to the drawing given in the previous section, from bottom to top, and use the corresponding M4 screws to fix it.
2. After the table has been secured, use the appropriate support structure to secure it to the underside of the left and right STAGES. And secure it to the optical breadboard.

3. Finally, the appropriate height should be tested, firstly to ensure that the objective lens of the microscope does not hit the carrier stage and that enough space is left for subsequent experiments.

4.2.2 Assembling of microscope

1. The microscope was first attached to the objective lens by measuring its body according to the microscope's instructions and fixed to the custom-made 38mm support bar using a stand. Adjust the height and position of the matter and fix it securely to the optical breadboard.

2. After mounting the three objective lenses to the disk and tightening them, the disk is secured in the appropriate position on the microscope unit.

3. Attach the disk to the controller using the supplied cable and plug in the power supply as shown(instruction).

4. Attach the light source to the right side of the microscope's beam, connect the light source to the light source controller using the cable, and turn the light source controller on.

5. There are two levers to the right and below the eyepiece, which can be pulled to adjust the aperture size as well as the dark field light source, respectively. Adjust the coarse and fine focus appropriately.

6. Place the small round support bracket, into the grooved hole above the eyepiece, and then place the camera inside the bracket. After securing it with the screw, turn on the camera switch. Connect the camera to your computer with the USB cable, download the EOS application and double-click to open it.

4.2.3 Assembling of heating device

1. Connect both ends of the two heating rods to the positive and negative terminals of the 24V controllable power supply, tuck the rods into the heating table at the top of the right STAGE, and secure them with a strong, high-temperature-resistant adhesive.

2. Plug the Dagasensor thermocouple also into the small hole in the heating table using strong adhesive to hold it in place, and plug the other end into the computer.

3. Download the corresponding software for the smart thermocouple through the dagasensor website, and when the wiring is connected correctly, subsequent operations can be carried out, at which time a real-time temperature profile will be displayed in the image.

4.2.4 Assembling of motors

1. The process is complex and it is recommended to contact a professional. First, we need to identify four objects, ESP32, motor controller, motor, and main power supply. At the same time we need to have enough wires to make the connections.(5)

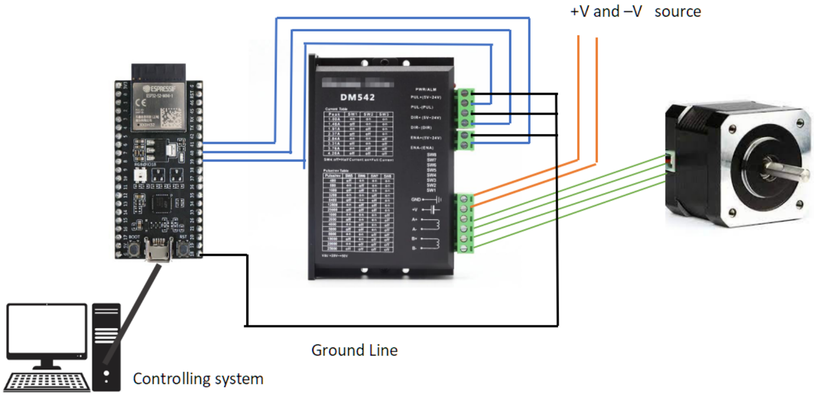


Figure 5: Connections for whole system

2. The general control diagram can be shown in Figure . The motor is connected to the power controller (A+,A-,B+,B-) end, and the corresponding color wire is connected to the motor according to the instructions written on the motor. Connect the VCC and GND terminals on the motor controller to the positive and negative terminals of the 12V power supply respectively.

3. As for the rest of the parts, connect them according to the following instructions. Generally for TB6600 model motors, connect PUL+, DIR+ to the corresponding and appropriate ports of the ESP32 respectively (the next sentence will explain which are the appropriate ports.) PUL- and DIR- are connected to the GND terminal of the Arduino. The ENV+ and ENV- terminals of the motor controller can be left unconnected.

4. Connect all the motors to the corresponding parts via shaft couplings or 3D printed moldings. Tighten the screws or raise the height to ensure that the motor shafts are securely fixed and co-axial with the micrometer (or other shafts).

5. We have basically completed the connection of all the hardware, we need to carry out the appropriate control to ensure the stability of the motor rotation, first of all, we can see that the motor has a 6-speed button on the side, we need to follow the instructions to select the appropriate current and speed. At the same time, we have to ensure that the motor and the corresponding ESP32 serial port is connected, we can carry out the subsequent compilation of the code. The following two tables are a reference for this.

No.	Connect to	PUL+	DIR+
1	Focus	2	4
2	Z direction stage	23	22
3	Y stage	5	18
4	X stage	19	21
5	Y Angle	13	12
6	X Angle	14	27
7	Rotating Stage	26	25
8	Heating device	32	33
9	Light controller	15	33

No.	S1	S2	S3	S4	S5	S6
1	OFF	ON	OFF	ON	OFF	ON
2	OFF	ON	ON	ON	OFF	ON
3	ON	ON	OFF	ON	OFF	ON
4	ON	ON	OFF	ON	OFF	ON
5	ON	OFF	OFF	ON	OFF	ON
6	ON	OFF	OFF	ON	OFF	ON
7	ON	OFF	ON	ON	OFF	ON
8	ON	OFF	ON	ON	OFF	ON
9	ON	OFF	ON	ON	OFF	ON

6. Taking the above ESP32 wiring as an example, we can use the following code for the program.

4.2.5 Assembling of other devices

- Pump and tube: Connect one end of the tube to the vacuum pump and insert the other end into the air nozzle to ensure that the tube does not leak.
- Vibration-proof platform: Place the full countertop on a vibration-proof platform and level it with a spirit level.
- Joystick: The joytokey software is used to map the joystick to the keyboard, and the joystick selects a Bluetooth connection to the computer controlling the Matlab.
- Fan: Connect the fan with a USB socket to the smart power supply, which will control whether the fan turns or not via a remote control.

4.3 Advanced settings: In glove box

The assembly in the glove box is the same, except that the operation will be difficult in order to ensure that the glove box works properly.
First of all, you need to move the whole setup into the glove box, and then you need to build it from the bottom up, placing the damping platform first, and then building the stage on top of it. The main difficulty in the glove box is the transmission of data, and here we have two sets of solutions.

- Stuff a laptop or steam deck into the glove box and use remote software to operate the device in the glove box.
- Wireless transmission of data is performed using the Bluetooth function of the microcontroller and the WIFI function of the Canon camera. Necessary USB cable connections (thermocouple and power cable) are made using the aviation plug.

SECTION

5 Guidance for use

5.1 Process for scanning samples

1. Turn on all main power supplies except the motor, including the microscope objective turntable, microscope light controller, and computer.

2. Check that the stage is free to rotate and is in a more centered position, after keeping the table stable. Gently place the wafer containing the sample on the stage. Turn on the vacuum pump so that the mounting piece is securely fastened to the table.

3. Start the matlab software on your computer, Canon EOS software.

4. Select the appropriate objective lens (usually a low magnification lens) and adjust the brightness and aperture so that the picture appears in the screen. Manually adjust the coarse focus to make the picture as clear as possible.

5. At this point, power on the motor, run the Matlab program, and use the keyboard or joystick to move the STAGE according to the instructions and focus appropriately to find the target sample.

5.2 Process for transferring samples

1. Turn on all main power supplies except the motor, including the microscope objective turntable, microscope LIGHT CONTROLLER, computer, etc., and additionally turn on the power to the heating unit.

2. Check that the stage is free to rotate and is in a more centered position, after keeping the table stable. Gently place the wafer containing the sample on the stage.Turn on the vacuum pump so that the mounting piece is securely fastened to the table.
3. Start the matlab software on your computer, Canon EOS software, Dagasensor software
4. Select the appropriate objective lens (usually a low magnification lens) and adjust the brightness and aperture so that the picture appears in the screen. Manually adjust the coarse focus to make the picture as clear as possible.
5. At this point, power on the motor, run the Matlab program, and use the keyboard or joystick to move the STAGE according to the instructions and focus appropriately to find the target sample.
6. Attach the mount with PDMS and PC to the exclusion of the left shelf and move it between the sample and the objective. Determine its exact position by focusing.
7. Adjust the button on the keypad or handle that corresponds to the Z-direction control to slowly bring the glass slide closer to the sample. When the PDMS is on the mounting plate but is not close to the sample, the lowering stops.
8. Slowly adjust the heater in 5V increments while observing the temperature displayed by the thermocouple in real time. As the temperature rises, the PDMS moves closer to the sample and stops heating when it has covered the sample. A fan is used to cool down the temperature. And then raise the stage in Z direction.
9. Depending on the nature of the material, different heating times and temperatures are used for different kinds of samples. This is only an example using PDMS and PC.

5.3 Special Process

- In glovebox: A vacuum check is required. As well as turning on the VM for line testing.
- How to run the program: Click on the run button of matlab, then this window will pop up, please make sure, the last mouse click position in the pop-up window. Please follow the instructions in the popup window to select the specific key and gear.

6

SECTION

Common Problems and Solutions

6.1 Related to Stage & Motors

1. **Why is my motor getting hot badly?**

   Motor heating is a natural phenomenon, if the temperature does not exceed 50 degrees please do not worry. However, if the temperature exceeds 80 degrees, please stop the operation of the motor immediately to prevent fire.

2. **Why is my motor not working?**

   It's a classic question. First check that the power is on, and second check that the wiring is not connected incorrectly. If the wiring is connected correctly and the power is turned on, the power light of the motor controller will be on and the motor is in a locked state. Check that the code side is not reporting an error, if there is an error please refer to the section that follows. Next, check whether the knob of the table to which the motor is connected is in the state of being screwed. If you still can't solve it, please contact us.

3. **Why does the motor turn normally but the knob turns discontinuously?**

   Check for looseness between the shaft coupling and the knob, be sure to secure it.

4. **Why does the motor jam and shake when turned at certain angles?**

   This is due to the fact that the motor is not coaxial with the knob and is generally not a cause for concern. If it is particularly noticeable, please make appropriate modifications.

6.2 Related to Microscope

1. **Why is it that when I turn on the light source, the screen is still dark?**

   Make sure you have adjusted the aperture, then pull the lever that controls the aperture to the proper position.

2. **Why does my EOS software show that the camera cannot be detected?**

   Make sure that the camera's cable is connected to the computer, or that the computer is connected via WIFI. If you still can't solve it, please refer to the Canon website.

3. **Why is it difficult to find the frame when using the motor to focus?**

   The motor is connected for fine focusing, at which point the motor needs to be turned off for coarse focusing.

4. **Why is the picture very jittery when I adjust to high magnification?**

   Shock is inevitable. However, if the vibration is excessive, please test the attachment for the influence of a vibration source.

6.3 Related to Heating device

1. **Why does Dagasensor's software show that the device cannot be detected when it is opened?**

   Make sure the thermocouple wiring is not broken. If the intact thermocouple is connected into the computer under normal conditions, the temperature profile will be displayed automatically.

2. **Why is the thermocouple's temperature profile trending down as I heat it up?**

   Ensure that the red and blue wires of the thermocouple are connected correctly; reversing the connection will cause the temperature rise and fall trends to not match.

3. **Why do I get left and right jumps in the display when I use the motor to control the heater?**

   Caused by the differential speed of motor rotation, it is an inevitable phenomenon, and the motor parameters can be adjusted appropriately.

4. **Why does it heat up slowly?**

   Check that the two heating rods are connected to the positive and negative terminals of the heating unit, and check for a broken circuit.

5.

6.4 Related to software

1. **Why does the code give an error saying "port not detected"?**

Because the ESP32 is not connected to the computer, or the environment is not configured when it is connected to the computer. The solution is to connect the microcontroller to the computer, open the Arduino App of Matlab, configure it, and select "ESP32-WROOM-DevKitC" and the corresponding COM serial port to configure. After successful configuration, run the program.

2. **Why does Matlab code run with the message "Unavailable PIN used"?**

   This is due to Matlab's requirements for the ESP32 GPIO pins. Matlab only recognizes the wires on the featured pins, so please check the matching GPIO pins in the command bar belonging to Arduino.

3. **What to do if the code reports other errors?**

   Debug on your own. the source code must be right.

4. **Why doesn't it work with a joystick?**

   Make sure Bluetooth is connected to your computer and also mapped with joytokey. If nothing happens, please reboot.

5. **Why does the motor not rotate when I move the mouse and then click on the relevant keys on the keyboard?**

   It is necessary to keep the position of the last mouse click in the Matlab pop-up window at all times.

SECTION

7 Appendix

7.1 Matlab Code

```
function motor_control_test3()
% close all;clear;clc;
clear ardu
ardu = arduino('COM3', 'ESP32-WROOM-DevKitC');
global delayTime

fig = figure('Name', 'Keyboard-Driven␣UI','Position', [0, 200, 750, 400], 'KeyPressFcn',@
textLines = {
    '(1)Press␣UP,␣DOWN,␣LEFT,␣RIGHT␣to␣move␣the',
    '␣␣␣␣␣RIGHT␣Sample␣Platform␣␣aong␣different␣directions',
    '(2)Press␣A,␣D,␣W,␣S␣to␣adjust␣the␣the␣tilting',
    '␣␣␣␣␣angle␣of␣the␣RIGHT␣Sample␣Platform',
    '(3)Press␣N,␣M␣to␣control␣the␣twist␣angle',
    '(4)Press␣J,␣K␣to␣control␣the␣UP␣and␣DOWN␣motion␣of␣the␣pickup␣plate',
    '(5)Press␣O,␣P␣to␣control␣the␣focus␣of␣the␣microscope',
    '(6)Press␣Z,X,␣C,V␣to␣change␣the␣moving␣speed',
    '␣␣␣␣␣of␣the␣RIGHT␣sample␣platform␣(From␣Z␣to␣V,␣Speed␣low␣to␣high)'
};

annotation("textbox", [0.1, 0.1, 0.8, 0.8], 'String', textLines,...
    "FitBoxToText", "on", "VerticalAlignment", "top", "HorizontalAlignment", "left");
keyText = uicontrol('Parent', fig, 'Style', 'text', 'Position', [50, 80, 200, 30]);
```

```
shijianText = uicontrol('Parent', fig, 'Style', 'text', 'Position', [50, 50, 200, 30]);


ispress = struct('uparrow',false,'downarrow',false,'rightarrow',false,'leftarrow',false,'
    'o',false,'p',false,'n',false,'m',false,'j',false,'k',false,'z',false,'x',false,'c',f
%ispress = struct('uparrow',false,'downarrow',false,'rightarrow',false,'leftarrow',false)
game = timer('ExecutionMode','FixedRate','Period',0.01,'TimerFcn',@(obj,event) gamefcn(ob
start(game)
%%

function gamefcn(obj,event,ardu)
% PUL_PIN= 'D13';
% delayTime=0.01;
% if ispress.g
%     delayTime = di_3;
% end

    function change_delaytime_1
        delayTime=4e-1;
    end

    function change_delaytime_2
        delayTime=4e-2;
    end

    function change_delaytime_3
        delayTime=4e-3;
    end

    function change_delaytime_4
        delayTime=4e-4;
    end

if ispress.z
    change_delaytime_1
end
if ispress.x
    change_delaytime_2
end
if ispress.c
    change_delaytime_3
end
if ispress.v
    change_delaytime_4
end
```

```
if ispress.uparrow
    PUL_PIN_1='D13';
    DIR_PIN_1='D12';
    motor_control_cw(delayTime,PUL_PIN_1,DIR_PIN_1,ardu)
end
if ispress.downarrow
    PUL_PIN_1='D13';
    DIR_PIN_1='D12';
    motor_control_ccw(delayTime,PUL_PIN_1,DIR_PIN_1,ardu)
end
if ispress.leftarrow
    PUL_PIN_1='D27';
    DIR_PIN_1='D14';
    motor_control_cw(delayTime,PUL_PIN_1,DIR_PIN_1,ardu)
end
if ispress.rightarrow
    PUL_PIN_1='D27';
    DIR_PIN_1='D14';
    motor_control_ccw(delayTime,PUL_PIN_1,DIR_PIN_1,ardu)
end

if ispress.a
    PUL_PIN_1='D25';
    DIR_PIN_1='D26';
    delayTime = 0.01;
    motor_control_ccw(delayTime,PUL_PIN_1,DIR_PIN_1,ardu)
end

if ispress.d
    PUL_PIN_1='D25';
    DIR_PIN_1='D26';
    delayTime = 0.01;
    motor_control_cw(delayTime,PUL_PIN_1,DIR_PIN_1,ardu)
end

if ispress.w
    PUL_PIN_1='D19';
    DIR_PIN_1='D21';
    delayTime = 0.01;
    motor_control_ccw(delayTime,PUL_PIN_1,DIR_PIN_1,ardu)
end

if ispress.s
    PUL_PIN_1='D19';
```

```
    DIR_PIN_1='D21';
    delayTime = 0.01;
    motor_control_cw(delayTime,PUL_PIN_1,DIR_PIN_1,ardu)
end

if ispress.j
    PUL_PIN_1='D';
    DIR_PIN_1='D';
    delayTime = 0.01;
    motor_control_ccw(delayTime,PUL_PIN_1,DIR_PIN_1,ardu)
end

if ispress.k
    PUL_PIN_1='D';
    DIR_PIN_1='D';
    delayTime = 0.01;
    motor_control_cw(delayTime,PUL_PIN_1,DIR_PIN_1,ardu)
end

if ispress.m
    PUL_PIN_1='D32';
    DIR_PIN_1='D33';
    delayTime = 0.01;
    motor_control_ccw(delayTime,PUL_PIN_1,DIR_PIN_1,ardu)
end

if ispress.n
    PUL_PIN_1='D32';
    DIR_PIN_1='D33';
    delayTime = 0.01;
    motor_control_cw(delayTime,PUL_PIN_1,DIR_PIN_1,ardu)
end

if ispress.o
    PUL_PIN_1='D5';
    DIR_PIN_1='D18';
    delayTime = 0.01;
    motor_control_ccw(delayTime,PUL_PIN_1,DIR_PIN_1,ardu)
end

if ispress.p
    PUL_PIN_1='D5';
    DIR_PIN_1='D18';
    delayTime = 0.01;
    motor_control_cw(delayTime,PUL_PIN_1,DIR_PIN_1,ardu)
```

```
end
end

function krfcn(obj,event)
ispress.(event.Key) = false;
end

function kpfcn(obj,event)
ispress.(event.Key) = true;
end
% end

function myKeyPressFunction(src, event, fig)
    keyPressed = event.Key;

    if strcmp(keyPressed, 'g')
        delayTime = di_3;
        yourFunction();
    end
end


function myKeyReleaseFunction(src, event, fig)

end


function yourFunction()

    disp('a');
end
end

function choice = choosedialog

    d = dialog('Position',[300 300 250 150],'Name','Select One');

    txt = uicontrol('Parent',d,...
           'Style','text',...
           'Position',[20 80 210 40],...
           'String','Select Speed');

    popup = uicontrol('Parent',d,...
           'Style','popup',...
           'Position',[75 70 100 25],...
```

```
        'String',{'0.1';'0.01';'0.001';'0.0001'},...
        'Callback',@popup_callback);

    btn = uicontrol('Parent',d,...
        'Position',[89 20 70 25],...
        'String','Close',...
        'Callback','delete(gcf)');


    choice = 0.1;

    set(d, 'KeyPressFcn', @processKeyPress);

    function popup_callback(popup,event)

        idx = popup.Value;

        popup_items = popup.String;

        choice = str2double(char(popup_items(idx,:)));
    end

    function processKeyPress(src, evt)
        keyPressed = evt.Key;

        if strcmp(keyPressed, '1')
            choice = 0.1;
            popup.Value = 1;
        elseif strcmp(keyPressed, '2')
            choice = 0.01;
            popup.Value = 2;
        elseif strcmp(keyPressed, '3')
            choice = 0.001;
            popup.Value = 3;
        elseif strcmp(keyPressed, '4')
            choice = 0.0001;
            popup.Value = 4;
        elseif strcmp(keyPressed, 'return')
            delete(d);
        end
    end

    uiwait(d);
end
```

```matlab
function motor_control_cw(delayTime,PUL_PIN_1,DIR_PIN_1,ardu)
    % PUL_PIN_1 = 13 ;
    % DIR_PIN_1 = 12 ;
    %note: PUL_PIN_1 controls motor #,
    writeDigitalPin(ardu, DIR_PIN_1, 0);
    writeDigitalPin(ardu, PUL_PIN_1, 1);
    pause(delayTime);
    writeDigitalPin(ardu, PUL_PIN_1, 0);
    pause(delayTime);
end

function motor_control_ccw(delayTime,PUL_PIN_1,DIR_PIN_1,ardu)
    % PUL_PIN_1 = 13 ;
    % DIR_PIN_1 = 12 ;
    %note: PUL_PIN_1 controls motor #,
    writeDigitalPin(ardu, DIR_PIN_1, 1);
    writeDigitalPin(ardu, PUL_PIN_1, 1);
    pause(delayTime);
    writeDigitalPin(ardu, PUL_PIN_1, 0);
    pause(delayTime);
end
```

7.2 Shopping List

...