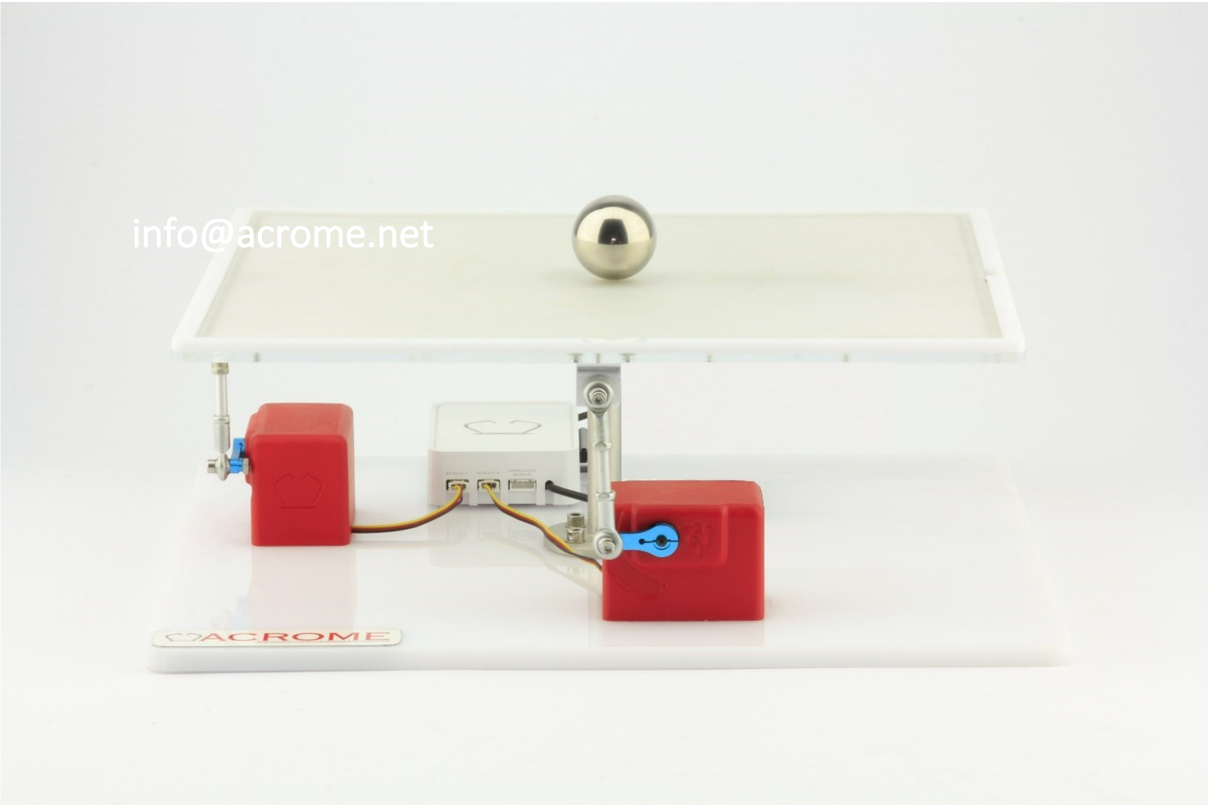


BALL BALANCING TABLE

GETTING STARTED

Version:2021.1

Acrome Ltd.
ITU ARI4 Science Park
Maslak, Istanbul
Turkey
info@acrome.net
Phone: +90 212 807 0456

Printed in Maslak, Istanbul

This document is prepared for using with Raspberry Pi™ platform. Examples are based on Python software.

For more information on the solutions Acrome Inc. offers, please visit the web site at:

http://www.acrome.net

Contents

List of Figures

1. OVERVIEW

1.1. System Description

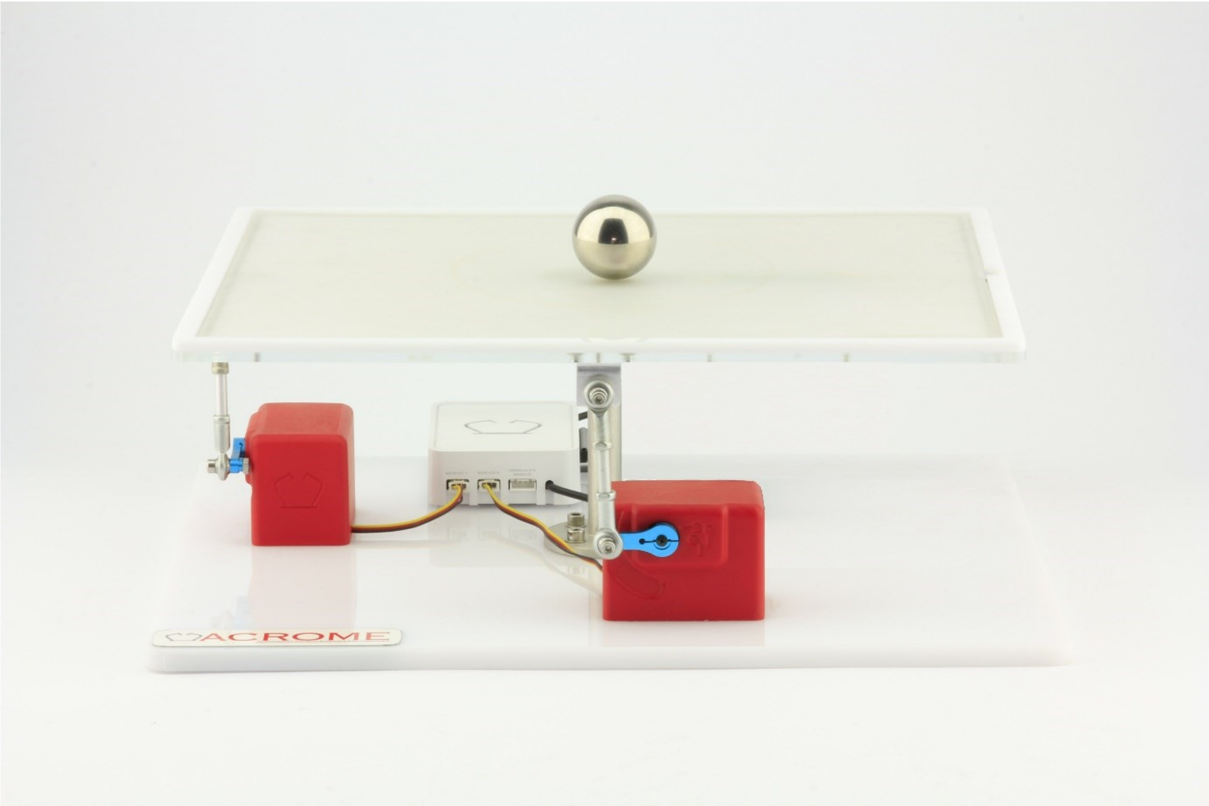


Figure 1.1 - ACROME Ball Balancing Table System

Ball Balancing Table has two rotational degrees of freedom. Each degree of freedom is controlled by an RC servo motor which is attached to the table. Raspberry Pi reads the position data of the ball from the 17'' touchscreen. Raspberry Pi also controls the RC servo motor angles by sending a PWM signal. Through an implemented PID algorithm, system controls the position of the ball by getting the position feedback and actuating the servo motors over PWM signals.

2. COMPONENTS

The components of ACROME Ball Balancing Table System are shown with identification numbers below.

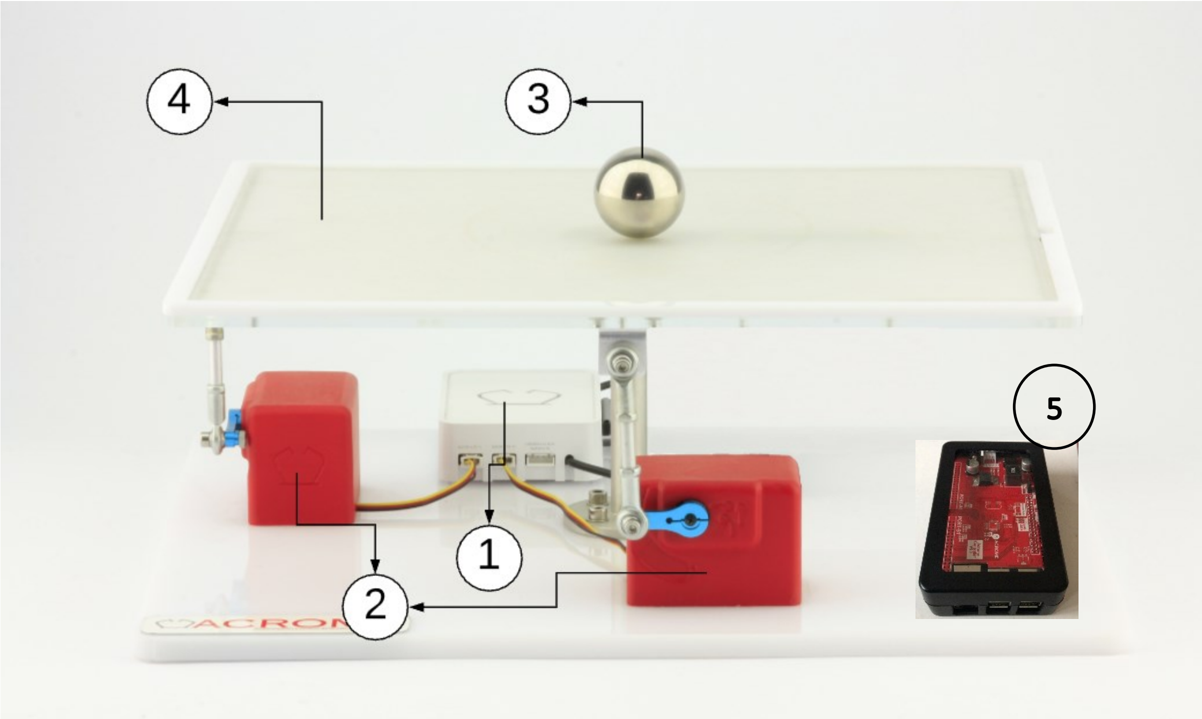


Figure 2.1 - Components of Ball Balancing Table System

1. ACROME Power Distribution Unit
2. RC Servo Motors
3. Steel Ball
4. Touch Screen
5. Raspberry Pi & Shield

2.1. ACROME Power Distribution Unit

The RC servo motor, sensor (potentiometer) and Arduino Mega connections are located on ACROME power distribution box which is shown in Figure 2.5. It also has a RGB led and switch mode regulators.

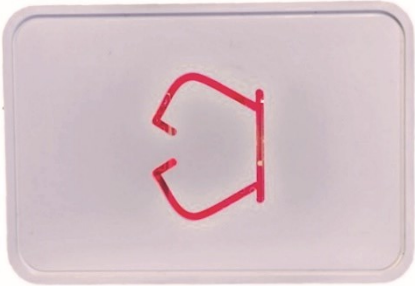


Figure 2.2 - ACROME Power Distribution Box

2.2. RC Servo Motor

RC servos are electromechanical devices that convert electrical signals to movement. They provide simple and handy solutions to most control and robotic applications.

Figure 2.3 - RC Servo Motor

2.3. Steel Ball

It is a magnetic steel ball, with a very-low friction constant.

Table 2.1: Steel Ball's Parameters

Definition	Value	Unit
Radius of the Ball	0.02	[m]
Mass of the Ball	0.26	[kg]

2.4. Touch Panel

It is a resistive touch panel with 17" diagonal dimension. A bridge measurement method is used to measure the X and Y positions of the ball. Below are the parameters of the touch panel:

Table 2.2: Touch Panel Parameters

Definition	Value	Unit
Length (X direction)	355	mm.
Width (Y direction)	288	mm.
Resolution in X direction	0.34	mm.
Resolution in Y direction	0.28	mm.

2.5.Raspberry Pi™ and ACROME RPi Shield

Raspberry Pi is a low cost, credit-card sized single board computer that is used in Remote Lab to control the mechatronic and robotic systems. Raspberry Pi is supported in Matlab®/Simulink® with version 2019.r1 and above, enabling this software to compile a control algorithm and run on the Pi headless.

Figure 2.4 - Raspberry Pi (left) is used with a custom designed shield card (middle).

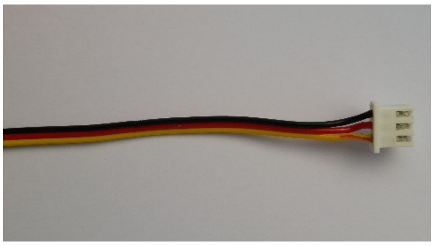
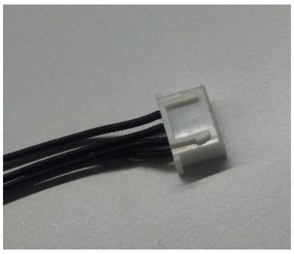
The ACROME RPi shield has an on-board microcontroller and used to measure analog and digital signals of the experiment systems' sensor and actuator signals. Shield is connected to RPi over serial communication lines and forms a bi-directional link with RPi. The potentiometer value is measured, PWM based motor speed position commands are generated within the RPi shield board. A UART communication channel transfers the data between the RPi and the microcontroller.

3. WIRING

3.1. Cable Names

Cables which are used in the ball balancing table are defined and denominated as seen in Table 3.1 below.

Table 3.1: Cable Descriptions

Cable Name	Cable Picture	Definition
1. Ethernet Cable		This cable provides the connection between the PC/network switch and Raspberry Pi.
2. RC Servo Motor Cables		These cables which transmit PWM signals connect RC servo motor to power distribution box.
3. Ribbon Cable		This cable transmits signals between Raspberry Pi Shield and power box.
4. Touch Screen Cable		One terminal of this cable enters the touch screen. The other terminal connects to Raspberry Pi Shield in order to read the position of the ball.

3.2. Connections

In order to use the system, all cables mentioned above should be connected properly. All the essential connections among the components are shown in Figure 3.1.

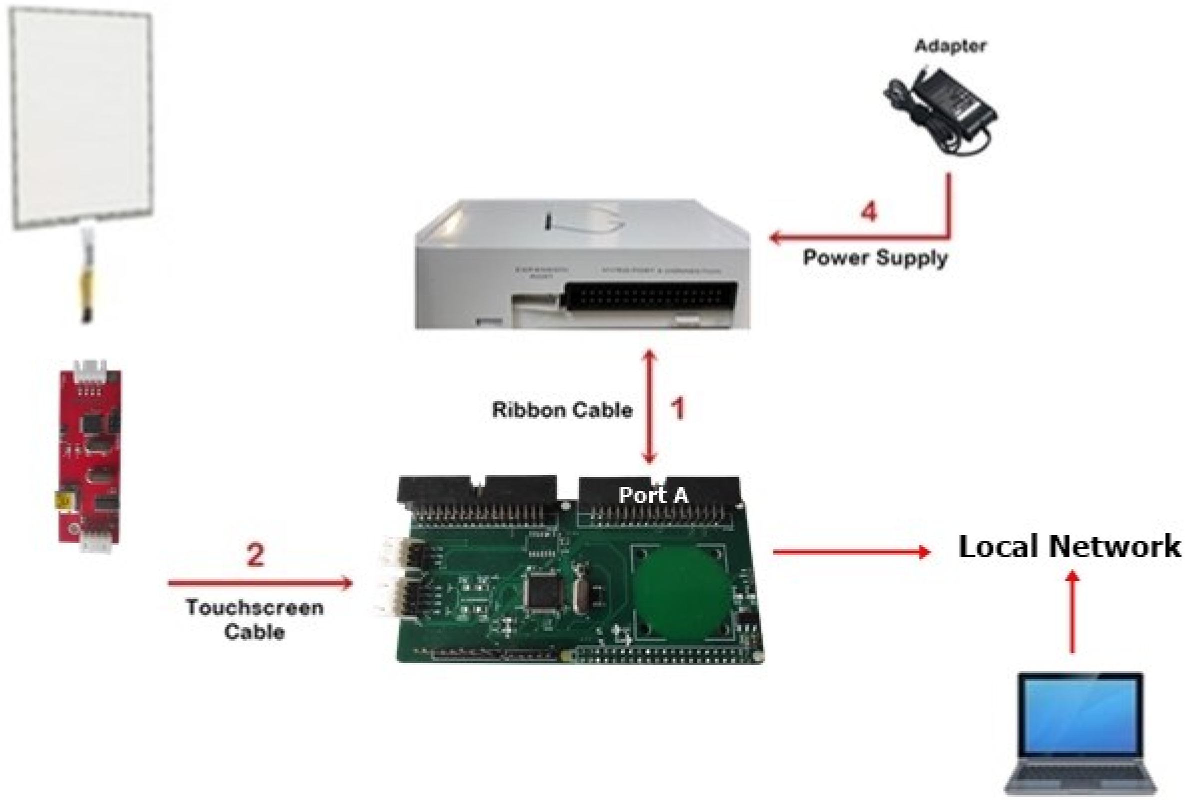


Figure 3.1 - Ball Balancing Table Cable Connections

4. GETTING STARTED WITH THE SYSTEM

4.1.Software Installations and Setup

After the connections are done successfully, please install a network scan tool and an SSH client for remote connection to Raspberry Pi.

We use Advanced IP Scanner (available at https://www.advanced-ip-scanner.com/) and MobaXterm (available at https://mobaxterm.mobatek.net/) in this manual, but any software with similar functionality can be used.

MACBOOKPRO-D63E	192.168.1.109		A0:78:17:77:D6:3E
HPB0227A52F8B5	192.168.1.211		B0:22:7A:52:F8:B5
DESKTOP-JOSONTM	192.168.1.108		56:18:64:49:AA:35
192.168.1.121	192.168.1.121		D4:5D:64:7F:3D:A0
192.168.1.122	192.168.1.122	REALTEK SEMICONDUCTOR CORP.	00:E0:4C:36:F0:A4
192.168.1.104	192.168.1.104	TP-LINK TECHNOLOGIES CO.,LTD.	28:EE:52:93:CF:B1
192.168.1.102	192.168.1.102	TP-LINK TECHNOLOGIES CO.,LTD.	70:4F:57:11:09:75
192.168.1.112	192.168.1.112		26:F0:EF:32:1C:0A
192.168.1.103	192.168.1.103	Raspberry Pi Foundation	B8:27:EB:B9:2E:D0
192.168.1.2	192.168.1.2		A0:78:17:94:B1:48
Lenovo	192.168.1.111		B0:60:88:D4:C4:C0

Figure 4.1 – Use an IP scanner software to find out the IP Address of the Raspberry Pi in your network.

After acquiring the IP address, start an SSH connection to the Raspbbery Pi. Follow below image to see how it is done.

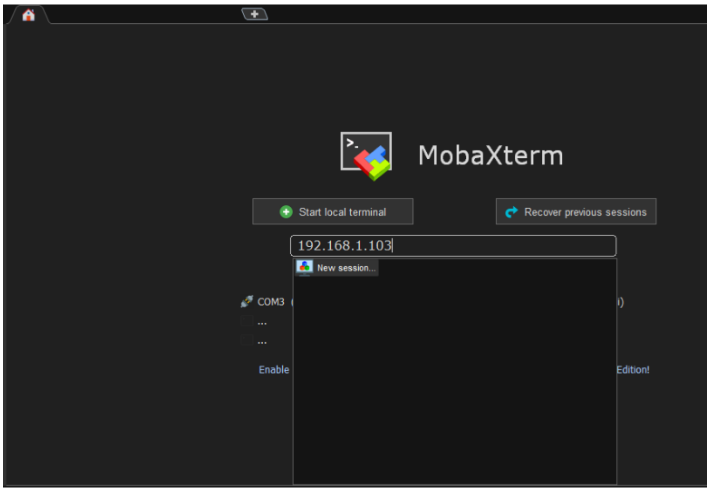


Figure 4.2 – Enter the IP address of the Raspberry Pi and click new session to start an SSH connection

When asked, please enter “**Pi**” as the login username and “**acromerobotics**” (without the “ “ marks) as the password.

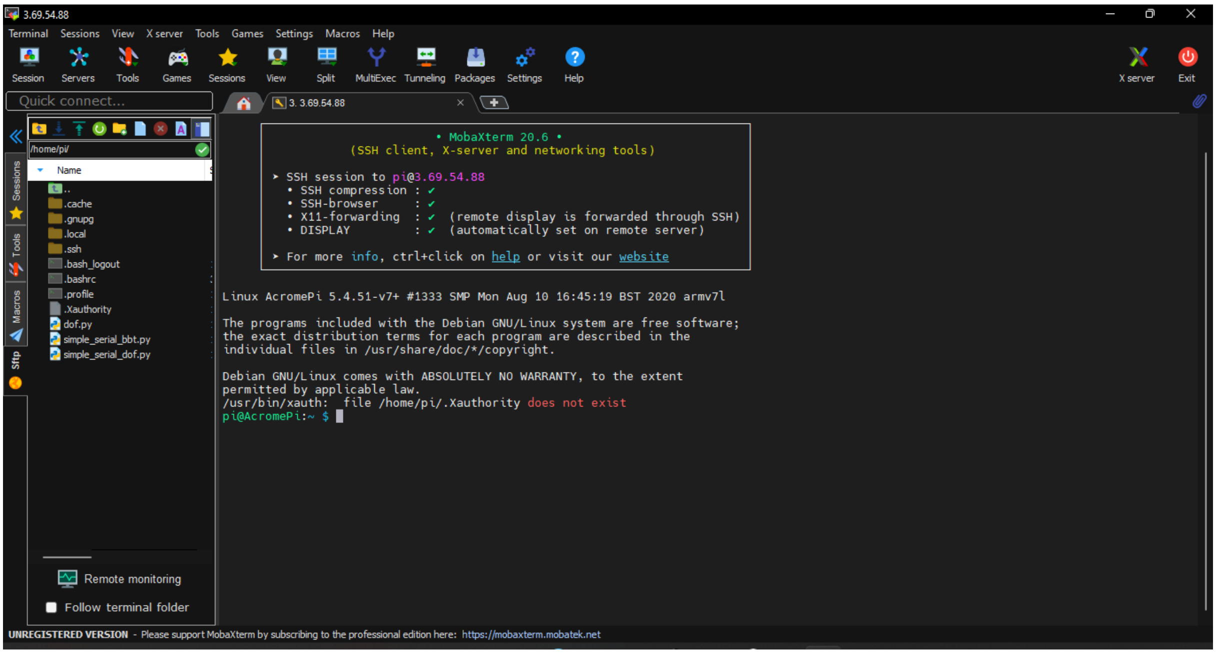


Figure 4.3 - If SSH connection is succesful then you should reach the user terminal of the Raspberry Pi

Next, we should install the necessary Python library for communication between the Raspberry Pi and ACROME Raspberry Pi shield. Enter the following command to the terminal window:

pip3 install remotelab

(Please note that the same library is also used in ACROME’s Remotelab setups, hence the naming is identical.)

To obtain the example Python code, please clone the Github repository of the Ball Balancing table with the following command:

git clone https://github.com/Acrome-Remote-Laboratory/bbt.git

You should have a terminal output similar to below screenshot.

```
pi@AcromePi:~ $ pip3 install remotelab
Looking in indexes: https://pypi.org/simple, https://www.piwheels.org/simple
Collecting remotelab
  Using cached https://files.pythonhosted.org/packages/57/dc/c41eef6f8615d7c53d982ecb9c335d7290a0fc497604a3ac1de9df579e32/remotelab-0.0.6
-py3-none-any.whl
Requirement already satisfied: pyserial in ./.local/lib/python3.7/site-packages (from remotelab) (3.5)
Requirement already satisfied: crccheck in ./.local/lib/python3.7/site-packages (from remotelab) (1.1)
Installing collected packages: remotelab
Successfully installed remotelab-0.0.6
pi@AcromePi:~ $ git clone https://github.com/Acrome-Remote-Laboratory/bbt.git
Cloning into 'bbt'...
remote: Enumerating objects: 39, done.
remote: Counting objects: 100% (39/39), done.
remote: Compressing objects: 100% (29/29), done.
remote: Total 39 (delta 8), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (39/39), done.
pi@AcromePi:~ $
```

Figure 4.4 - Requirement packages installed successfully

4.2. Running the Example Python Code

1. Run the example code with the following command:

   cd bbt/src/

   python3 bbt.py

   You should receive a terminal output like the following:

```
31.6666666666667 30.3333333333333
31.6666666666667 30.3333333333333
31.6666666666667 30.0
31.6666666666667 30.0
31.6666666666667 30.0
31.6666666666667 30.0
31.6666666666667 30.0
31.6666666666667 30.0
31.6666666666667 30.0
31.3333333333333 30.0
31.3333333333333 30.0
31.3333333333333 30.0
31.3333333333333 30.0
31.3333333333333 30.0
31.3333333333333 30.0
31.3333333333333 30.0
31.3333333333333 30.0
31.3333333333333 30.0
31.3333333333333 30.0
31.3333333333333 30.0
31.3333333333333 30.0
31.3333333333333 30.0
31.3333333333333 30.0
31.3333333333333 30.0
31.3333333333333 30.0
31.3333333333333 30.0
31.3333333333333 30.0
31.3333333333333 30.0
```

Figure 4.5 - Ball's position is written to the terminal 10 times in each second.

2. You may modify the ball's target location by changing the "setpointx" and "setpointy" variables in the bbt.py example source file.

```
#Setpoint for the ball. You may change between -/+ 100 [mm] both in X and Y.
setpointx = 0
setpointy = 0
setpointx = (setpointx + 50) * 3 + 250 #scaleX
setpointy = (setpointy + 60) * 3 + 250 #scaleY

# Main Loop
while True:
    positionx, positiony = bbt.get_position()
```

Figure 4.6 - Desired location of the ball can be changed with setpoint variables.

3. Also the PID coefficients can be changed in this section to achieve different response types and different controller types such as P,PD, PI controllers.

```
#Control Parameters
feedforwardx = 725
feedforwardy = 725
windup_abs = 30
calibration_x = (0,-50)
calibration_y = (0,-8)

kpx = 0.45
kix = 0.001
kdx = 0.2

kpy = 0.45
kiy = 0.001
kdy = 0.2
```

Figure 4.7 - P,I,D coefficients for each X and Y planes.

ACROME Robotik Mekatronik Sistemleri San. ve Tic. Ltd. Şti.

İTÜ Ayazağa Kampüsü

Koru Yolu ARI 4 Binası B204

For further information on ACROME equipment please contact.

Website: http://www.acrome.net/

e-mail: info@acrome.net

Telephone: 0212 807 0 456