**מסמך עזר – פרויקט הזרוע**

זהו מסמך עזר לכל מי שעתיד לקרוא את הקוד של הפעלת הזרוע הרובוטית המחוברת לכיסא הגלגלים באלי"ן.

נתאר את הקבצים העיקריים ואת רצף הפעולות החל מאת הרצת התוכנית.

נתאר את ההבדלים בין הזרוע באלי"ן לזרוע בחדר של אלישי, ואיך ההבדלים האלה באים לידי ביטוי בקוד.

נתאר את הכיוונים השונים בהם הזרוע יכולה לזוז בפועל, ואילו כפתורים מתאימים לכך בג'ויסטיק.

**מערכת הצירים:**  
(0,0,0) – הבסיס של הזרוע  
ציר x – לכיוון הכיסא, כלומר הולך ימינה  
ציר y – הולך קדימה  
ציר z – הולך למעלה

**מנועי היד:**  
היד הרובוטית מורכבת מ9 מנועים:  
1. הבסיס   
2-3. באלין משלימות ל 361 ובאו"פ משלימות ל 360   
4-5. באלין משלימות ל 361 ובאו"פ משלימות ל 362  
6. סיבוב סביב ציר ה y  
7. סיבוב סביב ציר ה x  
8. סיבוב של הגריפר סביב ציר ה y  
9. גריפר (תופס/משחרר)

**הסבר על מצבי הזרוע:**על מנת לשפר חווית משתמש, לזרוע יש כמה פעולות אוטומטיות שמהוות התחלה טובה עבור המשתמש, ומשם הוא יכול לכוון את הזרוע אל המטרה באופן יותר מדוייק באמצעות הג'ויסטיק.  
לזרוע 4 מצבים שונים: Home, Drinking, Shelf, Floor  
לכל מצב (חוץ מ Home) יש 3 מעברים: Default, Target, Chair  
Default – מצב התחלה  
Target – תזוזה ראשונית אל עבר המטרה  
Chair – הבאת המטרה אל עבר המשתמש

RoboticArm.py

משהו שכדאי לדעת לפני:  
ישנם בפרויקט כמה מרחבי פעולה בהם אפשר "לדבר" עם הזרוע:  
1. לעשות IK ולקבל – 5 זוויות (בין -2pi ל 2pi )  
2. הזוויות ששולחים ל9 המנועים של הזרוע – 9 זוויות

* פונקציות שממפות בין המודל לרובוט (בין מצב 1 למצב 2 שהוזכרו למעלה), וההיפך.
* הגדרות של החומרה. דברים חשובים:
  + Profile Velocity ו- Profile Acceleration – מגדירים יחד את המהירות בה הזרוע זזה.
  + Priority – עדיפות מי מהמנועים יזוז קודם בעת שינוי של הזוויות.
* הגדרת המצבים השונים של היד.

IK.py

ה class שלנו הוא viper300  
מכיל את הפונקציות שמחשבות את ה FK וה IK.

PS4\_controller.py

מגדיר את השליטה, הן באמצעות ה joystick והן באמצעות החלק הנוירומורפי.

arm\_control\_ALYN.py

הקובץ שמריצים.

**התקנות:**

הסביבה עובדת על ubuntu 18.04  
פייתון 3.6.9  
תיקיית downloads for loihi:

<https://drive.google.com/file/d/1w_lup2umiwYxS9rsopCmGcL6e7DlIPi0/view?usp=sharing>

תוכנת dynamixel

Visual studio code

צור סביבת פייתון:

5.1 cd ~

5.2 python3 -m venv alyn\_env

5.3 source alyn\_env/bin/activate

ותתקין שם את requirements.txt

cat requirements.txt | xargs -n 1 pip install

## LOIHI Recommended Setup

The following recommendation are based on our test configuration. Additional steps might be necessary on other alternatives.

1. OS: Ubuntu 18.04
2. Python 3.6.9
3. pip (latest)

* we’ll use nxsdk version 0.9.8

## Installation Guide

1. Plugin Kapoho Bay to your Linux Machine
2. Verify USB devices is properly recognized

2.1 Run lsusb or lsusb -t to see if the USB was plugged in and recognized by your machine

user@machine:~$ lsusb

...

Bus 001 Device 107: ID 0403:6010 Future Technology Devices International, Ltd FT2232C Dual USB-UART/FIFO IC

Bus 001 Device 106: ID 0403:6010 Future Technology Devices International, Ltd FT2232C Dual USB-UART/FIFO IC

Bus 001 Device 105: ID 0403:6010 Future Technology Devices International, Ltd FT2232C Dual USB-UART/FIFO IC

user@machine:~$ lsusb -t

...

/: Bus 01.Port 1: Dev 1, class="root\_hub", Driver=ehci-pci/2p, 480M

|\_\_ Port 1: Dev 2, If 0, class="Hub", Driver=hub/6p, 480M

|\_\_ Port 2: Dev 39, If 0, class="Human" Interface Device, Driver=usbhid, 1.5M

|\_\_ Port 6: Dev 104, If 0, class="Hub", Driver=hub/4p, 480M

|\_\_ Port 1: Dev 105, If 0, class="Vendor" Specific Class, Driver=, 480M

|\_\_ Port 1: Dev 105, If 1, class="Vendor" Specific Class, Driver=, 480M

|\_\_ Port 2: Dev 106, If 0, class="Vendor" Specific Class, Driver=, 480M

|\_\_ Port 2: Dev 106, If 1, class="Vendor" Specific Class, Driver=, 480M

|\_\_ Port 3: Dev 107, If 0, class="Vendor" Specific Class, Driver=, 480M

|\_\_ Port 3: Dev 107, If 1, class="Vendor" Specific Class, Driver=, 480M

2.2 Remove "ftdi\_sio" kernel modules

2.2.1 Run  
 lsusb -t   
and check if your see the FTDI interfaces.   
If the driver says ftdi\_sio, you may remove these kernel modules using  
 sudo rmmod ftdi\_sio 2.2.2 Run lsusb -t   
to ensure that **ftdi\_sio** is no longer there

1. If you not yet created a virtual environment, do it. Then activate it.
2. 5.1 cd ~

5.2 python3 -m venv alyn\_env

5.3 source alyn\_env/bin/activate

5.4 pip install -U pip

5.5 Do all following steps within the virtual environment

1. There should be 2 packages within the sub-directory (downloads\_alyn)

6.1 nxsdk-0.9.8.tar.gz - A pip installable tarball to install NxSDK

6.2 nxsdk-apps-0.9.8.>.tar.gz - Tarball for Jupyter/Python based tutorials and complete NxSDK documentation

1. Install NxSDK: **python -m pip install nxsdk-0.9.8.tar.gz**

7.1 Ignore the "Failed building wheel for nxsdk" and the associated "Failed to build nxsdk". This is a known error and pip will retry with setup.py.

1. Verify NxSDK installation: **python -c "import nxsdk; print(nxsdk.\_\_version\_\_)"**
2. Find NxSDK installation directory (nxsdk\_install\_dir): **python -c "import nxsdk; print(nxsdk.\_\_path\_\_)"**
3. Unzip Tutorials, Docs and Modules in your home directory: **mkdir nxsdk-apps && tar xzf nxsdk-apps-0.9.8.tar.gz -C nxsdk-apps --strip-components 1**
4. Copy 99-kapoho\_bay.rules from nxsdk-apps/utils to /etc/udev/rules.d and run **sudo udevadm control --reload**
5. Install g++ multilib, libgflags-dev, libgoogle-glog-dev and libprotobuf

12.1 sudo apt-get install g++-multilib  
  
12.2 sudo apt-get install libgflags-dev  
  
12.3 sudo apt install libgoogle-glog-dev  
  
12.4 You may compile libprotobuf17 from source or install libprotobuf17 from debian packages. Latter is shown below.

* 1. wget <http://ftp.debian.org/debian/pool/main/p/protobuf/libprotobuf17_3.6.1-4~bpo9+1_amd64.deb>
  2. sudo dpkg -i libprotobuf-lite17\_3.6.1-4~bpo9+1\_amd64.deb

13.**Test FPIO** /bin/x86/lakemont\_driver --test-fpio. If you see the following, Kapoho Bay is detected. Sample output:

$ `python3 -c "import nxsdk; print(nxsdk.\_\_path\_\_[0])"`/bin/x86/lakemont\_driver --test-fpio

Using Kapoho Bay serial number 998

test\_fpio\_loopback: chips=1 num=10000 len=16 time=238266us => 0.671518Mb/s

14. export KAPOHOBAY=1

15. Run tutorials per instructions in the following sections

16. When complete, deactivate virtual environment

16.1 deactivate

**טיפול בשגיאה נפוצה ב loihi:**  
<https://github.com/sirfz/tesserocr/issues/130>

**התחלה מ0 אחרי שהכל מותקן ושום דבר לא מחובר למחשב:**  
הפעל את הסביבה בה התקנת את ה loihi:

source alyn\_env/bin/activate

קודם מחברים את הusb של הזרוע למחשב.  
לאחר מכן נותנים הרשאות לusb, לדוג':

sudo chmod -R 777 /dev/ttyUSB0

לאחר מכן מחברים את ה loihi למחשב.

כל פעם שמחברים את ה loihi, יש לבצע את שלושת הפקודות הבאות אחת אחרי השנייה:

lsusb -t

sudo rmmod ftdi\_sio

`python3 -c "import nxsdk; print(nxsdk.\_\_path\_\_[0])"`/bin/x86/kb/lakemont\_driver –test-fpio

והפלט שמתקבל מהפקודה האחרונה צריך להיות התיאור של ה chip, כל פלט אחר הוא לא תקין.  
  
תקלות:

בדוק בתוכנת dynamixel שאכן הזרוע מחוברת למחשב.  
במידה והמחשב לא מזהה את ה usb, החלף כניסה ונסה שוב. לפעמים צריך לעשות את התהליך הזה כמה פעמים.  
  
לאחר ש dynamixel מזהה את הזרוע, אל תעשה disconnect, עד שעשית את הבדיקה שה loihi עובד.   
לפני הרצת התוכנית יש לעשות disconnect ב dynamixel אחר התוכנית לא תוכל לגשת לזרוע.