Using LEGO NXT kit for Control System Laboratory

S.R.Manikandasriram, *Student Member, IEEE,*

***Abstract*—This report analyses the scope of using the LEGO Mindstorms NXT Kit as a platform for teaching Control Engineering concepts for both Undergraduate and Graduate students. A set of simple experiments using only a LEGO DC Motor and the "NXT Intelligent Brick" have been studied. The experiments cover the introductory concepts in Control Engineering like Transfer Function, PID Control, System Identification and Bode Plots and is thus ideal for an Undergraduate Laboratory course on Control Systems. The later half of the report deals with building a 2-wheel Self-balancing Robot using LEGO Mindstorms kits and a HiTechnic Gyro sensor. The mathematical model of the 'Segway' type robot is explained in detail and a Servo PID control for stabilising the 'Inverted Pendulum' is studied. This would be ideal as a Course Project for first year Masters students.**

I. Introduction

THE use of LEGO Mindstorms series of kits as a platform for education has received widespread acceptance since the release of Mindstorms Robotics Invention System in 2000. The LEGO Mindstorms NXT kit is relatively cheap, robust, customizable, reprogrammable and induces enthusiasm and creativity in students. The official software for LEGO NXT is a NI LabView based programming language called NXT-G. However, enthusiasts have extended the software and hardware in various ways to make the LEGO NXT kits as a favourable Rapid prototyping platform for academic and research activities. In this report, we analyse the scope of using LEGO NXT kits for teaching Control Engineering.

The first half of the report explains relatively simple control experiments for analysing a DC Motor system. The analysis is divided into 4 parts - 1. Observing the in-built PID Controller, 2. Writing custom PID controller to meet design requirements, 3. Determining the Transfer Function of the DC Motor and 4. Designing PID controller using the derived Transfer Function. In order to avoid students from having to get used to a new programming language, ROBOTC - a C-like programming language developed by Carnegie Mellon University's Robotics Academy - will be used as software for this section.

The second half of the report details the building of a 2-wheel Self-balancing robot using the LEGO NXT kit and a HiTechnic Gyro Sensor. The 2-wheel Self-balancing robot can be modelled as a *2D Inverted pendulum* system which is a well studied classic problem in Control Engineering. While the system agrees well with this relatively simple mathematical model, the Inverted Pendulum system is a widely used example in explaining many concepts in Control Engineering. This makes it ideal for a 1 month course project in a first year Masters programme in Control Engineering. The analysis is divided into 3 parts - 1. Mathematical Modelling, 2. Servo PID Controller design and 3. State Feedback Controller and other advanced controllers. MATLAB provides official Simulink support package for LEGO Mindstorms NXT hardware which allows the students to use the various toolboxes available in Simulink and deploy the model to the LEGO NXT hardware.

II. Observe PID Controller in action

In this first experiment, the students observe the functioning of the in-built PID Controller available in ROBOTC. The system under analysis is comprised of an *Interactive Servo motor* which is controlled by Pulse Width Modulation by the NXT Intelligent Brick. The feedback is provided by the *rotary encoder* present inside the *Interactive Servo motor* which gives 360 counts per single revolution of the motor shaft. The block diagram for the PID controller is shown in Figure 1. Using the feedback from the encoders, the "*NXT Intelligent Brick*" calculates the angular position of the motor (with 1 degree accuracy) and also the instantaneous speed of the motor. The formula used for speed calculation is

$$speed = \frac{\Delta\theta}{\Delta t}\text{degrees/sec} \tag{1}$$

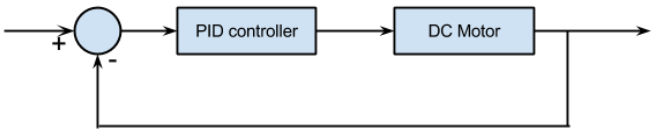

Fig. 1. Block Diagram of the in-built closed loop PID Controller

The LEGO motors are assigned a power rating between -100 and 100 with a negative value denoting reverse direction. Under ideal conditions, the maximum speed of the LEGO motors is 1000 degrees per second when the power rating is set at 100%. But this value drops with decrease in battery voltage and increase in motor load.

To enable PID Speed Control and set the maximum regulated speed, the following commands were used

```
nMotorPIDSpeedCtrl[motorB] = mtrSpeedReg;
nMaxRegulatedSpeedNxt = 750;
```

The in-built PID controller provides consistent speed by continuously adjusting the raw power sent to the motor.

The LEGO NXT has a *Datalog* feature which allows sensor data and status information to be stored in memory as `(Key,Value)` pairs, which can later be exported to a PC for post-processing. In this experiment, the encoder counts

and Motor PWM level are logged every Δtms. The ROBOTC code for this experiment is given below as the motive of this experiment is to get the students acquainted with the software and hardware interfaces.

```
task main()
{
  nMaxRegulatedSpeedNxt = 500;
  // Reset the Motor Encoder
  nMotorEncoder[motorB] = 0;
  nMotorPIDSpeedCtrl[motorB] = mtrSpeedReg;

  int motorRAWpower, motorDegrees;
  motor[motorB] = 50;     // 50/100
  time1[T1] = 0;          // Timer

  // Allocate memory for datalog
  nDatalogSize = 1600;

  while (time1[T1] < 10000) {
    motorDegrees = nMotorEncoder[motorB];
    motorRAWpower = motorPWMLevel[motorB];
    // store value to Datalog
    AddToDatalog(1,motorPIDdegrees);
    AddToDatalog(2,motorPIDpower);
    wait1Msec(50);
  }
  motor[motorB] = 0;    // Stop the motors
  SaveNxtDatalog();
}
```

In order to observe the PID control action, the students are asked to run the above program and apply friction on the motor. The in-built PID controller will then step in and increase the motor PWM level in order to maintain the speed. This can also be quantitatively verified from the log file. The datalog file (which would be named as *DATAnnnn.rdt*) can be exported to a PC using the *File Management Utility* available in the ROBOTC Development Environment under *Robot→NXT Brick→File Management Utility*. The *Spreadsheet Upload* feature transfers the datalog file to the PC and additionally converts the *.rdt* file into a *.csv* file which can then be processed using any Spreadsheet Processors.

As mentioned earlier, the data is stored as *Key,Value* pairs which constitute the two columns in the exported CSV file. The datalog sequentially stores data with every call to the `AddToDatalog(<index>,<data>);` creating a new entry in the file with a *key* which is proportional to the *index* and the *data* stored as a $16-$bit unsigned integer. Due to this implicit type casting, negative values would get "wrapped" around. Hence appropriate post-processing has to be done to handle the same.

The students can now plot the speed in degrees per second and motor PWM level in a single graph to observe the PID action. One such plot is shown in Figure 3.

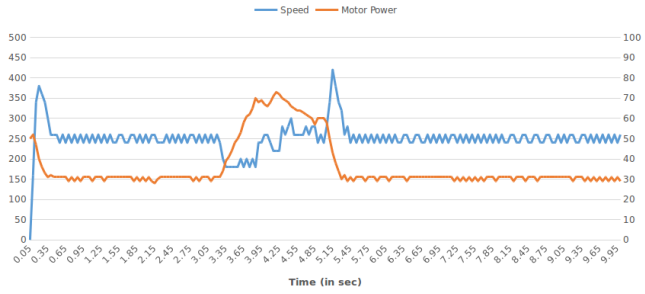

Fig. 3. Plot of Speed vs PWM level of the LEGO motor

As can be seen from the plot, there is a dip in the speed of the motor between 3 and 5 seconds which caused an increase in the PWM power level applied to the motor. When the friction is removed, the speed shoots up due to higher power level before settling down to the desired speed of 250 deg/sec.

III. Conclusion

The conclusion goes here.

Fig. 2. Experimental setup for observing in-built PID Controller

Appendix A
Zeigler-Nichols Tuning

- Set Ki and Kd to zero.
- Slowly increase Kp to a value Ku at which sustained oscillations - constant amplitude and periodic - are observed.
- Note period of oscillation
- Refer table below for initial values

Z-N Model	Kp	Ki	Kd
P controller	0.5*Ku	0	0
PI controller	0.455*Ku	0.833*Tu	0
PID controller	0.588*Ku	0.5*Tu	0.125*Tu

Since Z-N method usually results in aggressive tuning, alternatively Tyreus-Luyben method can be adopted.

T-L Model	Kp	Ki	Kd
PI controller	0.312*Ku	2.2*Tu	0
PID controller	0.454*Ku	2.2*Tu	0.159*Tu

ACKNOWLEDGMENT

The authors would like to thank Sabiha A Wadoo and Rahul Jain from Department of Electrical and Computer Engineering, New York Institute of Technology, USA. Their paper "A LEGO based Undergraduate Control Systems Laboratory". The first half of this report is an adaptation of their proposed method.

REFERENCES

[1] J.B. Weinberg and Xudong Yu, "Robotics in education: Low-cost platforms for teaching integrated systems," *IEEE Robot. Automat. Mag.*, vol. 10, no. 2, pp. 46, 2003.