**Android robot instructions**

**Installation:**

Install from Google Play Store:

Pydroid 3  
Pydroid repository plugin  
Pydroid Permissions Plugin  
Stockfish Engines OEX (if using Stockfish)

Installing Pydroid 3 will automatically install Python3  
  
Open Pydroid 3 and choose “Pip” from the menu. In there you can install Python modules by merely typing their names:

kivy  
pyserial  
psutil  
numpy  
opencv-python  
plyer

Arduino files are at  
<https://github.com/rpd123/Arduino-Firmware>

After downloading to the PC, use the Arduino IDE to upload them to the Arduino. But before that …

Change two lines in an Arduino library file, which in my case is at:

ProgramFiles(x86)\Arduino\hardware\arduino\avr\cores\arduino\HardwareSerial.h

A couple of lines with buffer size 64 – change to

#define SERIAL\_TX\_BUFFER\_SIZE 128

#define SERIAL\_RX\_BUFFER\_SIZE 128

Pair the phone and the HC-05 if not already paired

Download files for the phone from:

https://github.com/rpd123/chess-robot/tree/main/versiongui

**Configuration**

Change configuration variables in CBstate.py, robotmove.py and Arduino config.h as appropriate.

Line 5 of config.h should be changed to  
#define SERIALX Serial2

when Bluetooth is used.

**Hardware notes**

The RX output from the HC-05 is connected via a voltage divider to D16 on AUX-4 on the RAMPS.

The TX output is connected directly to D17 on AUX-4.

https://osoyoo.com/2016/07/03/reprap-3d-printer-circuit-connection-graph/

Voltage divider example (Ignore connections to Arduino):

https://electronics.stackexchange.com/questions/280500/why-do-you-have-to-use-a-voltage-divider-with-hc-05-bluetooth-module-arduino

**Running**

Hide the on-screen navigation buttons (if any). (See below).

Have the OTG and wireless mouse dongle in

Open Pydroid3, open **main.py** and press the yellow button. (Note: NOT CBint.py)

Then you should see “Let’s play chess” at the top of the screen.

This area will be used for messages. The phone is likely to be slow, so at each stage, wait for a new message.

Ensure that you can see the whole of the chessboard in the video view. Click on the “Calibrate chessboard” button. A static image will appear below, and a new message.

Using the mouse, click on the corners of the playing area **starting at the bottom left**. (Note: NOT top left)

You should then see the redlines image.

Ensure that you have power to the RAMPS.

Click on Start game, and wait … until it says Game Started.

Click on Start robot and wait.

That button text will change to say “Switch on steppers” Do that.

Button text will change to “Adjust ROBOT placement”. Adjust position of robot base if needed, then press button.

Make a move then press “I’ve moved”

**Further info**

Once the code is up and running, steppers are on and robot is in home position:

At any time you can put the chess pieces back to the opening position and  
 Optionally re-calibrate chessboard  
 Optionally re-calibrate robot  
 Click on New Game  
 Then make move

In particular, you can start a new game without re-calibrating the chessboard and without re-calibrating the robot.

You can re-calibrate the chessboard or the robot if the pieces are in the opening position.

If Pydroid is not running  
 Ensure chess pieces are in the opening position  
 Open Pydroid, open and run main.py  
 Optionally calibrate chessboard  
 Click on New Game  
 start and calibrate the robot  
 make move

When the program is closed, the connection to the robot is irretrievably lost, so the robot always has to be re-calibrated when starting from scratch. (But the chessboard does not have to be re-calibrated).  
  
The calibration of the chessboard is retained, so as long as the board, robot and phone have not moved, it is not necessary to re-calibrate the board

**Hiding on-screen navigation buttons**

This can usually be done via Settings:  
<https://support.google.com/android/answer/9079644?hl=en>