UE 2-4 Robotic Engineering

PhantomX Pincher Robot Arm - Technical Survey report

Submitted by **Meldrick REIMMER and Selma BOUDISSA**

Contents

Chapter 1

Introduction

1.1 History

The history of robots took root only in recent modern consciousness given the complex mechanisms involved in their making, but in fact, such devices have roots as far back as ancient Greece and in early Chinese dynasties. Descriptions of ancient robots date back to the 1st century. These devices are known as automatons, from the Greek word automatons, or "acting of one's own will." Essentially, automatons are non-electronic moving machines that mimic human or animal actions. It has become increasingly difficult to define exactly the word "Robot" given the rapid developments that is occurring in the science and business of robotics. Even Joseph Engelberger, often credited as the "Father of Robotics", was said to have once remarked, "I can't define a robot, but I know one when I see one." [1] Once the word Robot is mentioned it brings up many images in the mind of people. Cartoon lovers will go straight imaging "transformers", "Star Wars", also most readers will remember the "Rover Sojourner", which explored the Martian landscape as part of the Mars Pathfinder mission for NASA. Most recent usage of robots can be seen in our daily lives especially with industrial operations where most robots have one thing in common which is an arm. Most common jobs they do are sorting, assembling and drilling. Robots in the present have been a great asset because they can perform jobs that are too dangerous for humans.

1.2 Low cost review

There are many robots arm in which perform different tasks and some which performs the same task. In recent market survey, there has been a high rise of demand for robotic arm. Several robot manufacturers have come out with several robot arms. There have been many factors in which they produce it some being the cost, its strength, and its reliability. Some few examples are the Dobot robot arm. It has a four-axis robotic arm designed for Makers, artists, educators, and scientists. The high-quality aluminum frame provides stability and the Arduino-controlled stepper motors provide precision and accuracy. The arm can be controlled with Bluetooth via a smartphone app or PC. There is a control rig to allow Dobot to follow the motion of your hand, and it can recognize objects via a web camera. In addition to a gripper, the Dobot has multiple heads for different types of operation, including laser engraving. Another example is the 7Bot.It has six axis aluminum frame and custom high torque steel gear 24W servos make for an impressively robust and stable platform. The servos provide positional feedback with 0.18-degree accuracy. It is available with options for a vacuum cup gripper or a two-finger gripper. 7Bot is designed to be easy to use. You can train it by physically guiding it; their Kickstarter page brags that the grandfather of one of the creators taught 7Bot Chinese calligraphy. You can use computer vision to have 7Bot follow and copy your movements. It can be programmed to sort objects by color. There is also a 3D model UI for programming and an API for more advanced control.

1.3 Objectives

In this project, we will study the PhantomX pincher arm robot. We will give analysis on each part of the robot and give a general overview with regards on how it works together with its compatible components for example Arduino IDE and ROS interfacing, Since PhantomX pincher arm robot uses the Arduino board. We first know the Arduino IDE its best use for to check the stability of the motor in regards to how its moves and its other attributes.

Chapter 2

PhantomX Pincher Robot Arm

The PhantomX Pincher is a robotic arm designed by Trossen robotics. The phantomX Pincher programmable robotic arm is a robotic kit that allows you to perform basic tasks like grabbing, moving and flipping of small objects in your environment. It is a 5 Degrees of freedom robotic arm consisting of 5 joints which allows rotating, lifting or moving of objects. It is equipped with 5 Dynamixel AX-12A servomotors which enables each motor respond to specific command assign to it. This hardware kit comes with everything needed to physically assemble and mount the arm as a standalone unit or as an addition to your TurtleBot robot platform. The PhantomX Pincher is also designed to complement the TurtleBot and TurtleBot 2 robotic kits, enabling you to add a new possibility for interaction with your environment to your programmable robot. It is less expensive compared to other types of robotics arm.

Key Features:

- X-12A Dynamixel Actuators
- Solid Needle Roller Bearing Base
- Rugged ABS construction
- Arbotix Robocontroller for Onboard Processing
- Custom Parallel Gripper
- Mounting Brackets for Cameras and Sensors

Kit Includes:

- 5x AX-12A Dynamixel Actuators
- PhantomX Pincher Arm Hardware & Frame Kit
- ArbotiX Robocontroller
- 12v 5amp power supply
- FTDI 5v Programming Cable
- 2ml Bottle of Turbo-Lock

Chapter 3

Software specifications

In this part, we will be talking about the software that is best compatible to the PhantomX Pincher arm. This software collaborates with each other in making the PhantomX Pincher arm perform its specific task. There have been many tests with other substitute software?s but for the best satisfactory results the above software is recommended.

3.1 Ubuntu

This is an open source operating system first released on October 20, 2004. It is built upon the code base of Debian Linux. It is named after Ubuntu, a South African philosophy which embodies the belief in the bond of sharing which connects all of humanity. This operating system was chosen since we will be using the Robot Operating Systems(ROS) which is best operated in the Ubuntu environment. The ROS (Robot Operating System) provides libraries and tools to help software developers create robot applications. It provides hardware abstraction, device drivers, libraries, visualizers, message-passing, package management, and more.

3.2 Arduino

This is also an open source software which makes it easy to write code and upload it to the board. It also can be run on the Ubuntu Operating System. This will mostly be use to try the first couple of tests of the assembled PhantomX Pincher. It is intended for anyone making interactive projects which is similar to our case.

3.3 Dynamanager

This software is used to check non-responsive servos. If a servo's ID has become changed from what the user thinks, Dynamanager can identify the current ID and change it. The DynaManager software requires Java to run.

Chapter 4

Mounting of the arm

4.1 Blink Test

This step is really important and come before the setup of the IDs. Now that all our softwares are download and install correctly, we pickt the ArbotiX board from the boards menu. In the first part we select the proper board

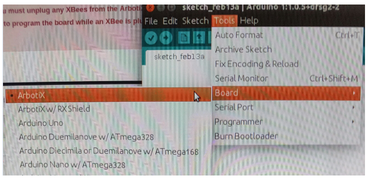

Figure 4.1: ArbotiX

Pick the serial port for the FTDI device.

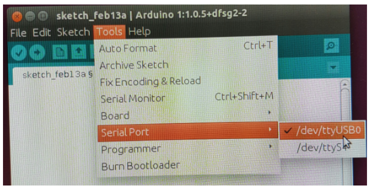

Figure 4.2: Serial port

Now that the board is setted and the serial port too, open the "ArbotixBlink" sketch

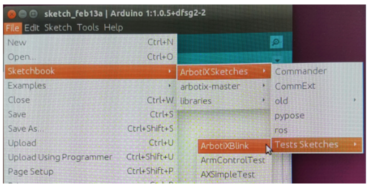

Figure 4.3: select the blink test

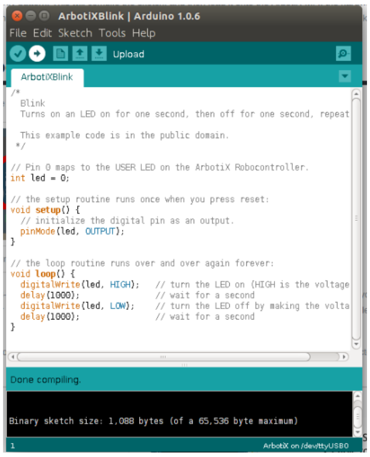

Figure 4.4: ArbotiX Blink

Click on the "Verify Button" this will attempt to compile the sketch. If all the software is installed properly you will see a 'Done Compiling'.Click on the "Upload" button, this will compile the sketch, and then load it onto the ArbotiX-M.

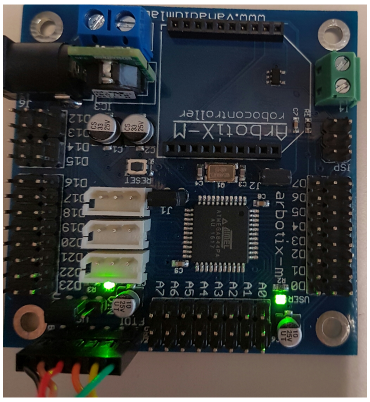

Figure 4.5: Blink test

As you can see in the picture above, the hardware is connected properly this is signalize by the green user light flicker while the Arduino IDE displays 'Uploading' message.

4.2 Setup the ID servos

To setup the servos ID it's simple you have first to download and install DynaManager when it's done you have to connect with the cable the servos to the right emplacement in the arduino board, plug the USB port and the battery.

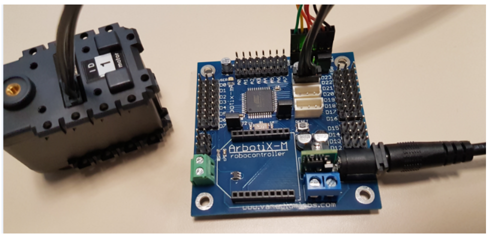

Figure 4.6: Setup of the ID for the servos

With the dynamanager you just have to attribute the number of the ID from 1 to 5.

Figure 4.7: Setup ID n°1

4.3 Mounting

When all the servos ID are attributed we can start the mounting of the arm following the step on the phantomx assembly guide. First we receive a box (as show figure 4.8) with all the components and tools to assembly the robot.

For the mounting we follow the assembly guide, that you can found in the references link n°4

After we start to assemble the arm step by step as the instructions in the guide. And then we obtain the robot arm that you can see in the first page of this report.

Figure 4.8: Assembling of the arm

Figure 4.9: PhantomX Pincher Robot Arm

4.4 Testing

When the arm is well assembled, you have to do some basic test to check that everything was ok with the mounting of the robot arm. The test use is the Pincher Test. To run the build check test you will need to upload the build check firmware onto your ArbotiX Robotcontroller. For more details go check check on the website of the link reference n°2.

Open Arduino IDE and open:

- File->Sketchbook->Tests Sketches->PincherTest

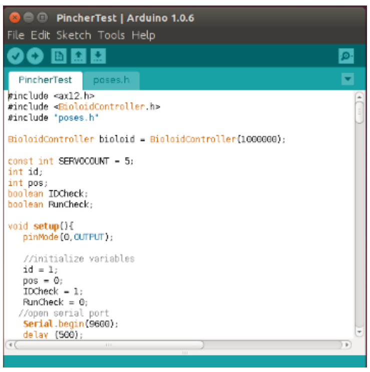

Figure 4.10: Pincher Test

Once the program, show above, is loaded make sure that your robot is powered from a 12V power supply, and that the power jumper is set to "VIN".

Chapter 5

ROS Interfacing

This part is the step to learn how to setup and work with PhantomX Pincher Robot Arm with ROS using the arbotix ROS package.

5.1 Creating a ROS package

For this ROS interfacing part, we have to check if we setup correctly the ID to each servos motor, which we have already given details of how to do it in the chapter 4. First we have to create a new package. The main idea of creating a package is because to be integrated to ROS, a ROS node have to be included in a package. Here we used the catkin build system. You should create the new package in the source folder of your catkin workspace show in the command below:

- cd /catkin_ws/src

Now that we already use a script to create a new package which depends on std_msg rospy and rocpp

- catkincreatepkg nameofthepkg stg_msg rospy roscpp

Now we build the package in the catkin workspace

- cd /catkin_ws
- catkin_make

To add the workspace to your ROS environnement you need to source the generated setup file with the command below:

- . /catkin_ws/devel/setup.bash

Your package is well created.

5.2 Check up of the ID servos

Now we start to check up the ID servos. Make sure that you install the arbotix package as follow:

- sudo apt-get install ROS-hydro-arbotix

Then check that your arm is effectively connected to your robot with:

- sudo chmod 777 /dev/ttyUSB0

When it's well connected execute

- arbotic_terminal

And finally to check that all your servos are all active use the ls command.

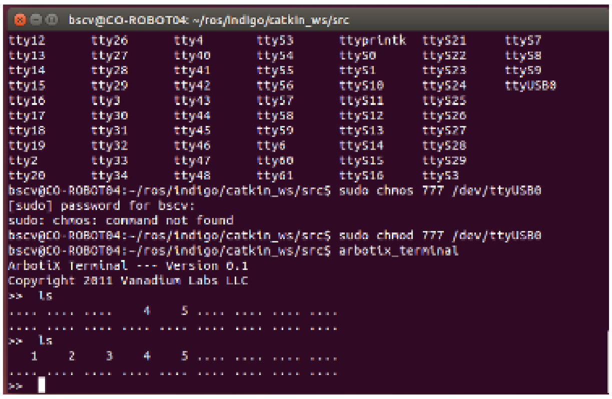

Figure 5.1: IDs servos test

Please notice that you have to open and run ROS in parallel. For this open arduino then

- file->sketchbook->ROS->verify->upload

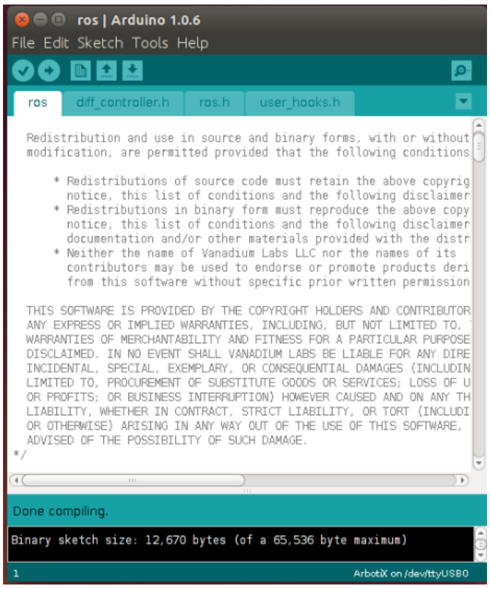

Figure 5.2: ROS

5.3 Test and develop the arm PhantomX Pincher Robot

In this part, we clone the turtlebot folder found in the github link: `https://github.com/turtlebot/turtlebot_arm` To help you to clone all the folder from the turtlebot_arm package use the command, in your terminal: gitclone and copy paste the URL link above. Be careful to clone it in the exact place (cd /catkin_ws/src).

When the turtlebot package is download you have to correct and add some files. First you have to check if the turtlebot_arm.yaml is the same as this one (see figure 5.3).

```
port: /dev/ttyUSB0
read_rate: 15
write_rate: 25
joints: {
arm_shoulder_pan_joint: {id: 1, neutral: 205, max_angle: 180, min_angle: -60, max_speed: 90},
arm_shoulder_lift_joint: {id: 2, max_angle: 150, min_angle: -150, max_speed: 90},
arm_elbow_flex_joint: {id: 3, max_angle: 150, min_angle: -150, max_speed: 90},
arm_wrist_flex_joint: {id: 4, max_angle: 100, min_angle: -100, max_speed: 90},
gripper_joint: {id: 5, max_speed: 90},
}
controllers: {
arm_controller: {type: follow_controller, joints: [arm_shoulder_pan_joint, arm_shoulder_lift_joint
arm_wrist_flex_joint], action_name: arm_controller/follow_joint_trajectory,  onboard: False }
}
```

Figure 5.3: yaml file

When this is done, create a new file inside the launch folder where you will put the new launch file with the following code:

```
<launch>

<node name="arbotix" pkg="arbotix_python" type="arbotix_driver" output="screen">
<rosparam file="$(find turtlebot_arm_bringup)/config/turtlebot_arm.yaml" command="loac
</node>

</launch>
```

Figure 5.4: new launch file

The code above is define to run the arbotix driver for the phantomX Pincher arm. When all the necessary modification is done execute:

- ros launch turtlebotarmbringup name_of_the_file.launch

```
SUMMARY
========

PARAMETERS
 * /arbotix/controllers/arm_controller/action_name: arm_controller/fo...
 * /arbotix/controllers/arm_controller/joints: ['arm_shoulder_pa...
 * /arbotix/controllers/arm_controller/onboard: False
 * /arbotix/controllers/arm_controller/type: follow_controller
 * /arbotix/joints/arm_elbow_flex_joint/id: 3
 * /arbotix/joints/arm_elbow_flex_joint/max_angle: 150
 * /arbotix/joints/arm_elbow_flex_joint/max_speed: 90
 * /arbotix/joints/arm_elbow_flex_joint/min_angle: -150
 * /arbotix/joints/arm_shoulder_lift_joint/id: 2
 * /arbotix/joints/arm_shoulder_lift_joint/max_angle: 150
 * /arbotix/joints/arm_shoulder_lift_joint/max_speed: 90
 * /arbotix/joints/arm_shoulder_lift_joint/min_angle: -150
 * /arbotix/joints/arm_shoulder_pan_joint/id: 1
 * /arbotix/joints/arm_shoulder_pan_joint/max_angle: 180
 * /arbotix/joints/arm_shoulder_pan_joint/max_speed: 90
 * /arbotix/joints/arm_shoulder_pan_joint/min_angle: -60
 * /arbotix/joints/arm_shoulder_pan_joint/neutral: 205
 * /arbotix/joints/arm_wrist_flex_joint/id: 4
 * /arbotix/joints/arm_wrist_flex_joint/max_angle: 100
 * /arbotix/joints/arm_wrist_flex_joint/max_speed: 90
 * /arbotix/joints/arm_wrist_flex_joint/min_angle: -100
 * /arbotix/joints/gripper_joint/id: 5
 * /arbotix/joints/gripper_joint/max_speed: 90
 * /arbotix/port: /dev/ttyUSB0
 * /arbotix/read_rate: 15
 * /arbotix/source: turtlebot_arm.yaml
 * /arbotix/write_rate: 25
 * /rosdistro: indigo
 * /rosversion: 1.11.20

NODES
```

Figure 5.5: message

You should see the following output show in figure 5.4. This message mean that your arm is not successfully connected and ready to receive motion command. The easiest way to control the servos using the arbotix_gui.

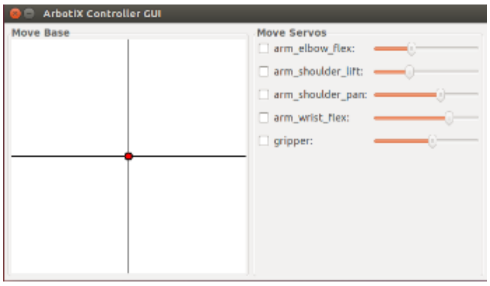

Figure 5.6: Arbotix gui command

With this command will appear the interface of the arbotix simulator (figure 5.6) which with you can check the good fonctionality of every servos.

Chapter 6

Conclusion and Perspectives

Having to complete this first part of the project we will say it was a good learning atmosphere. After assembling the PhantomX Pincher Arm Robot and testing it to perform specific tests and it responding with much satisfactory results. We can say our entering into this new world of technology has developed greater interest in us. Yes, we encountered many problems but we learnt a lot from that and made our next step an easy task for us has it came. We have now seen how the PhantomX Pincher operates hand in hand with its basic components and compatibles software in achieving and exploiting is capabilities. With our future studies, we will now be innovative with the PhantomX Pincher arm. We will try to develop ways to let this robotic arm perform certain tasks and try to connect it with the turtlebot in achieving specific tasks. Our goal is to let the robotic arm be a substitute to the human and at the end of the year we will be part of the world of achievers in robotics.

Chapter 7

References

1. **Joseph Engelberger, LEO - Center for Service Robotics**

   `http://www.leorobotics.nl/definition-robots-and-robotics`

2. **PhantomX Pincher Robot Arm Build Check**

   `http://learn.trossenrobotics.com/component/tags/tag/9-phantomx-pincher.html`

3. **Setting Dynamixel IDs with DynaManager**

   `http://learn.trossenrobotics.com/index.php/getting-started-with-the-arbotix/1-using-the-tr-dynamix`
   `&panel1-1`

4. **General help guide**

   `http://learn.trossenrobotics.com/38-interbotix-robots/122-phantomx-pincher-robot-arm.`
   `html`

5. **Assembly guide**

   `http://learn.trossenrobotics.com/16-interbotix/robot-arms/pincher-robot-arm/163-phantomx-pincher-r`
   `html`