İSTANBUL TECHNICAL UNIVERSITY ★ INSTITUTE OF SCIENCE AND TECHNOLOGY

HARDWARE IN THE LOOP SIMULATION OF UNDERACTUATED MECHANICAL SYSTEMS

M.Sc. Thesis by
Ömer ÇETİN, B.Sc.

Department : Electrical Engineering

Programme : Control & Automation Engineering

JUNE 2007

İSTANBUL TECHNICAL UNIVERSITY ★ INSTITUTE OF SCIENCE AND TECHNOLOGY

HARDWARE IN THE LOOP SIMULATION OF UNDERACTUATED MECHANICAL SYSTEMS

M.Sc. Thesis by
Ömer ÇETİN,

504041114

Date of submission: 7 May 2007

Date of defence examination: 13 June 2007

Supervisor (Chairman): Prof.Dr. Hakan TEMELTAŞ

Members of the Examining Committee Assoc. Prof.Dr. Metin GÖKAŞAN (İ.T.Ü.)

Assoc. Prof.Dr. Ata MUĞAN (İ.T.Ü.)

JUNE 2007

İSTANBUL TEKNİK ÜNİVERSİTESİ ★ FEN BİLİMLERİ ENSTİTÜSÜ

EKSİK UYARIMLI MEKANİK SİSTEMLERİN DONANIM ÇEVRİMLİ SİMÜLASYONU

YÜKSEK LİSANS TEZİ
Müh. Ömer ÇETİN
504041114

Tezin Enstitüye Verildiği Tarih : 7 Mayıs 2007
Tezin Savunulduğu Tarih : 13 Haziran 2007

Tez Danışmanı :	**Prof.Dr. Hakan TEMELTAŞ**
Diğer Jüri Üyeleri	**Doç.Dr. Metin GÖKAŞAN (İ.T.Ü.)**
	Doç.Dr. Ata MUĞAN (İ.T.Ü.)

HAZİRAN 2007

PREFACE

Primarily, I would like to thank my advisor Hakan TEMELTAŞ. I wish to give my special thanks to Murat YEŞİLOĞLU for his theoretical approaches and advices.

In addition, my most special thanks go to my wife Nuray for her unconditional love. Her emotional support and patience were the driving force of my efforts.

May, 2007 Ömer ÇETİN

CONTENTS

PREFACE **i**
CONTENTS **ii**
ABBREVIATIONS **iv**
LIST OF TABLES **v**
LIST OF FIGURES **vi**
LIST OF SYMBOLS **vii**
ÖZET **viii**
SUMMARY **ix**

1 INTRODUCTION **10**
1.1 Motivation 10
1.2 Real-Time Simulation 11
1.3 Inertia Wheel Pendulum (IWP) 12
1.4 Literature Review 12
1.5 Outline 13

2 SYSTEM MODELLING **14**
2.1 Analytical Model of the Simple Pendulum 14
2.2 Analytical Model of the Inertia Wheel Pendulum 15

3 HARDWARE IMPLEMENTATION **19**
3.1 M’Ax brushless AC servo motor drive 20
3.1.1 Programming the M’Ax drives 21
3.1.2 Advanced menus 21
3.1.3 M’Ax AC servo drive torque mode 22
3.1.4 Analog-input destination in torque mode 22
3.1.5 M’Ax AC servo drive signal connections 23
3.2 UNIDRIVE Brushless AC Motor Drive 26
3.2.1 Menu structure 26
3.2.2 Operating modes 28
3.2.3 Open loop mode 28
3.2.4 Open loop vector mode 28
3.2.5 Fixed V/F mode 28
3.2.6 Closed loop vector mode 29
3.2.7 Servo mode 29
3.2.8 Regen mode 29
3.2.9 Changing the operating mode procedure 29
3.2.10 Inertia measurement test 30
3.2.11 Rotating autotune 31
3.3 Hardware Management Box (HWMB) 31
3.4 dSPACE DS1104 Real-Time Processor Board 31

4 SOFTWARE IMPLEMENTATION **33**

4.1 dSPACE ControlDesk 33
4.2 M’AxSoft 34
4.3 CTSoft 35

5 HARDWARE-IN-THE-LOOP SIMULATOR ARCHITECTURE 37
5.1 System Overview 37
5.2 HIL Simulator Architecture 38

6 EXPERIMENTAL RESULTS 40
6.1 Simple Pendulum HIL Simulation Environment 40
6.1.1 M’Ax drive torque mode with analog-input destination 41
6.1.2 Simple pendulum simulation block 42
6.2 “Encoders” Block 43
6.3 “DS1104 ADC” Block 43
6.4 “DS1104 DAC” Block 44
6.5 “Digital Outputs” Block 45
6.6 Graphical User Interface Design in ControlDesk 45
6.7 Simple Pendulum HIL Simulation Experimantal Results 47

7 CONCLUSIONS 49
7.1 Hardware-in-the-Loop Simulation of the Simple Pendulum 49
7.2 Future Work 49

REFERENCES 50

ÖZGEÇMİŞ 53

ABBREVIATIONS

HIL	: Hardware-in-the-Loop
HILS	: Hardware-in-the-Loop Simulation
IWP	: Inertia Wheel Pendulum
HWMB	: Hardware Management Box
DOF	: Degrees-of-Freedom
RTI	: Real Time Interface
RTW	: Real Time Workshop
DAC	: Digital to Analog Converter
ADC	: Analog to Digital Converter
GUI	: Graphical User Interface

LIST OF TABLES

Page

Table 3.1: Important MA’x drive parameters…………………………………..….22

LIST OF FIGURES

Figure 1.1: A typical Hardware-in-the-Loop Simulator block scheme 11
Figure 1.2: Real-time system timing diagram... 11
Figure 1.3: Inertia Wheel Pendulum schematic... 12
Figure 2.1: Simple Pendulum 14
Figure 2.2: Inertia Wheel Pendulum.. 16
Figure 3.1: High Performance motion drives and actuators 19
Figure 3.2: Another view of High Performance motion drives and actuators 20
Figure 3.3: M’Ax Brushless AC Servo Motor Drive and connection diagram 20
Figure 3.4: M’Ax Drive control panel 21
Figure 3.5: M’Ax AC Servo Drive signal connections (top view) 23
Figure 3.6: Unidrive SP High performance motor drive (front view) 26
Figure 3.7: Unidrive SP menu structure... 27
Figure 3.8: Unidrive SP Menu 0 Cloning 27
Figure 3.9: Designed interface circuit Hardware Management Box (HWMB)........ 31
Figure 3.10: dSPACE DS1104 processor board 32
Figure 4.1: ControlDesk Screen.. 33
Figure 4.2: M’AxSoft program GUI 34
Figure 4.3: CTSoft program GUI... 36
Figure 5.1: General structure of the implemented HIL simulator............................ 37
Figure 5.2: HIL Simulator Signal Connections 38
Figure 6.1: Block scheme of the AC Servo motor as the simple pendulum HIL simulation.. 40
Figure 6.2: Block scheme of the AC Servo motor usage in large scale.................... 40
Figure 6.3: Simple Pendulum Model designed in MATLAB / Simulink 41
Figure 6.4: Simple Pendulum Model” block 42
Figure 6.5: Simulink Simple Pendulum Model “ENCODERS” Block 43
Figure 6.6: Simulink Simple Pendulum Model “DS1104 ADC” Block................... 44
Figure 6.7: Simulink Simple Pendulum Model “DS1104 DAC” Block................... 44
Figure 6.8: Simulink Simple Pendulum Model “Digital Outputs” Block................. 45
Figure 6.9: GUI designed in ControlDesk for the HIL visualization........................ 46
Figure 6.10: HIL simulation results of the Simple Pendulum (rotor position vs time)47
Figure 6.11 Computer simulation of the Simple Pendulum (theta vs time).............. 48

LIST OF SYMBOLS

m_1 : Mass of the pendulum
m_2 : Mass of the wheel
m : Combined mass of rotor and pendulum
I_1 : Moment of inertia of the pendulum about its center of mass
I_2 : Moment of inertia of the wheel about its center of mass
l_1 : Distance from pivot to the center of mass of the pendulum
l_2 : Distance from pivot to the center of mass of the rotor
l : Distance from pivot to the center of mass of pendulum and rotor
u : Control input torque
q_1 : Angle of the pendulum makes with the vertical
q_2 : Angle of the rotor
τ : Motor torque input applied on the disk
I : Moment of Inertia
K_1 : The kinetic energy of the pendulum
K_2 : The kinetic energy of the wheel
K : Total kinetic energy of the system
L : Lagrangian function
P : The potential energy of the system
g : The acceleration of gravity
$D(q)$: The inertia matrix

EKSİK UYARIMLI MEKANİK SİSTEMLERİN DONANIM ÇEVRİMLİ SİMÜLASYONU

ÖZET

Kontrol sistemleri tasarımında, mekanik sistemlerin benzetimi (simulation) ve modellenmesi önemli bir yer tutmaktadır. Hava ve Uzay araçları gibi kritik sistemler için kontrolör tasarımında asıl sisteme benzer ve gerçek-zamanlı (Real-Time) çalışabilen benzetim sistemleri; ileride çıkabilecek birçok arızanın, henüz tasarım ve test aşamasında çözülmesini sağlamaktadır.

Bu çalışmada, gerçek-zamanlı benzetim yapabilen ve klasik benzetim yöntemlerine göre asıl sistem davranışına daha yakın sonuçlar veren Donanım Çevrimli Benzetim (Hardware-in-the-loop Simulation) yöntemi kullanılarak, Eksik Uyarımlı (Underactuated) mekanik sistemlerin benzetimi üzerine çalışılmıştır. Birbirlerine rotorlarından akuple edilmiş yüksek performanslı motorlar ile gerçek bir mekanik sistemde birbirlerini karşılıklı etkileyen yük (load) ve eyleyici (actuator) çiftlerinin benzetimi yapılmıştır. Yük ve eyleyici çiftlerinden alınan veriler, üzerinde gerçek-zamanlı işlemci (real-time processor) bulunduran ve gerçek-zamanlı mekanik sistem modeli gömülmüş olan bir veri işleme kartı ile işlenerek eksik uyarımlı mekanik bir sistemin benzetimi üzerinde çalışılmıştır. Gerçeklenen sistem üzerinde ilk olarak, 1-Serbestlik derecesine sahip olan basit sarkaç modelinin benzetimi yapılmıştır. Platform üzerinde en fazla 2-Serbestlik derecesine sahip mekanik sistemler test edilebilmektedir.

Donanım Çevrimli Benzetim yöntemine dayalı bu platform sayesinde farklı mekanik sistemlerin, model üzerinde yapılacak değişiklikler ve bazı sistem parametrelerinin ayarlanmasıyla, gerçek-zamanlı benzetimi yapılabilir. Bu sistem, ileride yapılabilecek birçok araştırma için bir test platformu olma özelliğini taşımaktadır.

HARDWARE IN THE LOOP SIMULATION OF UNDERACTUATED MECHANICAL SYSTEMS

SUMMARY

Simulation and modeling of the Mechanical Systems plays a very important role in control systems design. In controller design for the critical systems like aircrafts and space vehicles, real-time simulation systems provides to designers to seize the failures that will be possible for the designed system, during the test and design process.

In this study, underactuated mechanical systems simulation is processed using Hardware-in-the-loop Simulation technique for its real-time simulation and more precise results superiorities with respect to conventional simulation techniques. Actuators and loads, affected each other in a mechanical system, are simulated as coupled high performance motors. Data from actuator and load couples are processed with a real-time processor board, which a real-time mechanical system model embedded in. A simple pendulum model is simulated in the implemented system as a 1-degree-of-freedom system. Mechanical systems, which have 2-degrees-of-freedom, could be tested on the platform.

Different kinds of mechanical systems could be simulated on this platform by means of Hardware-in-the-Loop simulation technique, only requires some parameter tuning process. This project will be a very important test platform for the future researches and controller designs.

1 INTRODUCTION

Computer simulations play an important role on mechatronic system design process. The convergence of the real system's response and the simulation results is one of the most important criteria for the simulation performance. Computer simulations based on the real system's mathematical expressions and models. However, in real world, systems are too complex and nonlinear to write an exact mathematical expression of the system because of the nonlinearities and complexities. In order to overcome these problems, Hardware-in-the-Loop (HIL) simulation technique is proposed by the researchers. In the HIL Simulation technique, some part of the mechatronic system incorporated into the simulation loop. HIL simulation provides an effective platform by adding the complexity of the plant under control to the test platform.

In this dissertation, Inertia Wheel Pendulum (IWP) as an underactuated mechanical system is investigated. Inertia Wheel Pendulum was first introduced by M.W.Spong. It is a physical pendulum with a symmetric disk attached to the end. A motor actuates the disk and the coupling torque generated by the angular acceleration of the disk can be used to control the system. Inertia Wheel Pendulum system has 2-degrees-of-freedom and one actuator as the motor coupled disk.

1.1 Motivation

In this study, a real-time Hardware-in-the-Loop simulator is aimed to be implemented and tested for proper operations. This simulator is used to simulate the simple mechanical system with real torques. In Figure 1.1, a typical Hardware-in-the-Loop Simulator block scheme is proposed.

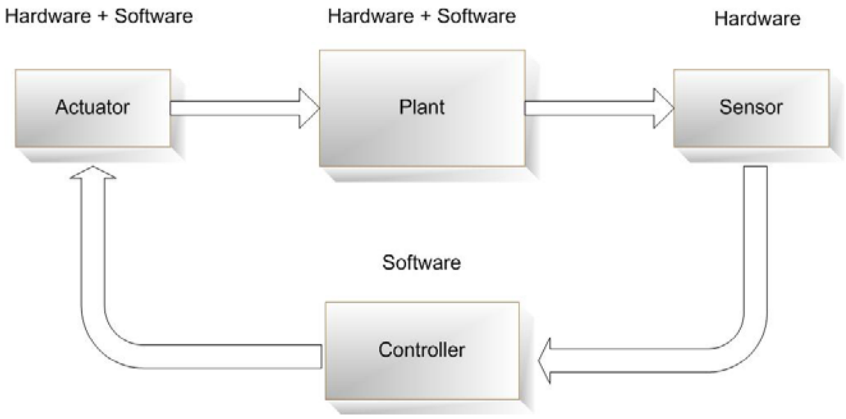


Figure 1.1: A typical Hardware-in-the-Loop Simulator block scheme

In the software part of the simulator, MATLAB / Simulink program used for real-time modeling and ControlDesk program for the real-time visualizations. MAXSoft and CTSoft programs are special programs of the high performance motor drives for parameter tuning.

1.2 Real-Time Simulation

Physical systems have certain dynamics associated with it. In controller design, engineers have to design a system based on those dynamics. This could be interpreted as the physical system time constant. This will derive a step size or sample time for the controller. The numerical calculations and the control algorithms of the controller have to be executed within that sample time. In Figure 1.2, Real-time system timing diagram shows the program execution times.

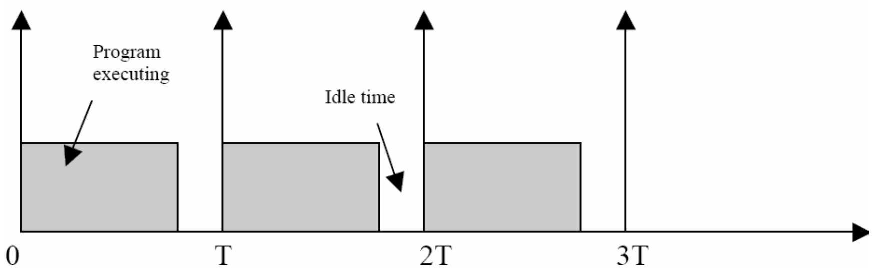


Figure 1.2: Real-time system timing diagram

If the sample time of the program is T, the program is executed at distinct points in time that are one sample time apart. Each step of the program finishes executing before the next step is due to start; thus, this program is running in real-time.

1.3 Inertia Wheel Pendulum (IWP)

The Inertia Wheel Pendulum (IWP) is a planar inverted pendulum with a revolving wheel at the end, as shown in Figure 1.3. The wheel is actuated and the joint of the pendulum at the base is unactuated.

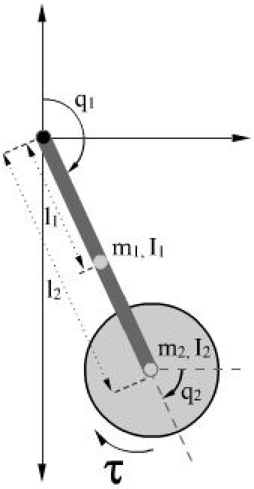


Figure 1.3: Inertia Wheel Pendulum schematic

The Inertia Wheel Pendulum was first introduced by Spong in [24] where a supervisory hybrid-switching control strategy is applied to asymptotic stabilization of the Inertia Wheel Pendulum around its upright equilibrium point. First, a passivity-based controller swings up the pendulum. Then, a balancing controller that is obtained by Jacobian linearization or (local) exact feedback linearization stabilizes the pendulum around its upright position.

1.4 Literature Review

From the 1970s to nowadays, Hardware-in-the-Loop simulation technique is became a widely used simulation method. In 1970, Pastrick and his colleagues used this method for the missile controller simulation [1]. This method is used not only in control systems' simulation but also used in biomedical electronics [2], power power electronics [3] and other electronics branches [4,5,6,7].

In [1], Pastrick and his colleagues in the design process of a controller for a laser guided missile; they include, some mechanical part of the missile into the simulation. Le and friends used HIL method as a fast prototyping method for the mechatronics system design in [5]. Abou-Samah and colleagues proposed a fast test platform for the nonholonomic mobile robot design in [6].

Lionel and Clément designed an underactuated robot manipulator in [7]. This manipulator is superior to other manipulators in the view environment adaptation and flexibility.

Grega, suggested a low cost hardware-in-the-loop simulator system for the control theory laboratories in [8].

Turan and colleagues designed a HIL simulation system for a PUMA type robot manipulator in [9]. In the same working group; Singaraju and colleagues proposed an HIL simulator of the PUMA type robot via Internet in [10].

In [11]; Temeltaş and colleagues designed a 2-DOF robotic manipulator simulation using direct drive motor couples.

Inertia Wheel Pendulum first revealed by Spong in [12]. Spong and colleagues introduced a nonlinear control algorithm for the Inertia Wheel Pendulum. In [13] , Fantoni and friends proposed a controller for swinging the linkage and rise the pendubot to its uppermost unstable equilibrium position. Ortega and colleagues proposed the passivity-based control (PBC) as interconnection and damping (IDA) assignment to the problem of stabilization of underactuated mechanical systems in [14].

1.5 Outline

In the Chapter 1 a overview of this dissertation is offered. The analytical expressions of the systems are briefly introduced in the Chapter 2. Chapter 3 covered the hardware considerations. Chapter 4 is about the software environment. In Chapter 5 the architecture of the implemented Hardware-in-the-Loop simulator is presented. Chapter 6 contains the HIL simulation of the mechnical systems' results. In Chapter 7 results are interpreted and future works introduced.

2 SYSTEM MODELLING

2.1 Analytical Model of the Simple Pendulum

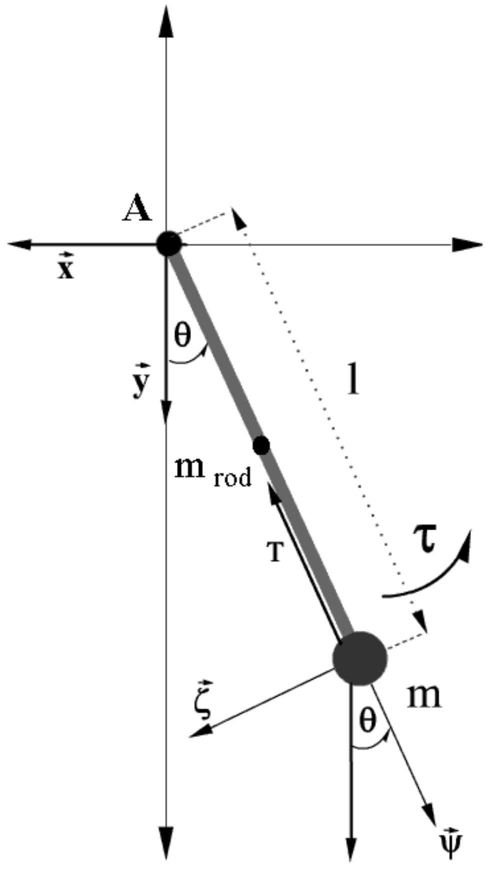


Figure 2.1: Simple Pendulum

In Figure 2.1 the simple pendulum schematic pointed out. l denotes the length of the massless rod and m denotes the mass of the bob. θ is the angle between the rod and the vertical axis. k is the friction constant proportional with the velocity. τ is the torque input applied on the point mass. I donates the moment of inertia. The equation of the torques acting on the mass figured out as below:

$$\sum \tau = 0 \tag{2.1}$$

$$\tau = I\ddot{\theta} \tag{2.2}$$

The point mass' moment of inertia about point A is:

$$I_{mass} = ml^2 \tag{2.3}$$

Then;

$$ml^2\ddot{\theta} = -mgl\sin\theta - kl^2\dot{\theta} + \tau \tag{2.4}$$

$$\ddot{\theta} = -\frac{g}{l}\sin\theta - \frac{k}{m}\dot{\theta} + \frac{1}{ml^2}\tau \tag{2.5}$$

If the mass of the rod is not neglected, the moment of inertia of the whole system changes. The point mass and the rod rotate on the same axis so the total moment of inertia could be extracted out from the parallel axis theorem. m_{rod} denotes the mass of the rod. Thus; the total moment of inertia about point A is:

$$I_{Total} = I_{mass} + I_{rod} \tag{2.6}$$

$$I_{Total} = ml^2 + \frac{1}{3}m_{rod}l^2 \tag{2.7}$$

$$I_{Total} = \frac{1}{3}\left(3m + m_{rod}\right)l^2 \tag{2.8}$$

Then;

$$\frac{1}{3}\left(3m + m_{rod}\right)l^2\ddot{\theta} = -\left(mg\sin\theta\right)l - \left(m_{rod}g\sin\theta\right)\frac{l}{2} - kl^2\dot{\theta} + \tau \tag{2.9}$$

$$\ddot{\theta} = -\frac{3}{2}\frac{\left(2m + m_{rod}\right)}{\left(3m + m_{rod}\right)}\frac{g\sin\theta}{l} - \frac{3k}{\left(3m + m_{rod}\right)}\dot{\theta} + \frac{3\tau}{\left(3m + m_{rod}\right)l^2} \tag{2.10}$$

2.2 Analytical Model of the Inertia Wheel Pendulum

The reaction wheel pendulum is a two-degree-of-freedom robot as shown in Figure 2.2 [25]. The pendulum forms the first link and the rotating wheel is the second one. The angle of the pendulum with the vertical axis is $q_{1,}$ and the angle of the rotating wheel is the $q_{2.}$

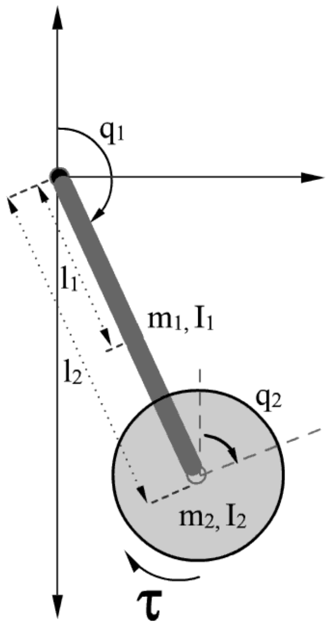


Figure 2.2: Inertia Wheel Pendulum

The Parameters of the system are described below:

m_1	mass of the pendulum
m_2	mass of the wheel
I_1	moment of inertia of the pendulum about its center of mass
I_2	moment of inertia of the wheel about its center of mass
l_1	distance from pivot to the center of mass of the pendulum
l_2	distance from pivot to the center of mass of the rotor
q_1	angle of the pendulum makes with the vertical
q_2	angle of the rotor
τ	motor torque input applied on the disk
K_1	kinetic energy of the pendulum
K_2	kinetic energy of the wheel
P	potential energy of the system
K	total kinetic energy
g	the acceleration of gravity

The kinetic energy of the pendulum is:

$$K_1 = \frac{1}{2}\left(m_1 l_1^2 + I_1\right)\dot{q}_1^2 \qquad \textbf{(2.11)}$$

The kinetic energy of the wheel is:

$$K_2 = \frac{1}{2}\left(m_2 l_2^2\right)\dot{q}_1^2 + \frac{1}{2} I_2 \left(\dot{q}_1 + \dot{q}_2\right)^2 \qquad \textbf{(2.12)}$$

The total kinetic energy is:

$$K = K_1 + K_2 = \frac{1}{2}\left(m_1 l_1^2 + m_2 l_2^2 + I_1 + I_2\right)\dot{q}_1^2 + I_2 \dot{q}_1 \dot{q}_2 + \frac{1}{2} I_2 \dot{q}_2^2 \qquad \textbf{(2.13)}$$

The potential of the system is given by:

$$P = \left(m_1 l_1 + m_2 l_2\right)\left(\cos\left(q_1\right) - 1\right) g \qquad \textbf{(2.14)}$$

The lagrangian function is given by:

$$L = K - P \qquad \textbf{(2.15)}$$

$$L = \frac{1}{2}\left(m_1 l_1^2 + m_2 l_2^2 + I_1 + I_2\right)\dot{q}_1^2 + I_2 \dot{q}_1 \dot{q}_2 + \frac{1}{2} I_2 \dot{q}_2^2 - \left(m_1 l_1 + m_2 l_2\right)\left(\cos\left(q_1\right) - 1\right) g \qquad \textbf{(2.16)}$$

Using Euler-Lagrange's equations:

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{q}}\left(q, \dot{q}\right)\right) - \frac{\partial L}{\partial q}\left(q, \dot{q}\right) = \tau \qquad \textbf{(2.17)}$$

From these equations:

$$\frac{\partial L}{\partial \dot{q}_1} = \left(m_1 l_1^2 + m_2 l_2^2 + I_1 + I_2\right)\dot{q}_1 + I_2 \dot{q}_2 \qquad \textbf{(2.18)}$$

$$\frac{\partial L}{\partial q_1} = \left(m_1 l_1 + m_2 l_2\right)\sin\left(q_1\right) g \qquad \textbf{(2.19)}$$

$$\frac{\partial L}{\partial \dot{q}_2} = I_1\dot{q}_1 + I_2\dot{q}_2 \tag{2.20}$$

$$\frac{\partial L}{\partial q_2} = 0 \tag{2.21}$$

The dynamic equations of the system is therefore given by:

$$\left(m_1 l_1^2 + m_2 l_2^2 + I_1 + I_2\right)\ddot{q}_1 + I_2\ddot{q}_2 - \left(m_1 l_1 + m_2 l_2\right)\sin\left(q_1\right) = 0 \tag{2.22}$$

$$I_2\ddot{q}_1 + I_2\ddot{q}_2 = \tau \tag{2.23}$$

In compact for the equations can be rewritten as below:

$$D(q)\ddot{q} + \rho(q) = u \tag{2.24}$$

where $q = \begin{bmatrix} q_1 \\ q_2 \end{bmatrix}$ is the vector of generalized coordinates, $u = \begin{bmatrix} 0 \\ \tau \end{bmatrix}$ is the vector of joint torques, $D(q)$ is the inertia matrix and is given by:

$$D(q) = \begin{bmatrix} m_1 l_1^2 + m_2 l_2^2 + I_1 + I_2 & I_2 \\ I_2 & I_2 \end{bmatrix} = \begin{bmatrix} d_{11} & d_{12} \\ d_{21} & d_{22} \end{bmatrix} \tag{2.25}$$

$$\rho(q) = \begin{bmatrix} -\left(m_1 l_1 + m_2 l_2\right)\sin\left(q_1\right) g \\ 0 \end{bmatrix} \tag{2.26}$$

3 HARDWARE IMPLEMENTATION

In Figure 3.1 and Figure 3.2, High Performance motion drives and actuators stated.

Figure 3.1: High performance motion drives and actuators

The overall system is based on these actuator-load couples. There exists three actuator-load motor couples. Each of the motors has high performance AC Servo capabilities. There is two kinds of drives, one of them is Control Techniques' product of high performance AC servo drive M'Ax and the other is from the same manufacturer Unidrive SP.

Figure 3.2: Another view of high performance motion drives and actuators

3.1 M’Ax Brushless AC Servo Motor Drive

M’Ax is the high performance AC Servo drive. In Figure 3.3, drive connection diagram is presented.

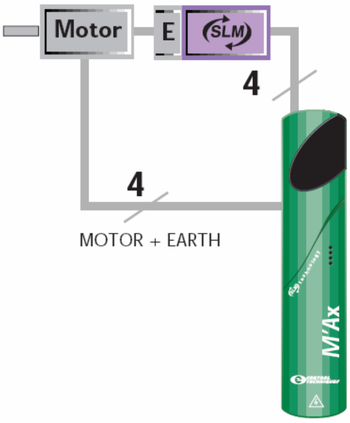


Figure 3.3: M’Ax Brushless AC servo motor drive and connection diagram

3.1.1 Programming The M’Ax Drives

Menu 0 contains a group of parameters that apply to simple applications; these parameters are duplicates of certain advanced parameters. Menu parameters could adjusted both front panel and from a PC over RS 232 or RS 485 serial connection. In Figure 3.4 the front panel of the M’Ax drive figured out. PC connection requires a program, MaxSoft.

3.1.2 Advanced Menus

The advanced menus are numbered 1 to 13; and contain all the (advanced) parameters, which are grouped according to function. After adjustment, new parameter-values can be saved for future use. The drive can be restored to its default state (all parameters returned to their default values). This is normally performed during the initial settingup of the drive, and can be performed again at any appropriate time.

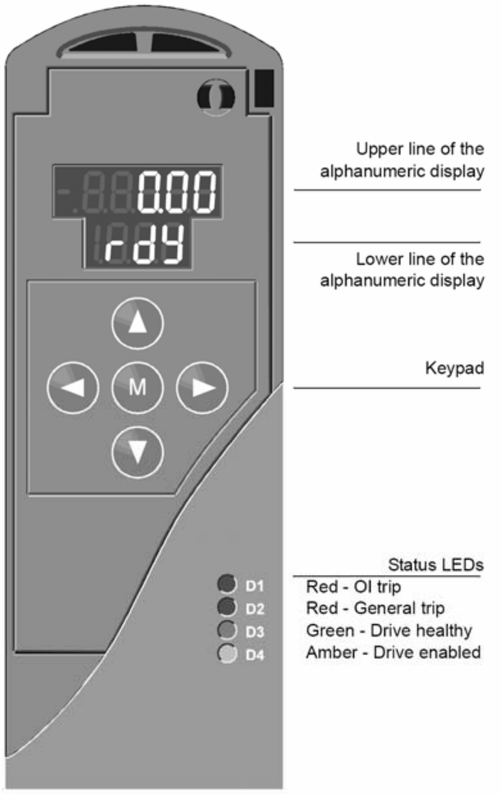


Figure 3.4: M’Ax drive control panel

The drive is configured and controlled by the proper adjustments of the parameters.

Table 3.1: Important MA'x drive parameters

	Operation	Par.	Alternative	Def.	Value	Range	Definition
1	Operating Mode					open loop	
2						closed loop vector	
3						closed loop servo	
4						regen	
5	Maximum Speed	0.08				0-3000	
6	Acceleration					0.2-unlimited	time given in sec to reach 1000 rpm
7	Decelaration					0.2-unlimited	time given in sec to reach 0 from 1000 rpm
8	Accessing Mode	0.00					menu 0 a erişim
9	Analog input mode						
10	Analog input	0.30	1.14	3	1		Must be ``1`` for Analog Input
11	Speed	0.05					
12	Save Command	0.50					
13	SLM on-line enable	0.17					
14	Torque mode selector	0.22				0 ~ 1	
15	Saving the values to the EEPROM	0.00			1000		
16	Load inertia	0.10					
17	Inertia units selected	0.11				0 ~ 1	0 kgm2 / 1 kgcm2
18	Torque reference	0.23					

3.1.3 M'Ax AC Servo Drive Torque Mode

An open loop current mode for a servo drive, giving a motor torque output proportional to an input demand. Current will be controlled by the drive, but torque value will be dependent upon the K_T value for the motor which may vary with temperature.

3.1.4 Analog-Input Destination In Torque Mode

When the drive is in torque mode (parameter 4.11 set to 1 or 2), and controlled by an analog input parameter 7.10 Analog input destination selector must be set to 4.08, parameter 1.14 Reference selector must be set to 3 to 5.

Summary for torque mode with analog reference

1. Set 1.14 to 3

2. Set 4.11 to 1 or 2

3. Set 7.10 to 4.08

4. Set XX.00 to 1000

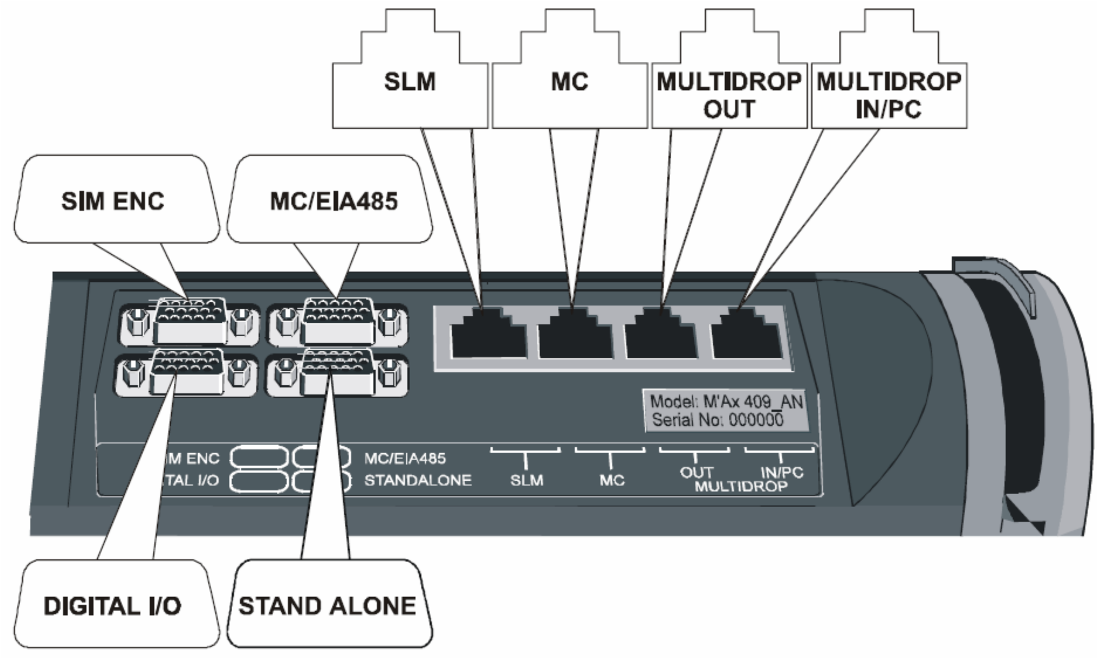


Figure 3.5: M’Ax AC servo drive signal connections (top view)

3.1.5 M’Ax AC Servo Drive Signal Connections

SIM ENC

(15-way high-density female D-type)

- Simulated-encoder quadrature AB plus Z marker-pulse outputs for supplying encoder speed and position to a system controller or PLC or another servo amplifier.
- Two analog outputs

- Standard-precision analog speed or torque reference input

MC/EIA485 (15-way high-density female D-type)

- I/O to a motion controller

- Hardware enable input (electrical enable signal for the drive)

- Status-relay contact

- Alternative use as an EIA485 port for control from a system controller, PLC or PC

- SLM-and-user back-up supply input for retaining position information when the drive is powered-down

- 24V user supply output generated in the drive

- Status-relay contact

DIGITAL I/O (15-way high-density male D-type)

- Eight digital inputs for electrical contacts for local or remote (system controller or plc) control of the drive

- Four digital outputs for local or remote monitoring and/or simple control of other equipment

- 24V user supply output generated in the drive

STANDALONE (15-way high-density male D-type)

- Frequency-and-direction, quadrature square-wave inputs and directional pulse inputs

- High-precision analog speed or torque reference input

- Touch trigger input

- SLM-and-user back-up supply input for retaining position information when the drive is powered-down

- 24V user supply output generated in the drive

- Single digital output

- Hardware enable input (electrical enable signal for the drive)

RJ45 connectors

SLM

- I/O to the SLM
- 24V DC supply to the SLM
- Hardware enable input (electrical enable signal for the drive)
- Drive-status supply

MC

- I/O to a motion controller
- Hardware enable input (electrical enable signal for the drive)
- Drive-status supply
- 24V user supply output generated in the drive

MULTIDROP OUT

- I/O to another drive operating in a master/slave system
- Hardware enable input (electrical enable signal for the drive)
- Drive-status output
- +24V loop output

MULTIDROP IN/PC

- I/O to another drive operating in a master/slave system
- Hardware enable input (electrical enable signal for the drive)
- EIA232 communications to a PC running a dedicated application program (for setting-up purposes only)
- Drive-status input
- +24V loop input.

3.2 UNIDRIVE Brushless AC Motor Drive

3.2.1 Menu Structure

The drive parameter structure consists of menus and parameters. The drive initially powers up so that only menu 0 can be viewed. The up and down arrow buttons are used to navigate between parameters and once level 2 access (L2) has been enabled the left and right buttons are used to navigate between menus. The menus and parameters roll over in both directions. In Figure 3.7 Unidrive SP menu structure figured out.

When changing between menus the drive remembers, which parameter was last viewed in a particular menu and thus displays that parameter. In Figure 3.6 Unidrive SP High performance motor drive is presented.

Figure 3.6: Unidrive SP high performance motor drive (front view)

Menu 0 is used to bring together various commonly used parameters for basic easy set up of the drive. Appropriate parameters are cloned from the advanced menus into menu 0 and thus exist in both locations.

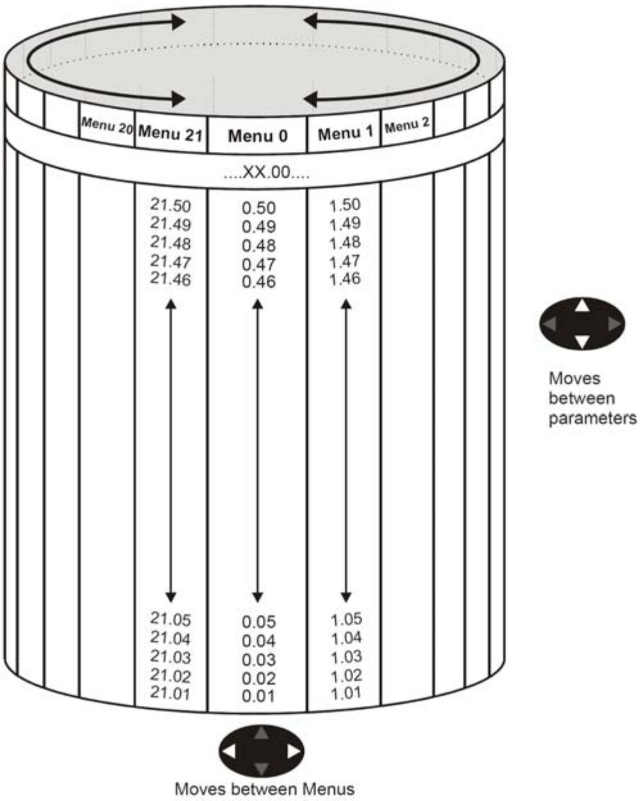


Figure 3.7: Unidrive SP menu structure

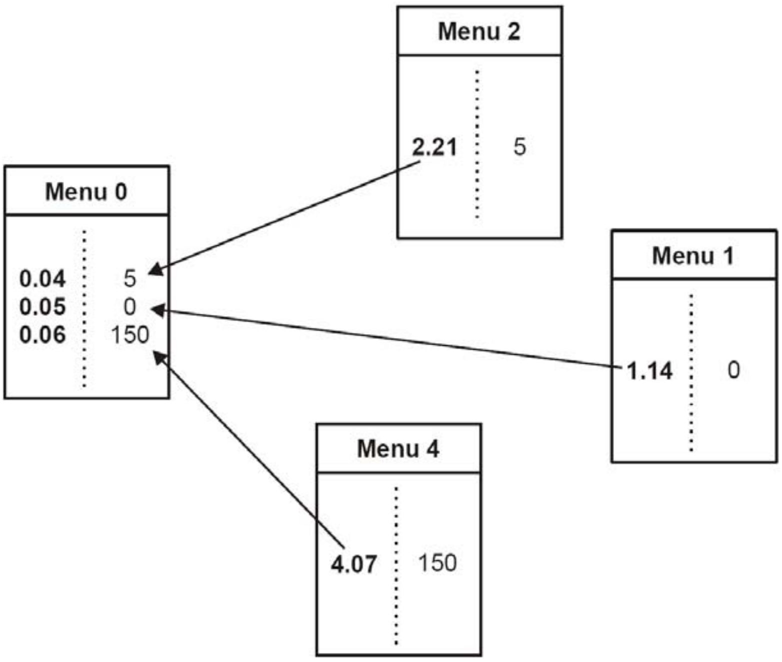


Figure 3.8: Unidrive SP Menu0 cloning

The advanced menus consist of groups or parameters appropriate to a specific function or feature of the drive. Menus 0 to 22 can be viewed on both keypads.

3.2.2 Operating Modes

The Unidrive SP is designed to operate in any of the following modes:

1. Open loop mode

    Open loop vector

    Fixed V/F mode (V/Hz)

    Quadratic V/F mode (V/Hz)

2. Closed loop vector

3. Servo

4. Regen

3.2.3 Open Loop Mode

For use with standard AC induction motors. The drive applies power to the motor at frequencies varied by the user. The motor speed is a result of the output frequency of the drive and slip due to the mechanical load. The drive can improve the speed control of the motor by applying slip compensation. The performance at low speed depends on whether V/F mode or open loop vector mode is selected.

3.2.4 Open Loop Vector Mode

The voltage applied to the motor is directly proportional to the frequency except at low speed where the drive uses motor parameters to apply the correct voltage to keep the flux constant under varying load conditions. Typically 100% torque is available down to 1Hz for a 50Hz motor.

3.2.5 Fixed V/F Mode

The voltage applied to the motor is directly proportional to the frequency except at low speed where a voltage boost is provided which is set by the user. This mode can be used for multi-motor applications. Typically 100% torque is available down to

4Hz for a 50Hz motor. Quadratic V/F mode the voltage applied to the motor is directly proportional to the square of the frequency except at low speed where a voltage boost is provided which is set by the user. This mode can be used for running fan or pump applications with quadratic load characteristics or for multi-motor applications. This mode is not suitable for applications requiring a high starting torque.

3.2.6 Closed Loop Vector Mode

For use with induction motors with a feedback device fitted. The drive directly controls the speed of the motor using the feedback device to ensure the rotor speed is exactly as demanded. Motor flux is accurately controlled at all times to provide full torque all the way down to zero speed.

3.2.7 Servo Mode

For use with permanent magnet brushless motors with a feedback device fitted. The drive directly controls the speed of the motor using the feedback device to ensure the rotor speed is exactly as demanded. Flux control is not required because the motor is self excited by the permanent magnets which form part of the rotor. Absolute position information is required from the feedback device to ensure the output voltage is accurately matched to the back EMF of the motor. Full torque is available all the way down to zero speed.

3.2.8 Regen Mode

For use as a regenerative front end for four quadrant operation. Regen operation allows bi-directional power flow to and from the AC supply. This provides far greater efficiency levels in applications which would otherwise dissipate large amounts of energy in the form of heat in a braking resistor. The harmonic content of the input current is negligible due to the sinusoidal nature of the waveform when compared to a conventional bridge rectifier or thyristor front end.

3.2.9 Changing The Operating Mode Procedure

Use the following procedure only if a different operating mode is required:

1. Ensure the drive is not enabled, i.e. terminal 31 is open or **Pr 6.15** is Off (0)

2. Enter either of the following values in **Pr 0.00**, as appropriate:

1253 (Europe, 50Hz AC supply frequency)

1254 (USA, 60Hz AC supply frequency)

3. Change the setting of Pr 0.48 as follows:

0.48 setting	Operating mode
1	Open-loop
2	Closed-loop Vector
3	Closed-loop Servo
4	Regen

3.2.10 Inertia Measurement Test

This test should be performed with the motor coupled to the load. It can only be used with constant inertia loads that are free to rotate with no electrical or mechanical travel limits.

1. Set parameter **Pr 5.12** to **3.**

2. Enable the drive.

3. Press the green start key to perform the autotune. The drive will attempt to accelerate the motor and load up to ¾ rated speed and back to zero. Multiple attempts may be made starting with current limit set to produce 1/16 of rated torque for the first attempt and stepping up to a maximum of rated torque if required. If the test is unsuccessful a **tune1 trip** will occur. Unless the application is intended to use very long acceleration times a tunel trip suggests that the motor may be too small for the application. On successful completion of the test the motor and load inertia is calculated and automatically stored in parameter **#3.18**.

3.2.11 Rotating Autotune

1. Set parameter **#5.12** to **2**.

2. Press the green start key to perform the rotating autotune. The motor will accelerate at the rate defined in parameter **#2.11** until it reaches 2/3 of rated speed. This speed will be held for up to 36 seconds. This test calculates and sets new values for stator inductance (**#5.25**) and motor saturation breakpoints (**#5.29** and **#5.30**).

3. These parameters are automatically saved to the drive EEPROM.

3.3 Hardware Management Box (HWMB)

Hardware Management Box (HWMB) is designed for securely enable and disable UNIMOTOR SP and M'Ax drives. Optocouplers are used in this circuit for DS1104 board Digital I/O's interfacing to the drives' digital inputs. "Reset" buttons and "Secure Hardware Enable" switches performs the drive management.

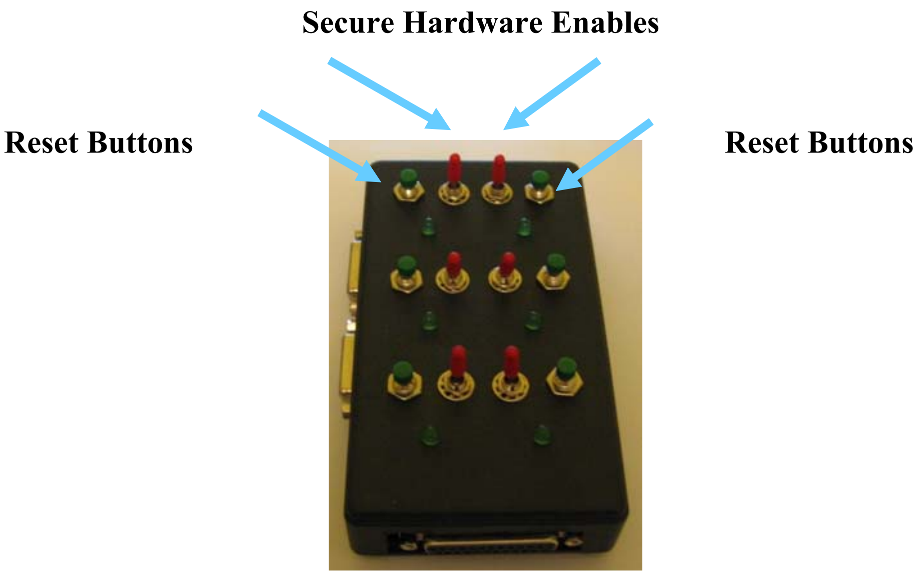


Figure 3.9: Designed interface circuit Hardware Management Box (HWMB)

3.4 dSPACE DS1104 Real-Time Processor Board

The DS1104 board is considered as a platform on which a simulation is run.

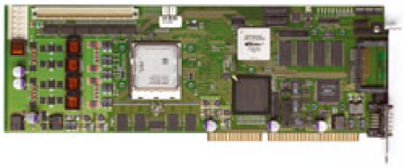


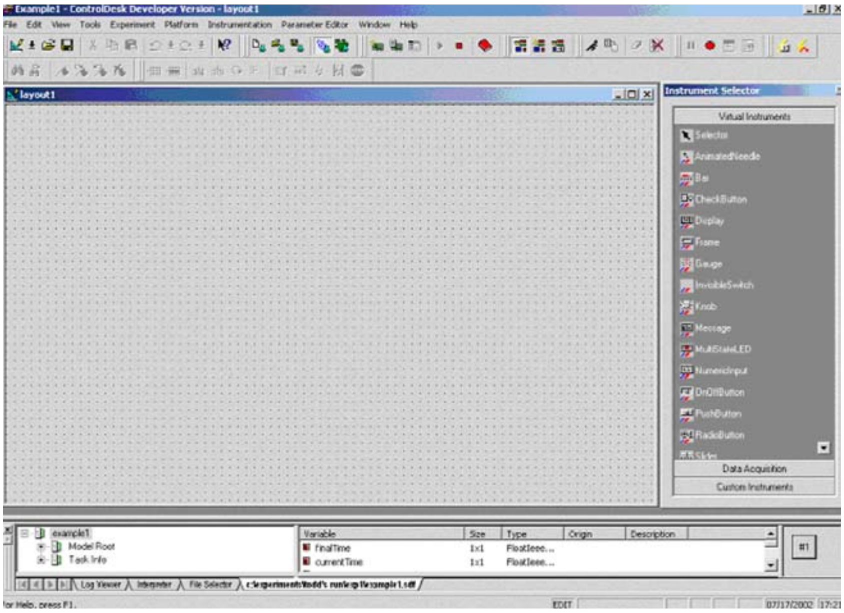
Figure 3.10: dSPACE DS1104 processor board

Real-Time Interface provides Simulink blocks for graphical configuration of A/D, D/A, digital I/O lines, incremental encoder interface and PWM generation.

4 SOFTWARE IMPLEMENTATION

4.1 dSPACE ControlDesk

ControlDesk is experiment software for seamless controller development. It performs all the necessary tasks, and gives you a single working environment, from the start of experimentation right through to the end. ControlDesk Standard can be operated in two modes: The Developer mode gives you the full functionality, and the Operator mode protects your experiments against unauthorized changes.



Variable	Size	Type	Origin	Description
finalTime	1x1	FloatIeee...		
currentTime	1x1	FloatIeee...		

Figure 4.1: ControlDesk program interface

Configuring Instrument Panels ControlDesk offers a variety of virtual instruments for building and configuring virtual instrument panels according to your needs. You can use drag & drop to combine any set of instruments in a virtual instrument panel that is specific to an application. Instrument operation can also

The dSPACE tools consists of both hardware and software. A combined processor and I/O board is mounted on an PCI slot in each of the PC's. The board also has various I/O functions, e.g., ADC, DAC, digital I/O and incremental encoder interfaces. A C-code generator in Simulink, (RTW) generates the C-code of a Simulink file, a software module RTI (Real Time Interface) and a compiler that compiles and download the code.

The File Selector will only display certain file types. It will show *.mdl (Simulink model files), *.ppc (Compiled object files for execution on the DS1104), and *.sdf (System Description File) files. The *.sdf file contains references to the executable file (either *.mdl or *.ppc), a Variable Description file (*.trc) and the platform the simulation is built for (Simulink, DS1104 or other dSPACE hardware).

4.2 M'AxSoft

M'AxSoft is a Windows based drive set-up program that is designed to enable the complete control and display of all parameters within a M'Ax drive. M'AxSoft provides the user with a graphical interface that is logically splitted into a series of screens, offering the quick and easy viewing and where appropriate editing of a parameter value. Individual detailed parameter information can be displayed at any time defining the parameters function, type and min/max permitted value.

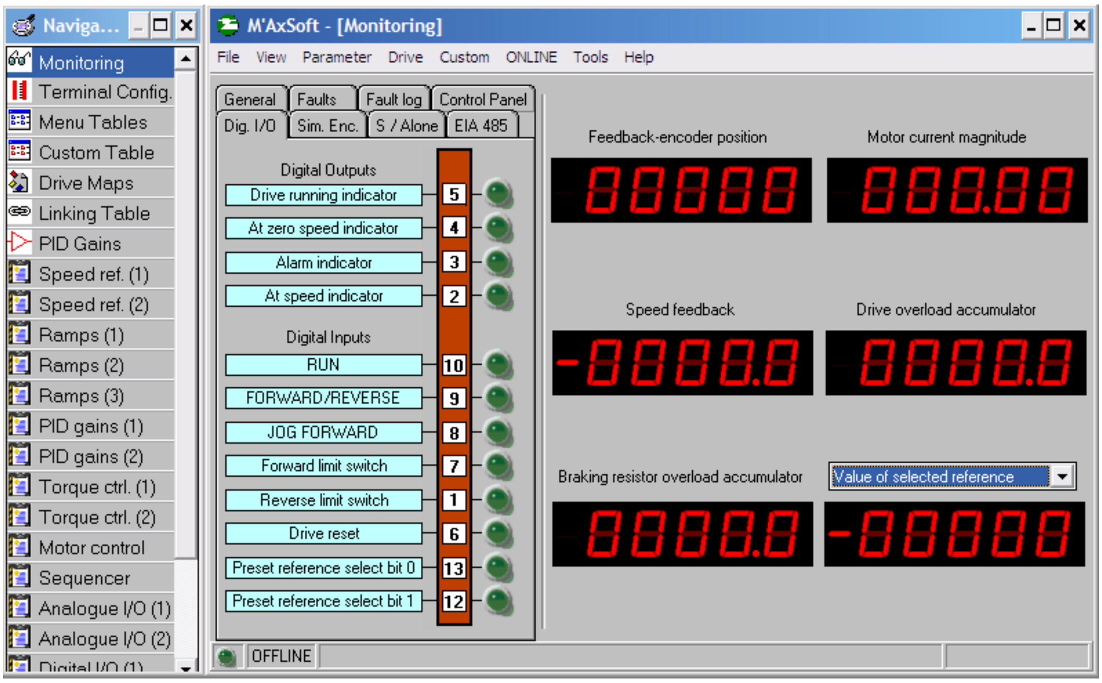


Figure 4.2: M'AxSoft program GUI

M’AxSoft provides the capability to save all the parameter values for a drive into a file on the PC's hard disk. This is useful for keeping a back up of the drive parameters or cloning the parameters to another drive.

4.2.1 Drive Setup Wizard

A drive setup wizard reads the motor data from the module. Dynamic braking can be selected and wiring is shown for internal or external braking. Speed input reference options are shown pictorially, enabling the user to visualise their selection. Drive feedback is shown pictorially and allows analogue outputs to be set along with Simulated Encoder output. Load inertia and drive stiffness can be entered and sent to the drive which calculates PID values for the gain selector.

4.3 CTSoft

CTSoft is a Windows based drive-commissioning program that allows the complete control and display of all parameters within Control Techniques' Unidrive SP drives.

CTSoft provides the user with a graphical interface that is logically splitted into a series of screens offering quick and easy viewing and, where appropriate, the ability to edit parameter values. Individual detailed parameter information can be displayed at any time showing the parameter function, type and range of permitted values.

The drive's parameter set is split up into a series of related groups referred to as “menus”. Many of these menus have an associated graphical block diagram which may be displayed and used interactively within CTSoft.

As well as listings and diagram views, a variety of other functions are provided to enable the maintenance of a drive's parameter set; these include configuring analogue and digital I/O references, uploading and downloading all drive parameters, saving and reading parameter sets from disk and comparing the drive's parameters to a disk file or default values.

CTSoft can function both as an offline tool and as an online tool. Being *online* means that the PC is connected to a drive via a communications lead and CTSoft is able to read and write parameter values directly.

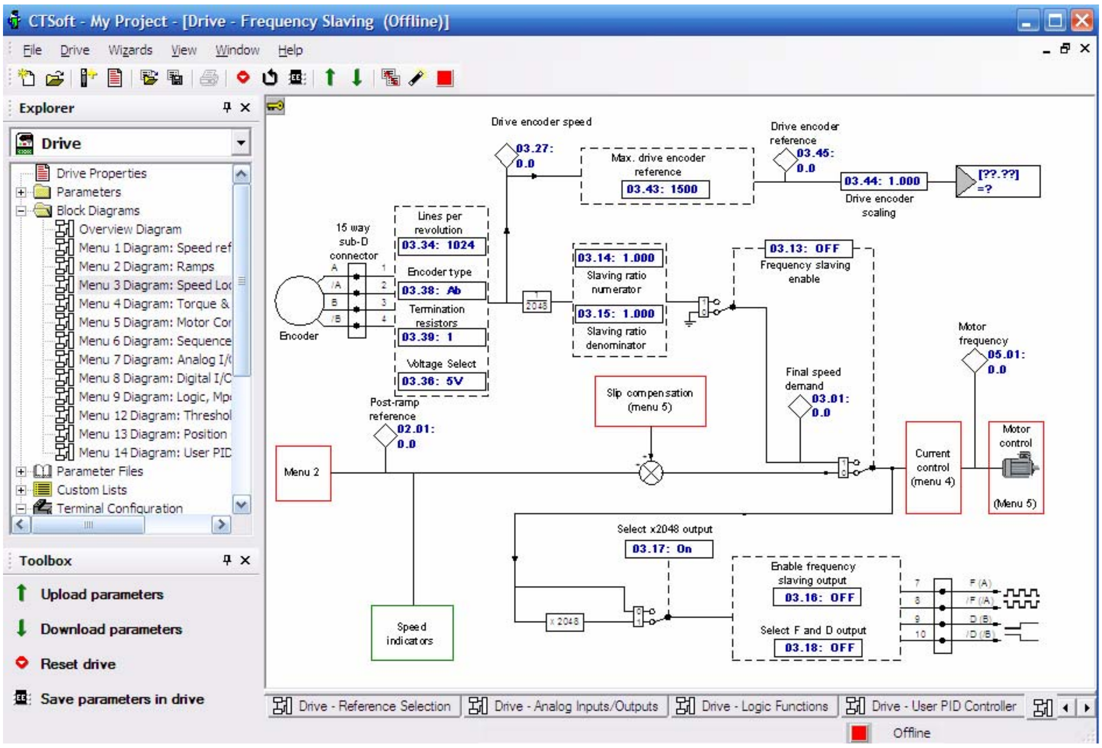


Figure 4.3: CTSoft program GUI

CTSoft provides the capability to save all the parameter values for a drive into a file on the PC's hard disk. This is useful for keeping a back-up of the drive parameters or cloning the parameters to another drive.

CTSoft provides functions to upload and download entire drive parameter sets. This may be done either for back-up purposes or for transferring the same configuration from one drive to another.

The value of a parameter can be edited offline and online via the edit parameter dialog. This is usually invoked by double-clicking on the parameter in a listing or diagram view. If the current drive is offline then editing a parameter's value will update CTSoft's parameter "memory" directly. If the drive is online then the value will be written to the drive over the communications link and the memory value will be updated.

5 HARDWARE-IN-THE-LOOP SIMULATOR ARCHITECTURE

5.1 System Overview

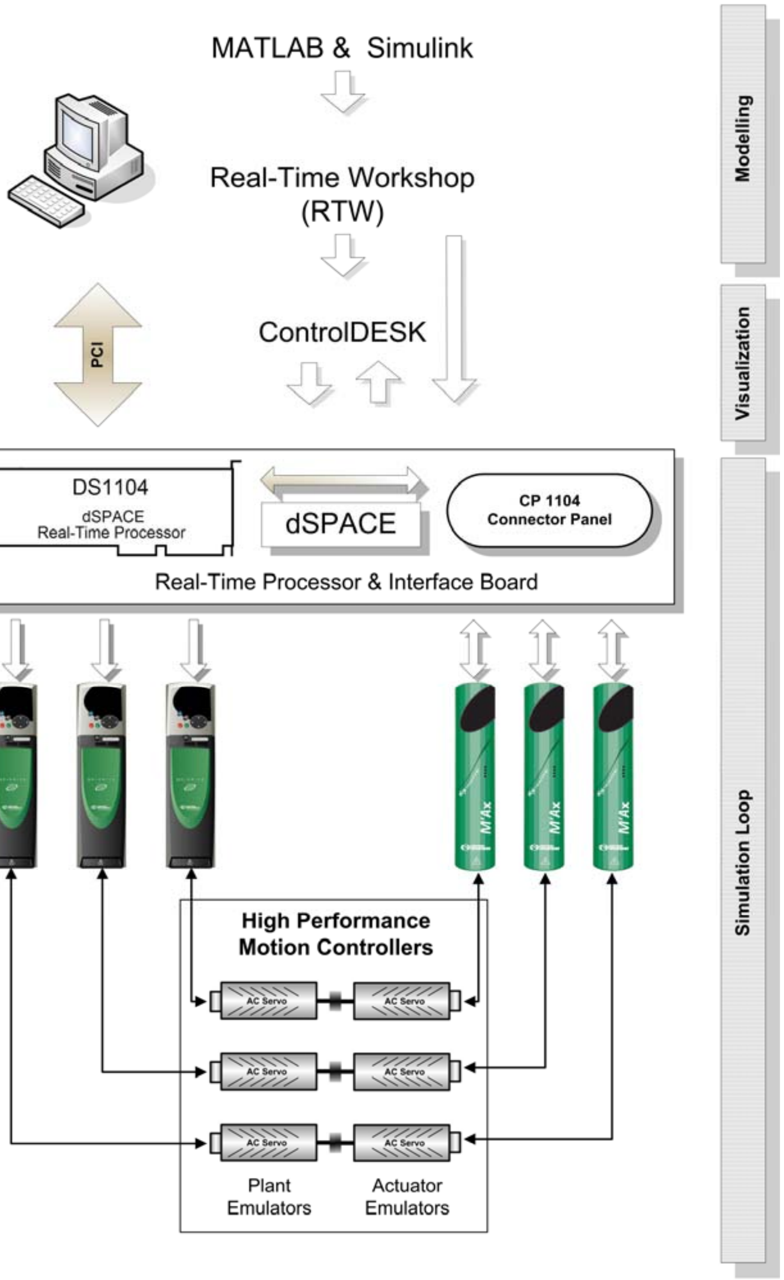


Figure 5.1: General structure of the implemented HIL Simulator

In Figure 5.1 general structure of the implemented Hardware-in-the-Loop simulator is revealed.

Modelling stage is performed in MATLAB / Simulink tools. The desired system model is implemented in Simulink. In ControlDesk program the real-time simulation results can be examined online. The model simulation loop occurs between the dSPACE DS1104 Board and the high performance motion controllers.

5.2 HIL Simulator Architecture

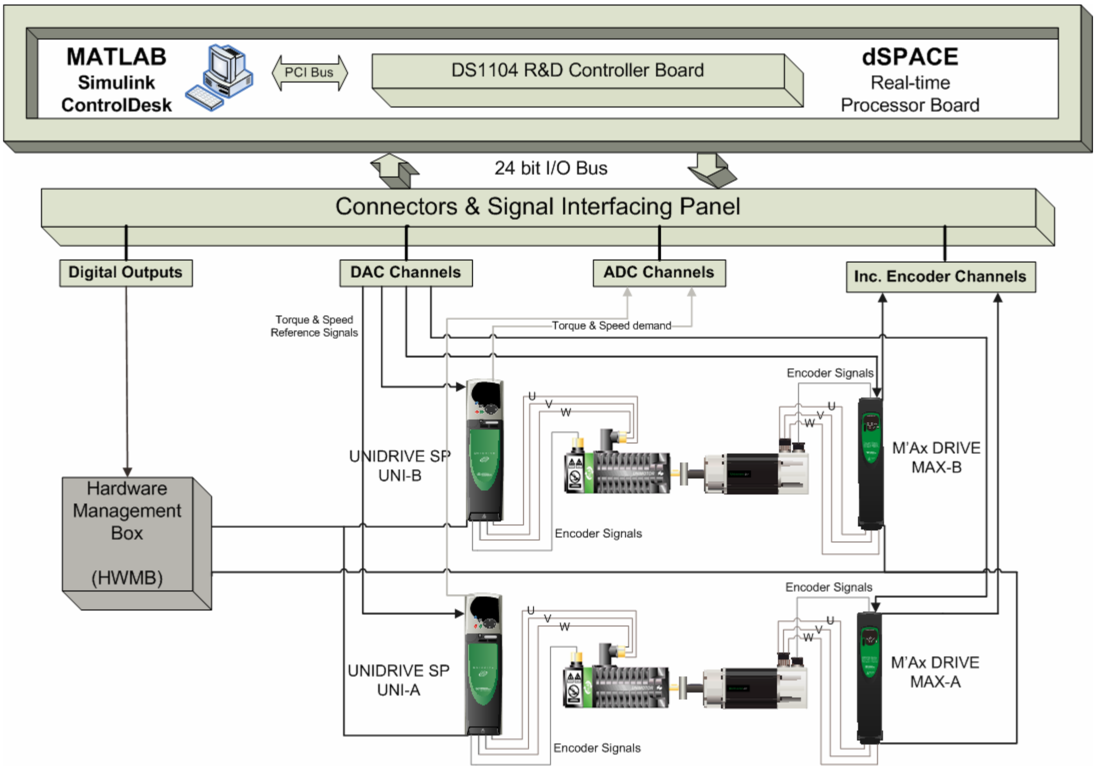


Figure 5.2: HIL Simulator signal connections

In Figure 5.2 HIL simulator diagram and signal, connections can be shown. In MATLAB / Simulink the system model constituted. In Simulink, Real Time Workshop generates the real-time model, Real Time Interface (RTI) builds the real-time C codes for the DS1104 Board Processor, and RTI embed the codes to the processor. CP1104 connector and signal interfacing panel achieve the signaling between the processor and the system. The simulation results are sent to the ControlDesk program continuously. DS1104 board sends the control signals to the motion controllers over DAC channels. Processor receives the position data from its Incremental Encoder Channels. Hardware Management Box (HWMB) controls the drives' “Secure Enable”, “Run” and “Reset” inputs. CP1104 digital outputs (I/O 0-5) used for the “Run” command for the motion controllers. Thus, users can manage the drives and motors from designed ControlDesk Graphical User Interface (GUI). Designed ControlDesk GUI can achieve visualization.

6 EXPERIMENTAL RESULTS

6.1 Simple Pendulum HIL Simulation Environment

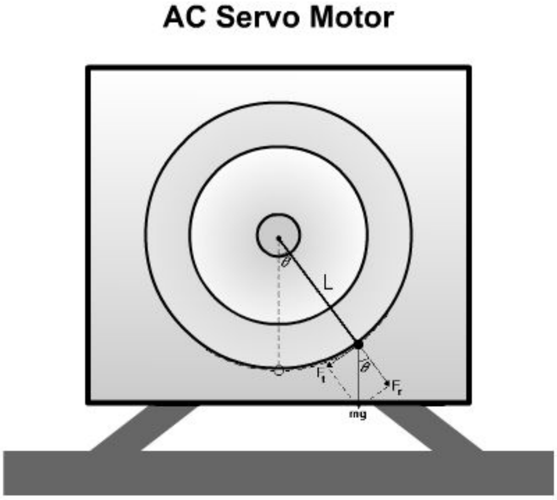


Figure 6.1: Block scheme of the AC Servo motor as the Simple Pendulum HIL simulation

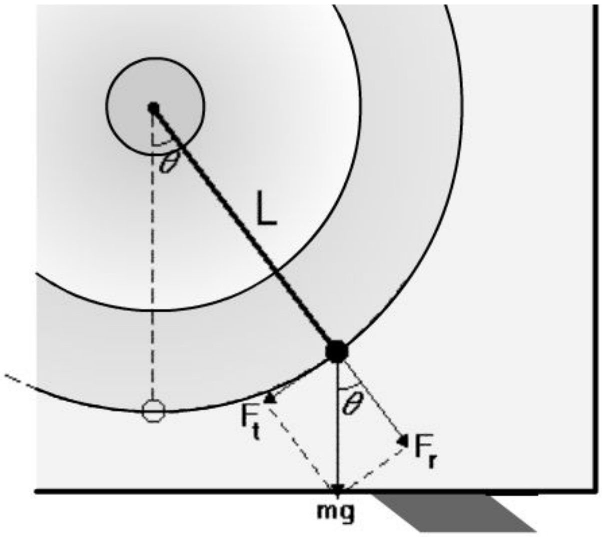


Figure 6.2: Block scheme of the AC Servo motor usage in large scale

In Figure 6.1 and Figure 6.2 the simple pendulum HIL simulation with high AC servo motors is presented. The rotor is used as the simple pendulum's motion surface.

6.1.1 M'Ax Drive Torque Mode With Analog-Input Destination

In order to prepare the motors for the simulation drives parameters have to be adjusted for proper operation in torque mode. The drive parameters adjusted for running in torque mode. M'Ax Drive When the drive is in torque mode (parameter 4.11 set to 1 or 2), and controlled by an analog input parameter 7.10 Analog input destination selector must be set to 4.08, parameter 1.14 Reference selector must be set to 3 to 5.

In Figure 6.3 Simple Pendulum Model designed in MATLAB / Simulink is revealed.

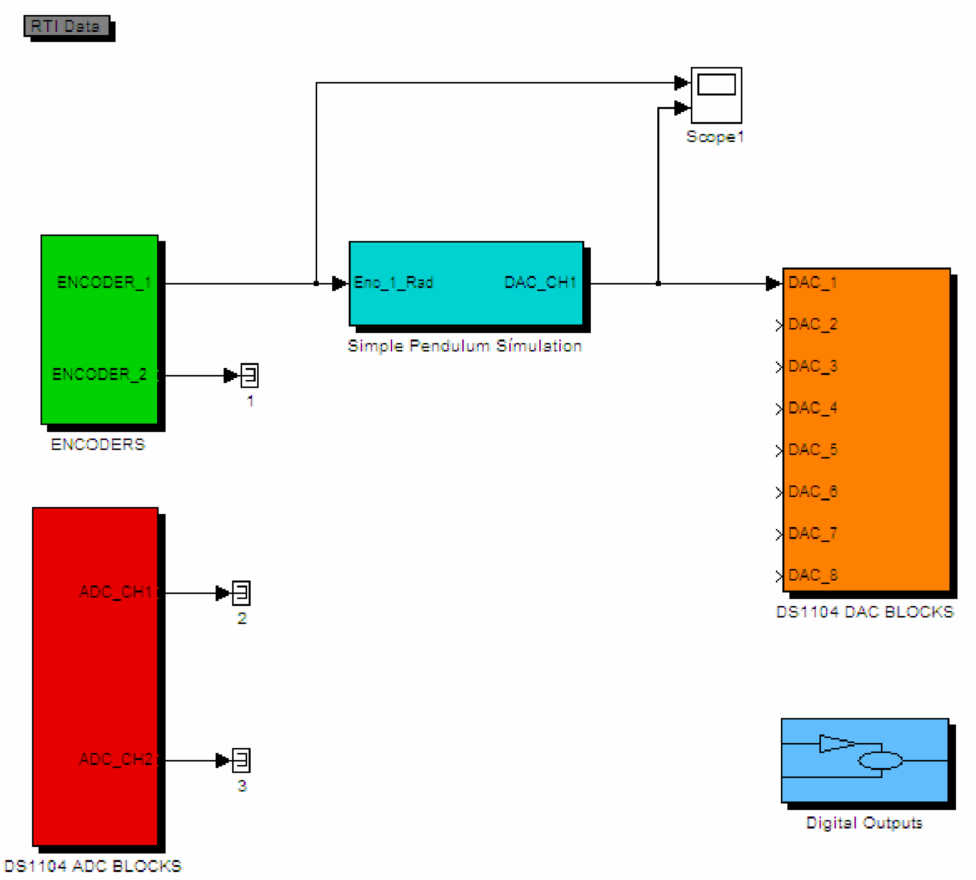


Figure 6.3: Simple Pendulum Model designed in MATLAB / Simulink

This model partitioned into five main blocks:

- “Simple Pendulum Simulation” Block
- “Encoders” Block
- “DS1104 ADC” Blocks
- “DS1104 DAC” Blocks
- “Digital Outputs” Block

6.1.2 Simple Pendulum Simulation Block

Inside of the “Simple Pendulum Model” block in the Simple Pendulum Model can be shown.

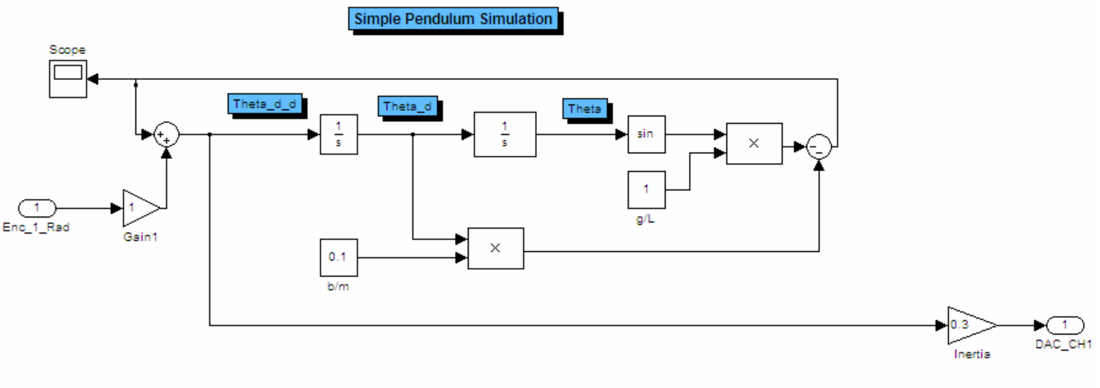


Figure 6.4: Simple Pendulum Model” block

6.2 “Encoders” Block

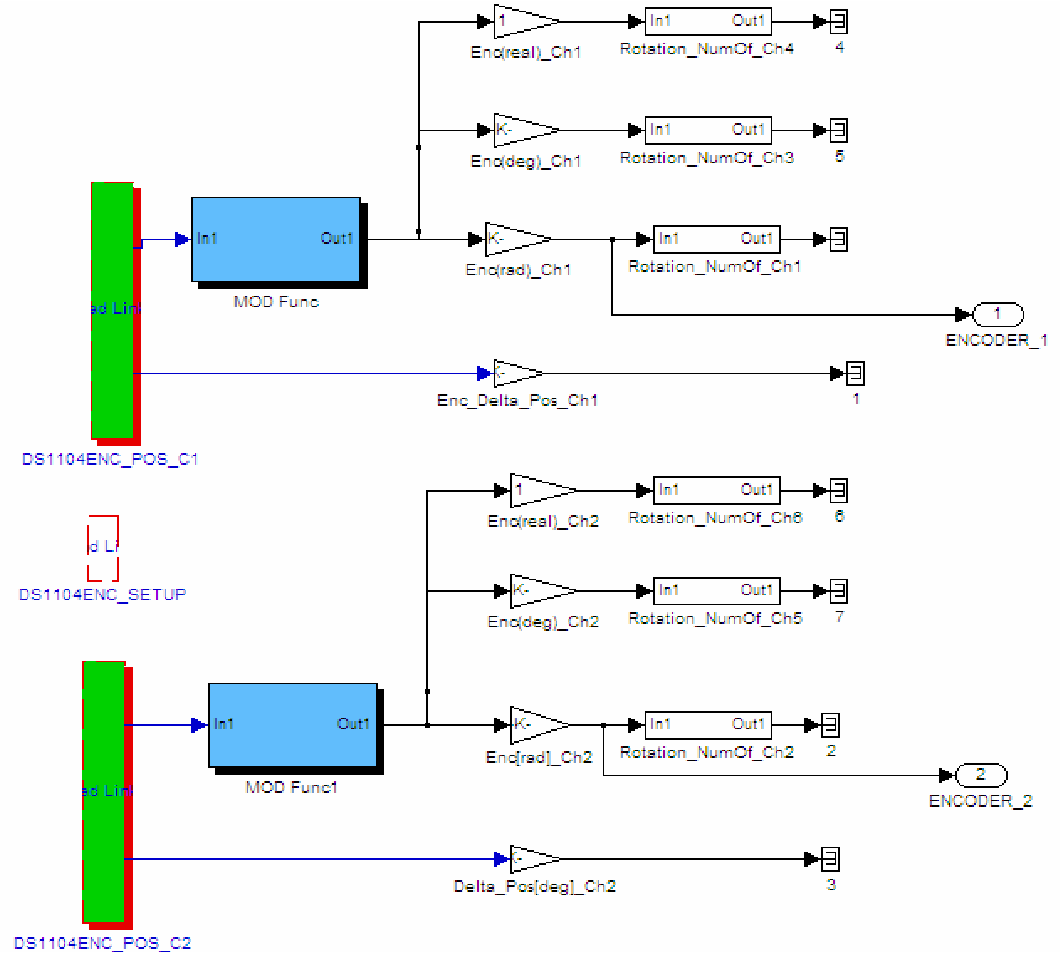


Figure 6.5: Simulink Simple Pendulum Model “ENCODERS” Block

In Figure 6.5 the “ENCODERS” Block is presented.

6.3 “DS1104 ADC” Block

In Figure 6.6 Interior of the “DS1104 ADC” Block in the Simple Pendulum Model is stated.

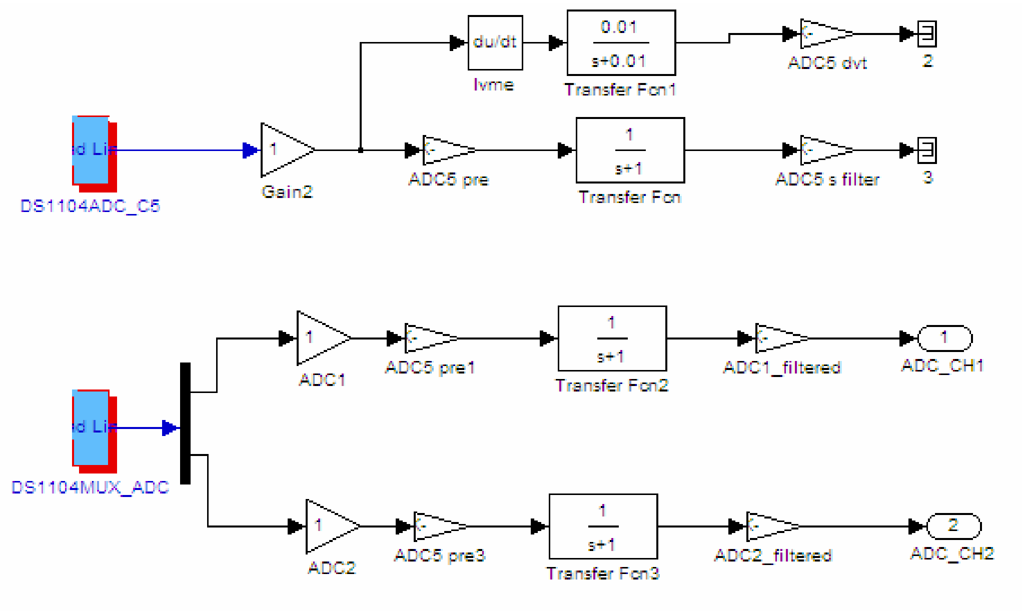


Figure 6.6: Simulink Simple Pendulum Model “DS1104 ADC” Block

6.4 “DS1104 DAC” Block

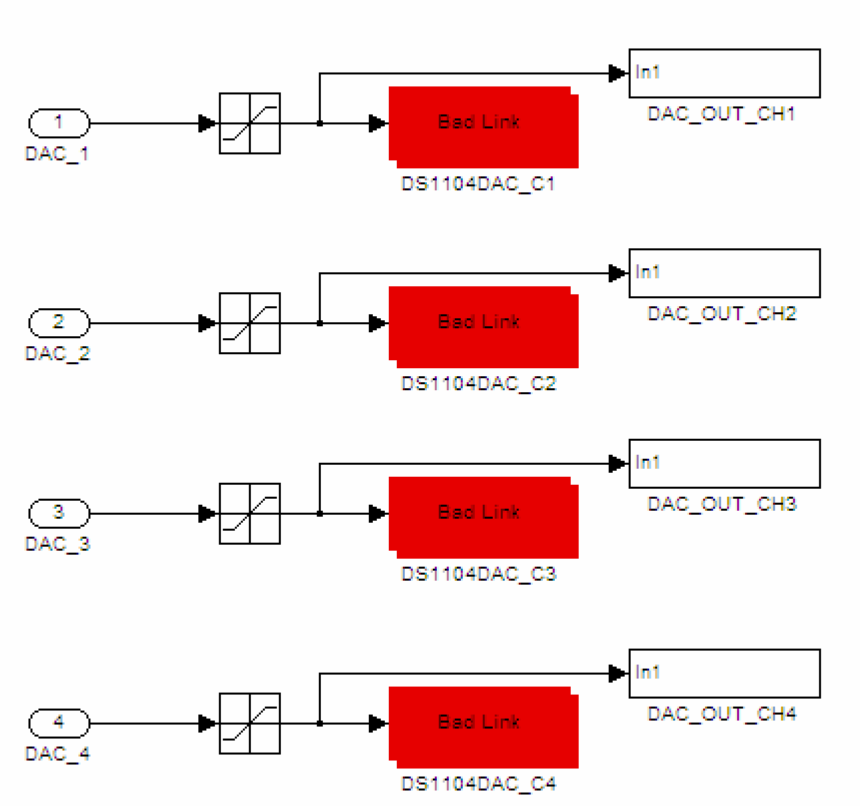


Figure 6.7: Simulink Simple Pendulum Model “DS1104 DAC” Block

6.5 “Digital Outputs” Block

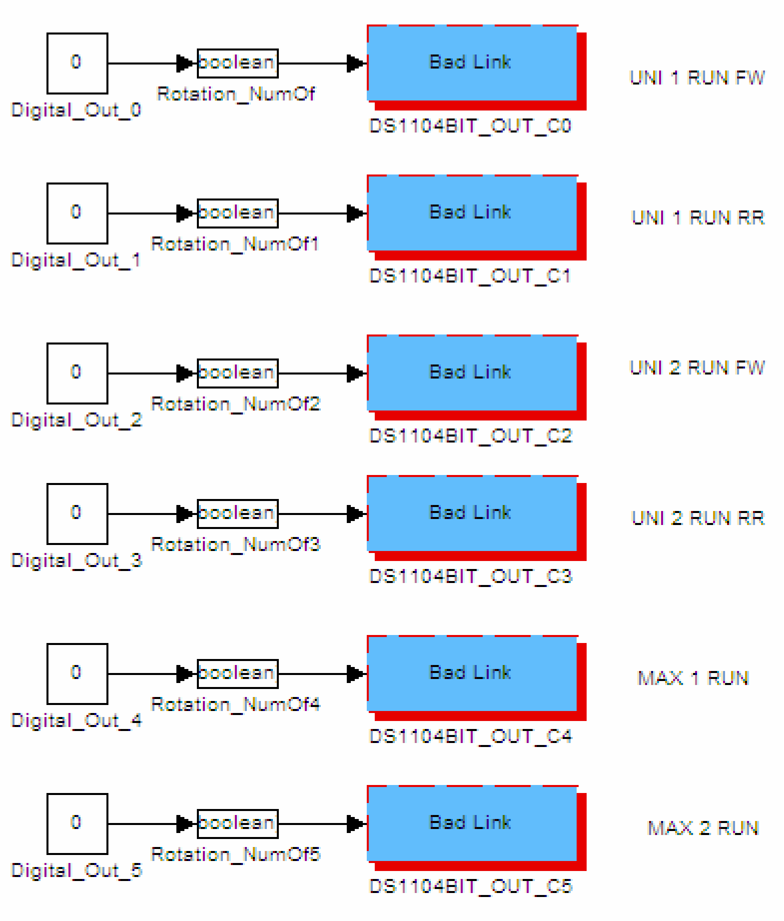


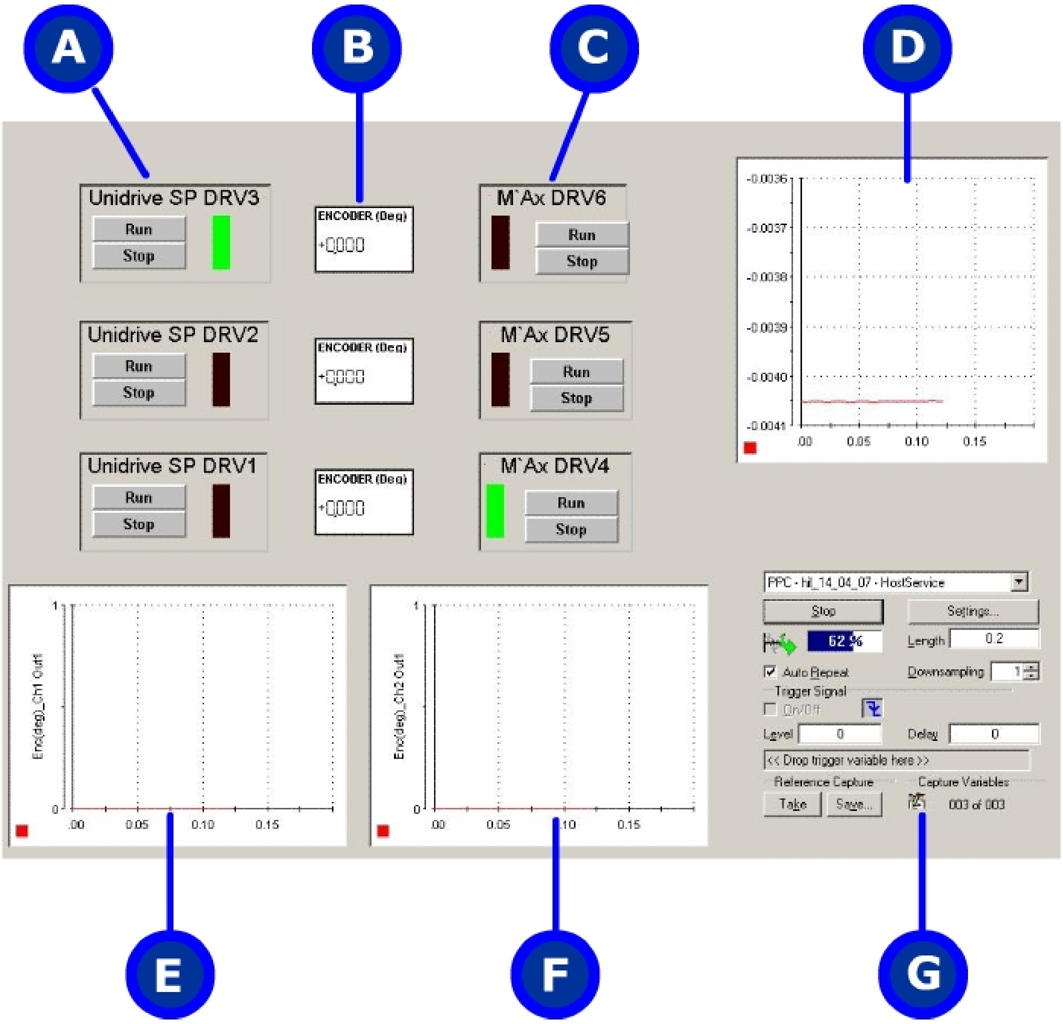
Figure 6.8: Simulink Simple Pendulum Model “Digital Outputs” Block

In Figure 6.8 Digital Outputs block is presented. Digital outputs are used for Run/Stop commands to the drives.

6.6 Graphical User Interface Design In ControlDesk

dSPACE ControlDesk platform allow designers to build a fully interactive simulation interfaces. In Figure 6.9 the designed interface for the HIL Simulation took part.

A) Buttons allow users to send Run/Stop commands to UNIDRIVE SP servo drives. Green LED indicates the running drives. Run/Stop buttons linked to the dSPACE digital I/Os to transmit command.



PPC - hil_14_04_07 - HostService

Length 0.2

☑ Auto Repeat

Downsampling 1

☐ On/Off

Level 0

Delay 0

Figure 6.9: GUI designed in ControlDesk for the HIL visualization

B) Indicator shows the UNIDRIVE SP and M'Ax Drive couple's instant encoder position value.

C) Buttons allow users to send Run/Stop commands to M'Ax servo drives. Green LED indicates the running drives.

D) Graph linked to the ENCODER Channel 1 to show the UNIDRIVE SP DRV2 and M'Ax DRV5 couple's encoder position versus time.

E) Graph linked to the ENCODER Channel 2 to show the UNIDRIVE SP DRV1 and M'Ax DRV4 couple's encoder position versus time.

6.7 Simple Pendulum HIL Simulation Experimantal Results

In Figure 6.10, the HIL simulation results of the Simple Pendulum revealed. This graph pointed out the rotor position from encoder data versus time.

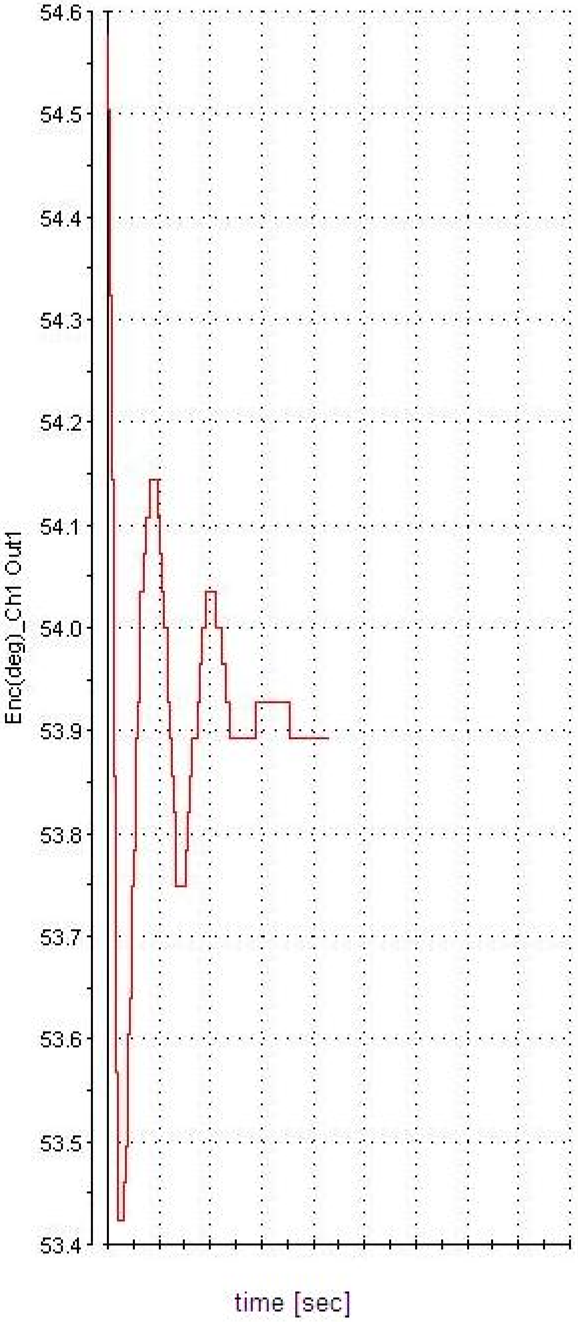


Figure 6.10: HIL simulation results of the Simple Pendulum (rotor position vs time)

In Figure 6.11, the computer simulation results of the Simple Pendulum are revealed. This graph shows the θ, the angle of the pendulum with the vertical axis versus time.

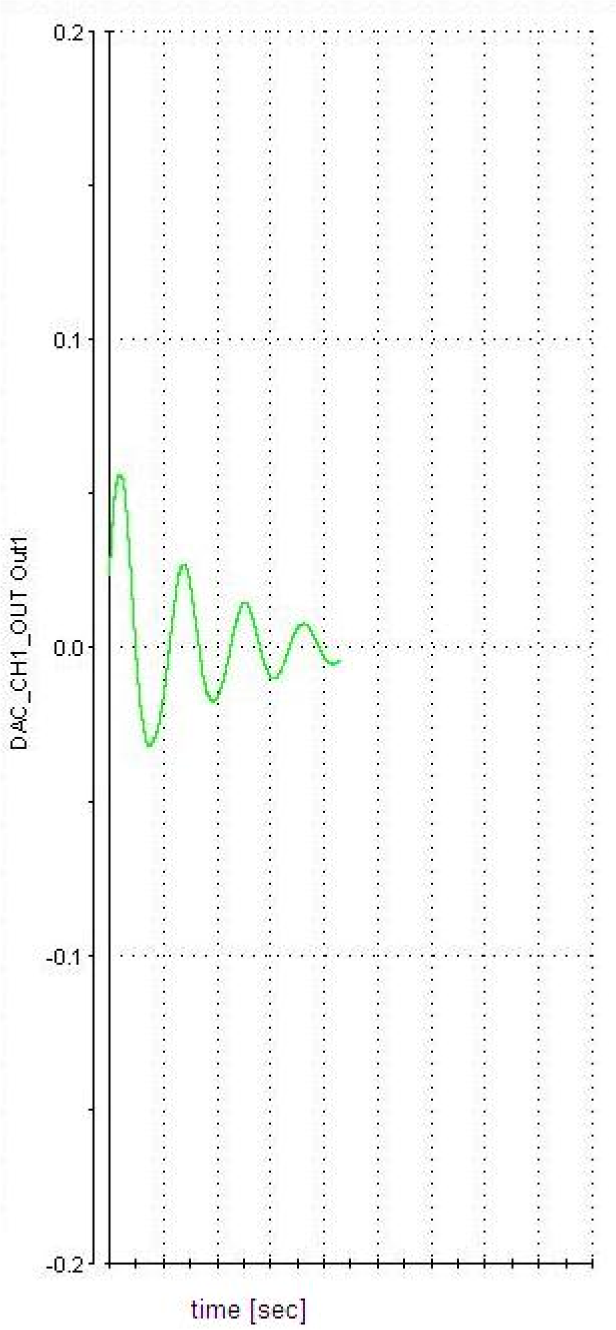


Figure 6.11: Computer simulation of the Simple Pendulum (theta vs time)

7 CONCLUSIONS

7.1 Hardware-in-the-Loop Simulation of the Simple Pendulum

In this dissertation, mechanical systems are intent to be simulated using Hardware-in-the-Loop technique. The HIL technique offers that to incorporate some part of the simulated real system to the simulation loop. A mechanical system is aimed to be simulated by electromechanical system and computer combination. The motion controllers simulate a simple pendulum as a 1-DOF mechanical system. The effective torques on the system is realized as high performance motion controllers. In this point of view, this system can be thaught as an emulator rather than a simulator.

According to the results in Figure 6.10 and Figure 6.11 there exists a remarkable error between the pure simulation and HIL simulation results. A lot of reason causes this error. Firstly, there exists a highly nonlinear and remarkable friction effects especially in the low speeds. Another effect is the permanent magnet rotor's magnetic interactions. Naturally, in the low speeds this effect became more influential.

7.2 Future Work

In this study, a simple mechanical system with a 1-DOF is simulated. The friction effects play a considerable role on the simulation results. To overcome these unwanted effects some regulations could be made on the model. In the high speeds the friction effects become linear and its effects reduces. Thus running in the high speeds could bring some positive effects to the simulation results. The speed of the motors could be increased as if there exists a gear train. Two couples of motors could be operated together as for the 2-DOF system simulation.

REFERENCES

[1] **Pastrick, Harold L.; Holmes, Willard M.; Will, Charles M.; Isom, Larmon S.; Jolly, Alexander C.,** 1973, Real-Tıme Hybrid Hardware-In-The-Loop Simulation of a Terminal Homing Missile, *Summer Computational Simulation Conference, Montreal*, pp 264-268.

[2] **B.M. Hanson, M.C. Levesley, K. Watterson and P.G. Walker**, 2007, Hardware-in-the-loop-Simulation of The Cardiovascular System with Assist Device Testing Application, *Medical Engineering & Physics,* Vol. **29**, Issue 3, pp. 367-374.

[3] **Wu, X.; Lentijo, S.; Monti, A.,** 2004, A Novel Interface for Power-Hardware-in-the-Loop Simulation, *Computers in Power Electronics, Proceedings of IEEE Workshop on Digital Object Identifier,* pp.178 – 182.

[4] **Shah, S.M.; Irfan, M.,** 2005, Embedded Hardware/Software Verification and Validation Using Hardware-in-the-loop Simulation, *Proceedings of the IEEE Symposium on Emerging Technologies*, pp.494 – 498.

[5] **Le, T.; Renner, F.-M.; Glesner, M.,** 1997, Hardware in-the-loop simulation-a rapid prototyping approach for designing mechatronics systems, *8th IEEE International Workshop on Rapid System Prototyping Proceedings,* pp. 116 – 121.

[6] **M. Abou-Samah, C. P. Tang, R. M. Bhatt and V. Krovi**, 2006, A Kinematically Compatible Framework for Cooperative Payload Transport by Nonholonomic Mobile Manipulators, *Journal Autonomous Robots,* Vol. **21**, No. 3,

[7] **Lionel B., Clément M.G.,** 2006, Grasp-state plane analysis of two-phalanx underactuated fingers, *Mechanism and Machine Theory*, Vol. **41**, No. 7, pp. 807—822.

[8] **Grega, W.**, 1999, Hardware-in-the-loop simulation and its application in control education, *Frontiers in Education Conference,* Vol.**2**, pp.12B6/7-12B61.

[9] **Turan, A.; Bogosyan, S.; Gokasan, M.,** 2006, Development of a client-server communication method for Matlab/Simulink based remote robotics experiments, *IEEE International Symposium on Industrial Electronics, Montreal, Que., Canada,* 3201-6

[10] **Singaraju T., Turan, A.; Gokasan, M., Bogosyan, S.;** 2006, Hardware-in-the-loop simulation of PUMA 560 via internet, *IEEE Industrial Electronics, IECON 2006,Paris, France,* 5426-5432

[11] **Temeltas, H.; Gokasan, M.; Bogosyan, S.; Kilic, A.**, 2002, Hardware in the loop simulation of robot manipulators through Internet in mechatronics education, *IEEE 2002 28th Annual Conference of the Industrial Electronics Society*, Vol. **4**, pp. 2617 - 2622

[12] **Spong, M.W.,** 1996. Energy Based Control of a Class of Underactuated Mechanical Systems, *IFAC World Congress* , San Francisco, CA.

[13] **Fantoni, I., Lozano, R., Spong, M.W.**, 2000. Energy based control of the Pendubot, *IEEE Transactions on Automatic Control,* AC-45, No. 4 , pp. 725 -729, April.

[14] **Ortega, R.; Spong, M.W.; Gomez-Estern, F.; Blankenstein,** G.; 2002. Stabilization of Underactuated Mechanical System via Interconnection and Damping Assignment, *Automatic Control, IEEE Transactions,*4, 1218 – 1233.

[15] **M. W. Spong, P. Corke, and R. Lozano.** Nonlinear control of the Inertia Wheel Pendulum. *Automatica*, submitted, September, 1999.

[16] **Spong, M.W., Corke, P., Lozano, R.**, 2001. Nonlinear Control of the Inertia Wheel Pendulum, *Automatica,* Vol. **37**, 1845-1851.

[17] **Spong, M.W., Block, D.J., Astrom, K.J.**, 2001. The Mechatronics Control Kit for Education and Research, *IEEE Conference on Control Applications, Mexico City,* Sept. 5-7, 105-110.

[18] **Olfati-Saber, R.**, 2001. Global stabilization of a flat underactuated system: the inertia wheel pendulum, *Decision and Control, 2001. Proceedings of the 40th IEEE Conference,* 4, 4-7 Dec.,:3764 – 3765.

[19] **Olfati-Saber, R.,** 2000. Control of Underactuated Mechanical Systems with two Degrees of Freedom and Symmetry, *Proc. of American Control Conference, Chicago,*June.

[20] **Spong, M.W., Praly, L.,** 1996. Control of Underactuated Mechanical Systems Using Switching and Saturation, *Proc. of the Block Island Workshop on Control Using Logic Based Switching* , Springer-Verlag.

[21] **Fantoni, I., Lozano, R., Spong, M.W.**, 2000. Energy based control of the Pendubot, *IEEE Transactions on Automatic Control,* AC-45, No. 4 , pp. 725 -729, April.

ÖZGEÇMİŞ

Ömer ÇETİN, 1979 yılında Çanakkale'de doğdu. Orta öğrenimini Çanakkale Anadolu Lisesi'nde tamamladı. Lise öğrenimini Çanakkale Fen Lisesi'nde bitirdikten sonra 1997 yılında Ege Üniversitesi Elektrik-Elektronik Mühendisliğinde Lisans öğrenimine başladı. 2002 yılında lisans derecesini aldıktan sonra 2004 yılında İstanbul Teknik Üniversitesi Elektrik Mühendisliği, Kontrol ve Otomasyon Yüksek Lisans programına başladı.