To Perform SIL and PIL Testing On Underactuated System Using Arduino

Ashish V. Patel[1], Prof. Keerti Vashishtha[2]

1 Student, Dept. of Instrumentation And Control, L. D. College of Engg. Gujarat, India **ashishadvance@hotmail.com**
2 Prof, Dept. of Instrumentation And Control, L. D. College of Engg. Gujarat, India **kpbs81@rediffmail.com**

Abstract- Embedded Systems is indubitable the core of the current electronic era. From simple calculators to high-end control of the spacecraft it plays the key role. With multi-domain applications, it is also used extensively in control domain. Applying Rapid Control Prototyping (RCP) methodology in controller development in model based design shorten the development period and provide higher accuracy compared to normal flow. This paper provide the economical way to perform the SIL (software in the loop) and PIL (processor in the loop) testing with Arduino, an open source hardware, compare to high-end expensive modules available in market. This paper focuses on performing SIL and PIL testing on Furuta pendulum using Arduino, which is an open source hardware.

I. INTRODUCTION

With the development of semiconductor industries and with Moore's law, the hardware in embedded system becomes more complex with time. To extract every capability from controller or processor require the detailed understanding of architecture and with numbers of vendors this varies and learning for all controllers it an impossible task. But with hardware, the Computer Aided Control System Design (CACSD) like MATLAB, LABVIEW develop into great extend. With using this CACSD software, dependency on architecture knowledge, can be eliminate to greater extent.

Traditional controller development method has separated processes such as controller design, implementation, test and verification [1]. The whole process has to be started over again when an error or deviation occurs, as all processes are separated its sometime not possible to find the origin of the problem, making the controller development process costly and time consuming.

With RCP methodology, which include MIL, SIL and PIL, all the process development integrated with each other allowing engineers to quickly test and iterate their control strategies with real hardware (controller). RCP methodology is highly used in Model based design as it eliminates potential bottlenecks, accelerate the development, and save cost and time [2]. It has been widely used in application like anti-lock braking, flight control, servo-control, vehicle stability and medical device development [2].

The Underactuated system chosen for the test is Furuta pendulum also known as rotary inverted pendulum (RIP). It was invented in 1992 at Tokyo Institute of Technology by Katsuhisa Furuta and his colleagues[6] and consider a classical problem in control. In Underactuated system the degree of freedom is greater than actuation hence make them highly non-linear. Other examples of Underactuated system include Cart-pole, Acrobot, Inertia wheel etc.[6] Its application in robotics used for development in humanoid.

But tuning the controller directly on plant or on its proto-type may cause damage because it may lead to unsafe operating point. So with plant parameter we can design controller and simulate it to verify its behaviour to all possible point without any risk.

The major advantage of RCP over the traditional method is that once the controller is designed and verified in simulation environment, we need not to develop software/code for controller in C/C++ or any other language with Auto code generation facility code for control strategy are generated automatically [3]. It saves a lot of time and useful to industries because it reduces the time-to-market which is one of the most important factor considering the high competition in these era.

ARDUINO UNO, containing AVR Atmega 328, have been used which is an open source hardware developed by Massimo Banzi and David Cuartielles, Italy [5].

II. RCP METHODOLOGY

As RCP methodology applies to many field of applications its structure gets modified accordingly but typical rapid control prototyping system is comprised of a math modelling program, a host computer, and a target hardware (controller) [4]. Here we have used MATLAB for modelling of the system. Below is the flow of RCP methodology,

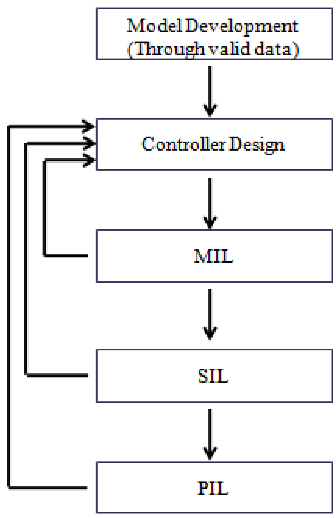


Fig.1:RCP Methodology

Because of multiple feedbacks the error propagation from one stage to another is eliminated. From the valid data the model of the system is developed in the modelling environment i.e. software like MATLAB/SIMULINK. The controller is then designed depending on control strategy to be implemented. The controller is then tested in a simulation environment and this process is called MIL (Model In the Loop). After verified output and satisfactory behaviour the C equivalent code for the controller is generated through Auto-code generation facility.

A. Software In the Loop (SIL)

C code generated are need to be verified with a real controller design to verify the behaviour of the generated code with real controller, thus eliminating error propagation to next stages. Below is a block diagram of SIL testing.

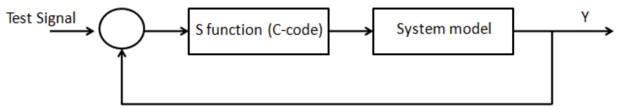


Fig.2: SIL Testing Block Diagram

The SIL testing is done by S-function generation from the controller in simulation environment. The S function represents the Generic C code generated; when we perform the SIL the local C compiler of the MATLAB [7] executes testing the S function block. Thus in SIL testing we basically works in two domains one C and other Simulink. The output of the simulation is compared with that of the MIL output and controller is redesign if both do not converge to predetermine tolerance. This allows the controller modification at an early stage of development and it also eliminates fix-point and floating point application error for algorithm implementation.

B. Processor In the Loop (PIL)

SIL verification is done with generic C code only, but ultimate application required to run on the microcontroller. So the C codes are then compiled with the IDE compiler for this we need to integrate the IDE with MATLAB. Thus PIL provides a good verification of compiled object code on the target platform to ensure functional equivalence of code running on the target processor.

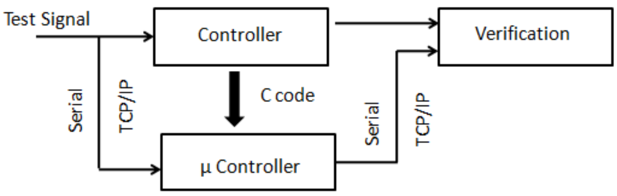


Fig.3: PIL Testing Block Diagram

As seen from block diagram, PIL testing is similar to SIL. The key difference is that, during PIL, code executes on the target processor. The communication between model and µ-Controller is achieved through serial or TCP/IP communication [7]. The advantages of the PIL testing that we can completely know the behaviour of the code in an embedded environment. MIL, SIL testing is done by the processor of the computer which is either 32 or 64 bit but ultimate application is to be run in 8 bit µ-controller so provide a better idea about the controller behaviour and take necessary step before connecting it to real system.

III. MODEL DEVELOPMENT AND CONTROLLER DESIGN

Furuta pendulum consist of a DC motor and 2 arms as shown in fig.4 [6].One arm moves in vertical while other in horizontal, vertical arm moves free without any actuation horizontal arm is actuated by DC motor. So we have 2 DOF and 1 Degree of actuation making it 1 degree of underactuation, making it a non linear system.

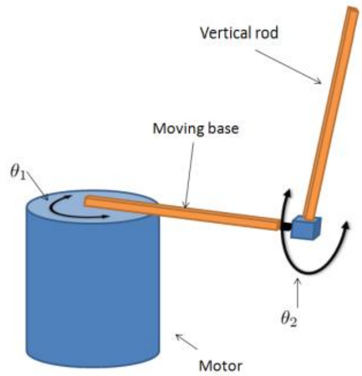


Fig.4: Furuta pendulum [8]

Below figure shows the simplified diagram of the RIP

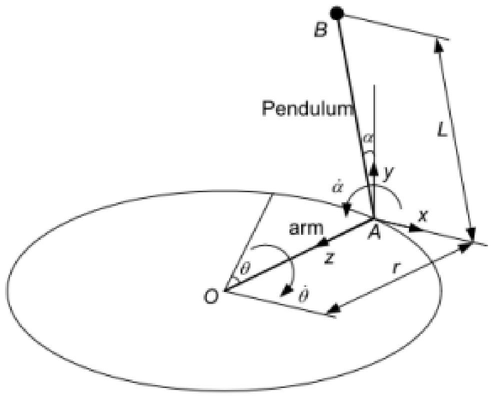


Fig.5 Simplified model of RIP [9]

A. RIP Dynamics

The governing differential equations of the RIP [8][9] [10] at Fig. 5 are as follows:

$$a\ddot{\theta} - bcos\alpha\ddot{\alpha} = cV_m - d\dot{\theta} - \dot{b}sin\alpha.\dot{\alpha}^2 \quad (1)$$

$$-bcos\alpha\ddot{\theta} + e.\ddot{\alpha} = f.sin\alpha \quad (2)$$

Where:

$$a = J_{eq} + mr^2 + \eta_g + K_g^2 J_m$$

b= mrL;

$$c = \frac{\eta_g \eta_m K_t K_g}{R_a}$$

$$d = \frac{\eta_m \eta_g K_m K_t K_g^2}{R_a} + B_{eq}$$

$$e = J_g + mL^2 ; f = mgL$$

α : Pendulum position , $\dot{\alpha}$: Pendulum velocity

$\ddot{\alpha}$: Pendulum acceleration

θ : arm position , $\dot{\theta}$: arm velocity

$\ddot{\theta}$: arm acceleration

When solved for $\ddot{\alpha}$ and $\ddot{\theta}$ for equation (1) and (2) it results in nonlinear solution. This paper only discusses the stabilizing pendulum in inverting position, for stable position α is small so $\sin\alpha \sim \alpha$ while $\cos\alpha \sim 1$. For smaller α , $\ddot{\alpha}$ is negligible and applying this in equation (1) and (2) we get linearized equation.

$$\ddot{\alpha} = \frac{a.f}{h}\alpha - \frac{d.b}{h}\dot{\theta} + \frac{c.b}{h}V_m \qquad (3)$$

$$\ddot{\theta} = \frac{b.f}{h}\alpha - \frac{d.e}{h}\dot{\theta} + \frac{c.e}{h}V_m \qquad (4)$$

$$where, h = a.e - b^2$$

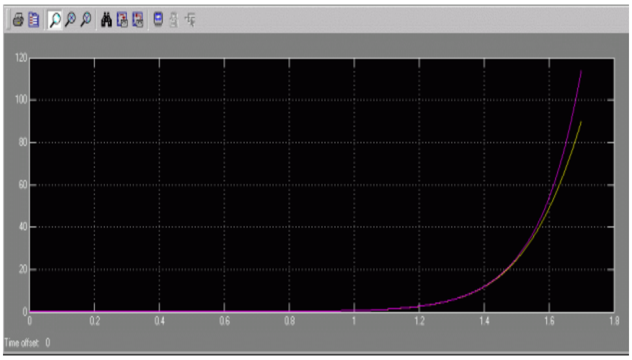


Fig.6 Simulation of Nonlinear and linear system

From above figure, where pink line indicate nonlinear system and yellow shows linear system output respectively, we can conclude that Non-linear and linear system shows same characteristic within 18-20 degree from stable point.

B. Controller Development

After the both models have been developed and compare, result into similar output within certain linear range near stage point.

So control strategy developed for linear system can be applied to nonlinear also. PID control is developed with help of PID block available in control tool box in Simulink.

The advantage of using this tool box is the auto tuning facility available saving time and effort in finding right value of P, I and D.

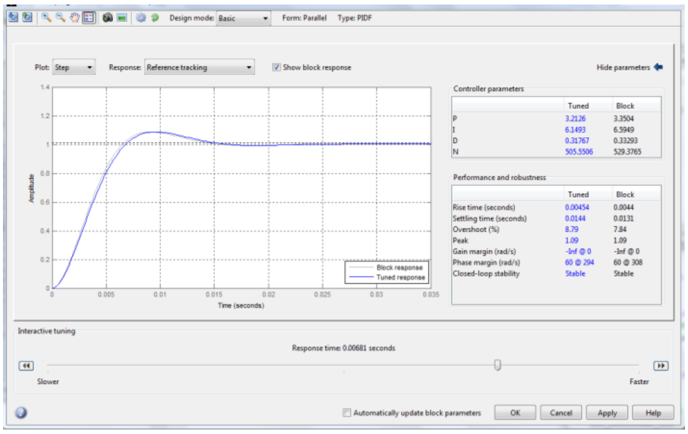


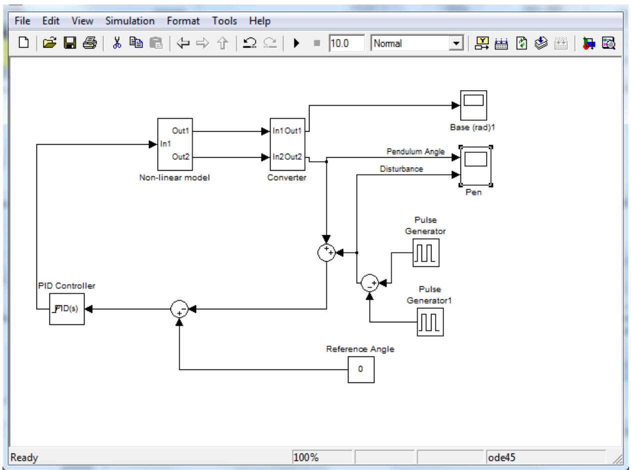
Fig.7: PID Tuner Window

As seen from fig.7 to tune the controller output user needs to change the slider and tune the response accordingly.

IV. TESTING AND VERIFICATION

A. MIL Testing

The Simulink block diagram for MIL testing is shown



in figure.

Fig.8: RIP Model for MIL Testing

RIP is open loop unstable system at inverted position for that value in predefines set to 0 degree. To check effectiveness of the controller disturbance in form of pulse of amplitude $\pm 5^{\circ}$ was given to system.

Below diagram, show the result of system with controller.

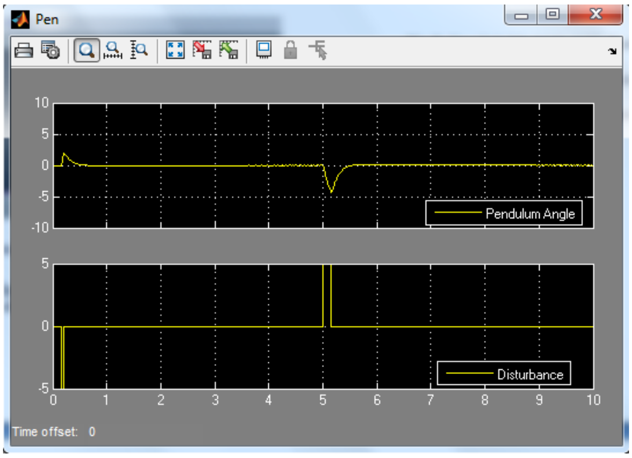


Fig.9: MIL Output

B. SIL Testing

As in RCP flow after MIL, SIL testing is done the Simulink block diagram for the SIL Testing shown. Fig. Simulink Block Diagram for SIL Testing .

The Green Box indicated the S-function generated.

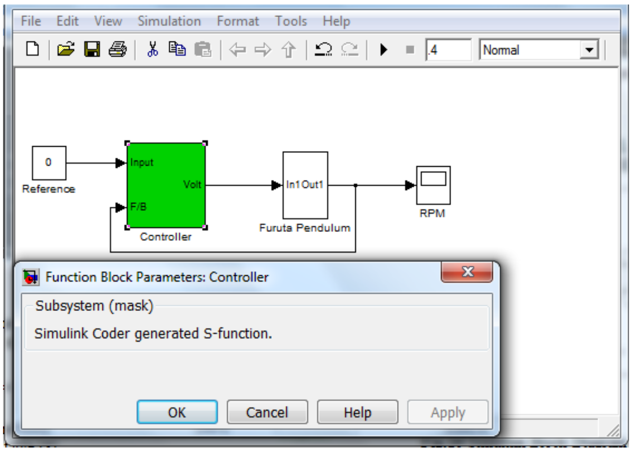


Fig.10: Simulink setup for SIL setup

After the simulation is done the data is logged with help of data logger. With the help of Simulink data inspector the output of MIL and SIL is compared which is shown in Fig.11

As seen there is no difference between the controller's output of MIL and SIL thus it ensure that the controller is translated correctly.

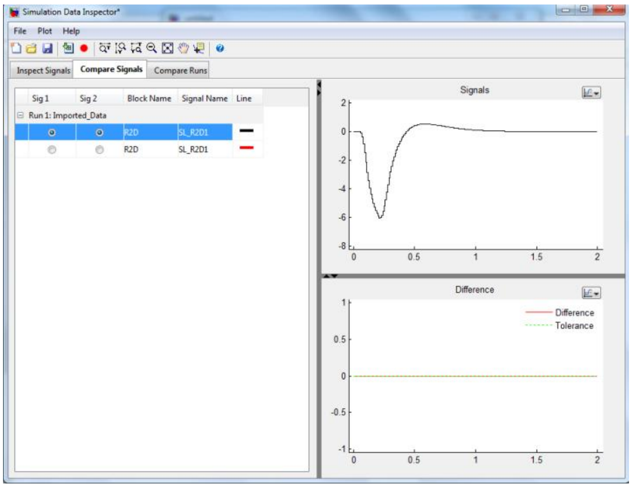


Fig.:11 Data Inspector Output

C. PIL Testing

As stated PIL test requires the embedded controller we have used Arduino Uno and integrate its IDE with MATLAB.

Arduino is an open source hardware board; it has Atmega328 AVR Controller, which is an 8-bit µ controller with 16 MHz operational frequency [5]. Data from the Arduino is retrieved through serial communication. For serial communication, the board has on board chip for serial communication capability.

PIL testing requires 2 Simulink programs

1. At Tx end i.e. for Arduino, having control strategy and communication setup. Shown in fig. 12.
2. At host (Rx) computer having MIL tested controller and Arduino Tx-Rx block for data transmission

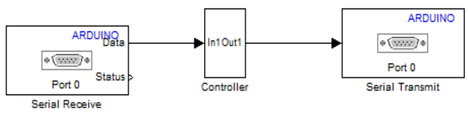


Fig.12 Simulink Model for Arduino

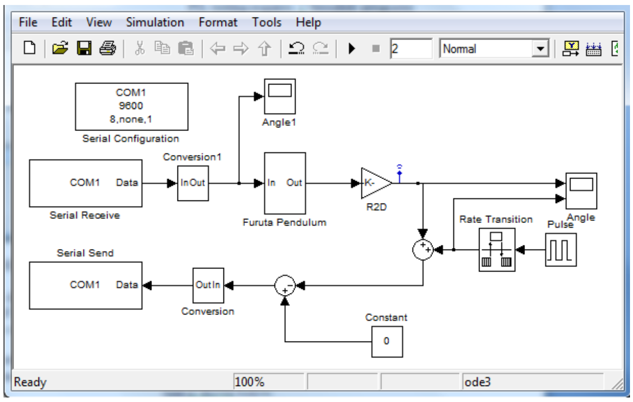
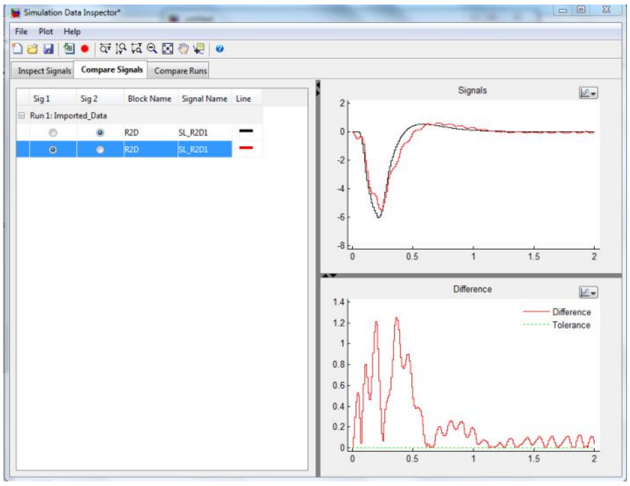


Fig.13 Simulink Model for Host PC

For verification purpose this paper shows a modified way to perform PIL testing, the output obtain after MIL is saved in workspace, same signal is send to Arduino through the Tx/Rx block. Arduino process the data as programmed and send data back to PC as shown in Fig 13.

So instead of comparing the output of the controllers only as traditional method, a closed loop of real controller (Arduino) and virtual system in MATLAB is formed and output is stored into workspace.

These data are compared with help of data inspector



as shown in Fig.14.

Fig.14 Difference between MIL and PIL

As seen above the data varies for maximum of 1.3 degree, which is acceptable comparing high-end processor in PC v/s 8-bit µ controller. This error is also elevated by the fact that serial communications took place in uint8 only and not in float.

V. CONCLUSION AND FUTURE SCOPE

By employing RCP methodology, the code for complete embedded control is automatically generated. No hand coding is required and generated code can be verified with simulated output.[11] The PIL testing ensures that the code generated is compatible and can be executed independently in AVR controller. The output obtained was within acceptable limits.

Because of low cost hardware, it can be used by students at university to implement various control strategies for other benchmark problems like 2 tank system, DC motor etc.

REFERENCES:

[1] Junwon Jang; Choon Ki Ahn; Sekyung Han; Wook Hyun Kwon, "Rapid Control Prototyping for Robot Soccer System using SIMTool," *SICE-ICASE, 2006. International Joint Conference* , vol., no., pp.3035,3039, 18-21 Oct. 2006

[2] PrecisionMBA, LLC. "Rapid Control Prototyping," Available: *http://www.precisionmba.com/rapid_control_prototyping.htm*

[3] Duma, R.; Petreus, D.; Sita, V. I.; Dobra, P.; Rusu, A., "Rapid Control Prototyping toolbox for Renesas M32C87 microcontroller," *Automation Quality and Testing Robotics (AQTR), 2010 IEEE International Conference on* , vol.1, no., pp.1,6, 28-30 May 2010

[4] Duma, R.; Dobra, P.; Abrudean, M.; Dobra, M., "Rapid prototyping of control systems using embedded target for TI C2000 DSP," *Control & Automation, 2007. MED '07. Mediterranean Conference on* , vol., no., pp.1,5, 27-29 June 2007

[5] Arduino datasheet. Available: *www.arduino.cc*

[6] Furuta Pendulum. Available: h*ttp://en.wikipedia.org/wiki/DC_motor*

[7] SIMULINK® user guide from MATLAB®

[8] SRV-02 Series Rotary Experiment #7 Rotary Inverted Pendulum, Revision 01, Quanser Co, pp. 3-7.

[9] Benjamin Seth Cazzolato and Zebb Prime, "On the Dynamics of the Furuta Pendulum," Journal of Control Science and Engineering., Hindawi Publishing Corporation, pp. 1-8, Feb 2011.

[10] J.Á.Acosta, "Furuta's Pendulum: A Conservative Nonlinear Model for Theory and Practice. Mathematical Problems in Engineering," Hindawi Publishing Corporation, pp. 1-29, 2010.

[11] A.V.Patel, K.Vashishtha," To perform SIL and PIL testing on fast dynamic system using economical AVR controller"International Journal for Scientific Research & Development| Vol. 1, Issue 2, pp.124-127 2013