HIL Simulation as Rapid Prototyping Method in Control Engineering

LORAND BOGDANFFY, EMIL POP, GABRIEL ILCEA
Department of Control Engineering, Computers,
Electrical Engineering and Power Engineering
University of Petroșani
Universității street, nr. 20.
ROMANIA
lorandbogdanffy@upet.ro, emilpop@upet.ro, gabrielilcea@upet.ro

Abstract: In this paper, control engineering design using hardware in the loop is approached. For a better understanding, the first part of the paper clarifies the concepts of Rapid Prototyping, Models in the Loop (MIL), Software in the Loop (SIL) and Hardware in the Loop (HIL). For the different forms of "loop", the general model and possible applications are presented. In the last part of the paper, an application regarding a hydro power system control is modeled and simulated using Simulink-dSPACE platform. The results show the benefits of using rapid prototyping of systems in the case of control design.

Key-words: modeling, simulation, rapid prototyping, model in the loop, software in the loop, hardware in the loop, controller, hydro application.

1 Modeling and simulation principles in rapid prototyping.

Rapid prototyping (RP) appeared and evolved in recent decades as a modern manufacturing technology of relatively small mechanical parts in CNC type machines and later on CIM's.

Currently, control systems engineering is dominated by devices with embedded software. These have parts of software that control a process and that are operating in a loop. In general, both hardware and software parts are connected in a loop to ensure the operation of the system. This mode is used in current technical systems ranging from the simplest, such as stabilizers to complex systems like vehicles for personal or industrial transport.

Current production of equipment, due to its complexity, requirements for reliability and quality, takes a long time, often years, from design, prototyping, testing, to manufacturing and finally shipping.

Modeling and Simulation was the first step in reducing the duration of the design stage. From here, the next step was to produce cars and manufacture equipment or parts on an industrial scale. The concept of rapid prototyping or RP was born. Often, the development and implementation of software is more difficult than that of hardware.

Rapid prototyping can be defined as a series of techniques used to fabricate a scale model of a part using 3D computer-aided design. The first variant was stereo lithography rapid prototyping. This was very useful in designing cars, where it was used for viewing and testing aerodynamics. The basic methodology for all current rapid prototyping in the field of mechanics is based on the following:

- A CAD model is built and converted into STL model.
- The RP machine processes the STL file by creating layers of the model
- The first layer of the physical model is created. The size of the second layer is lowered and the process is repeated until the end of the pattern.
- The physical model is removed, cleaned and checked.

A wide range of applications use this approach to manufacture the finished parts in small or large numbers, as desired.

The problem is more complex when applying RP to control engineering. In this case, the use of well known design models is recommended to avoid unpredictable design errors. In control engineering the model functions based on a mathematical model. Mathematical models for control engineering describe the system accurately, after that the next step is to simulate the model. Model simulation results allow corrections and finally the physical construction of the system. Simulation is necessary either because using a real system would cost too much, for reasons of security and safety or simply because the hardware is not available yet for testing.

Systems engineering uses three methods: Model in the Loop (MIL), Hardware in the Loop (HIL), and Software in the Loop (SIL). Modeling platforms such as Matlab-Simulink are common and useful.

In this paper we work with the following concepts:

- **The process (P)** is the assembly or facility subject to be controlled.

- **Controller (C)** drives or controls the process.
- **The System (S)** is the combined controller and interconnected process (S & C).
- **Hardware (HW)** represents the interconnected physical part of the system giving its physical form. Often times only a HW controller is used.
- **Software (SW)** or programs that access the hardware of the system and ensure its functionality.
- **Computer modeling** is achieving a working model of the functions of a system based on its mathematical model.
- **Simulation** working with the model so that it accepts input and generates output data, thus verifying the correctness of the model.
- **Loop (LOOP)** is the interconnection of P-HW-SW system components that ensures operation and communication between them.
- **Model in the loop (MIL)** is the case where all three parts of the system (P-HW-SW) are simulated for checking operation and response of the system.
- **Hardware in the loop (HIL)** works by testing hardware with embedded software connected to a platform with appropriate software running a simulated model of the process (P). Platform simulates the process, it connects to the controller, making real-time simulation system.
- **Software in the loop (SIL)** is another case in which a software model is tested, usually a prototype software built on the principles of software engineering, that connects using a hardware interface, together ensuring system simulation.

In systems engineering, rapid prototyping of control systems can be achieved with *MIL, HIL, SIL,* which involves the use of a modeling-simulation platform. The Modeling and simulation platform Matlab-Simulink together with dSPACE is well suited for rapid prototyping in systems engineering using MIL-HIL-SIL methods. The following are some guidelines:

- Whatever the type of simulation, a process that can not be brought into the laboratory is simulated on a modeling platform using the mathematical model.
- Process management is performed by a controller with embedded software.
- The connection between the controller and the process is done in loop with interfaces connecting sensors, transducers and actuators with real outputs of the process.
- The Loop system has three parts: process (P), and interface controller (C) & (I) and software (SW). Often interfaces are included in the process, sometimes the controller or both.

Regarding the differences between the two systems SIL and HIL, the most important are:

- MIL is a test in which all parts of the system (C, SW, P) are modeled / simulated.
- HIL testing is a process that connects a real controller to a simulated process and recording the system behavior. For example in the automotive industry the vehicle or engine controller (ECU) is connected to a platform that simulates real conditions in which it will be used.
- SIL is a simulation and test method where sensors or actuators are simulated via software. For example, for a car central locking system, the SIL is the software that controls the actuators on doors and receives information from sensors showing their opening / closing.

In the following part, the philosophy of modeling and simulation hardware-software in the loop for a process is controlled by embedded software is presented. In fig.1.1. we see the block diagram of general modeling / simulation, control loop, MIL-SIL-HIL, which are dotted line blocks to model / simulation platform. The blocks are:

- controlled process, which usually can not be brought on the modeling / simulation platform;
- Interface with translators and sensors, including the ones that can be sometimes brought on the modeling / simulation platform;
- Interface with Servo Actuators and including that which can often be made on the platform modeling / simulation;
- controller with embedded software, can be always brought on the platform modeling / simulation

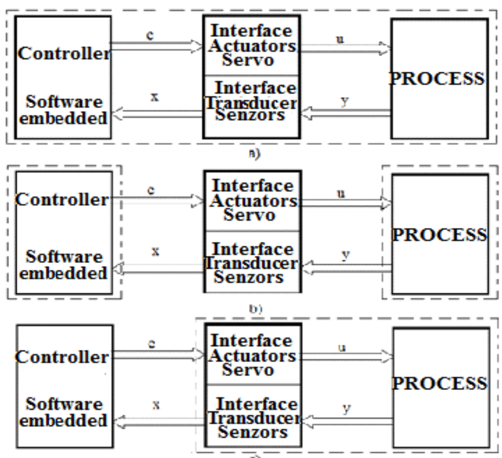

Fig.1.1. Modeling and simulation principles: a) MIL; b) SIL ; c)HIL

In conclusion we can say that MIL-HIL-SIL levels represent faces of modeling-simulation of a system that depends on a process and which part of it is modeled. If we want to test the proper functioning of a system controlled by a hardware controller, then we make a model of the process and eventually interface with the controller, implement simulation model on a platform that has outputs that connect to the controller. We run the model (simulate) and obtain data that will allow us to see virtual system operation. HIL simulation modeling is most commonly used in rapid prototyping method. If we want to design software to control a system, we simulate the operation of software and process on an adequate simulation platform and we connect one and other through real interfaces obtaining SIL. Before doing HIL-SIL modeling, it's recommended to do a MIL modeling / simulation, that will gives us useful information which will then help the other two methods.

MIL, HIL, SIL is the foundations of rapid prototyping in systems engineering.

Next we insist on the HIL method since it is the most used in the RP stage in systems engineering. Then we will present a few of HIL methods that we will continue to apply.

As shown, HIL is a method of modeling simulation of real processes that combine real hardware with a programming environment with embedded software that provides results in real time during the design-prototyping.

This kind of simulation modeling is used in many areas of production and is a modern method of obtaining a new product.

HIL is a technique used in the development and testing of complex real-time embedded systems. HIL Modeling and Simulation is achieved by using mathematical modeling of dynamic systems which cannot all be brought to the platform and connected to real hardware.

A HIL simulation has to include an emulation of the sensors and actuators used. They are designed to interface between system and test platforms. Each of the simulated electrical components is controlled and read by the test platform.

For example, if we use HIL simulation for research and development of a vehicle ABS system, the following need to be taken into consideration:

- vehicle dynamics, such as suspensions, wheels, tires, tilt angles of the 2 axis;
- the dynamics of hydraulic brake system;
- the characteristics of the road on which the vehicle is.
- sensors which measure the condition of the vehicle and ABS.

In many cases, the most effective way of developing an embedded software system is by connecting it to an actual real environment but the results could be unsatisfactory. Therefore HIL simulation is more effective. Usually the choice is made taking into account the factors which contribute to RP namely: improving the quality of testing, limited time devoted to research, testing and putting into production, the possible high cost of the real process/plant, simulating the human factor etc.

2. Using MIL-SIL-HIL on a hidro system

2.1. Analysis of the controlled process

The aim is to show how modeling / simulation MIL-HIL-SIL is essentially a rapid prototyping of a control system. In this regard, it is particularly the case for automatic control of a process that uses a hydro pump to drain a basin constructed in advance.

Pump stations have mostly a certain level of automatic control using wired logic. This is mainly due to higher power engines and switching elements and that make pumps more difficult to control. Before starting full automatic control, a priming operation done manually is usually preferred. In fig.2.1. an automated system with wired logic controlling a two-pump station is used

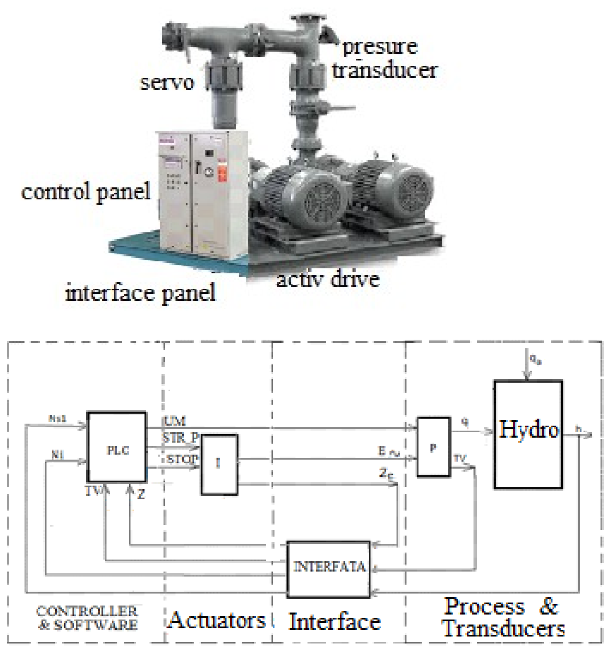

Fig.2.1. Two-pump station control: a) general view, b) control system

Water tank has a volume *V*, a depth *h* and surface *S* for a flow accumulation of *qa*. When the water reaches a level of 80-90% of the height of the tank (*NS1)*, water must be emptied. Before that, however, pumps need to be primed by servo-valves, filling that last usually tens of seconds. After filling the pump is switched on by a switch *I*. Water pressure is controlled using a pressure transducer or more often an engine speed sensor *TVs*.

If the pump is not primed it must be stopped, refilled, restarted but for no more than 3-4 times, after that a failure is considered and the backup pump starts following the same procedure. Once the pump goes into operation, it will drain the water until the level in the tank reaches a value of about 10% of maximum, called *Ni,* lower level, and the pump will be stopped. Electro-mechanical protections of the system identified by Z will stop the pumps and control system in case of failures or function anomalies.

We will determine the mathematical model of the process, where we noted 1/s integration operator. During Water accumulation phase, we have:

dV/dt=Sdh/dt=qa. That means: h=qa/S*1/s+Ni.

During Emptying of tank:
dV/dt=Sdh/dt=qa-q1. So: h=(qa-q1)/S*1/s+Ns1.

The process is controlled by a PLC equipped with interface blocks *Ni, Ns1,TV, Z*, actuators and servos for filling and starting the pumps according to the diagram in fig.2.2.b.

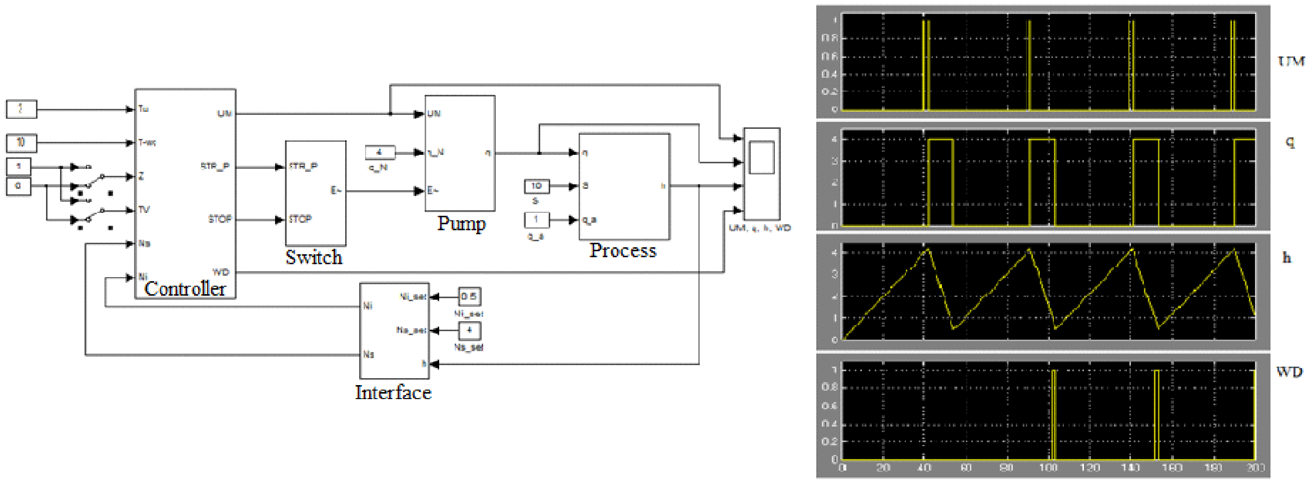

Fig.2.2. MIL; a) Bloc design, b) Simulation results

2.2. Determining requirements, MIL type modeling and simulation of control

The presented description allows us to enumerate the requirements that the controller will have via software :

- Reaching the high level *Ns*, a command UM needs to be issued
- filling rate is controlled by constant *Tu*
- after filling, the breaker *I* is closed by *STR_P* command
- *P* breaker starts the pump motor is controlled by transducer *TVs*
- the reaching of the lower level *Ni* issues *STOP* command
- Watchdog of a PLC, checks the time constant *Tw* and pump priming status
- if WD stopped the pump, an attempt to be primed about 3 or 4 times is carried out
- Any electro-mechanical failure will issue a *Z* command, causing pump and system stop
- If the pump was stopped by *Z* command, system will operate only after removing the cause of restarts

The transducers used are:

- The transducer *superior* and *inferior Ni, Ns*
- *TVs* encoder speed and crash sensor *Z*

The signals from the transducers are processed by the interface and then sent to a PLC controller. The interface generates from the analog signal *h,* the logic signals ***Ns*** and ***Ni***, from the analog *Tv* the digital signal *Tv* and all logic signal from the transducer Ze is adapted logic signal Z. Thus all input signals are logical type.

In fig.2.3. the transducers in MIL simulation for speed and crash are manual switches.

The simulation results are presented in the same figure. As water builds up in 40 seconds and takes 2 seconds to start. If the *Tv* signal does not arrive in time intervention of WD causes shutdown.

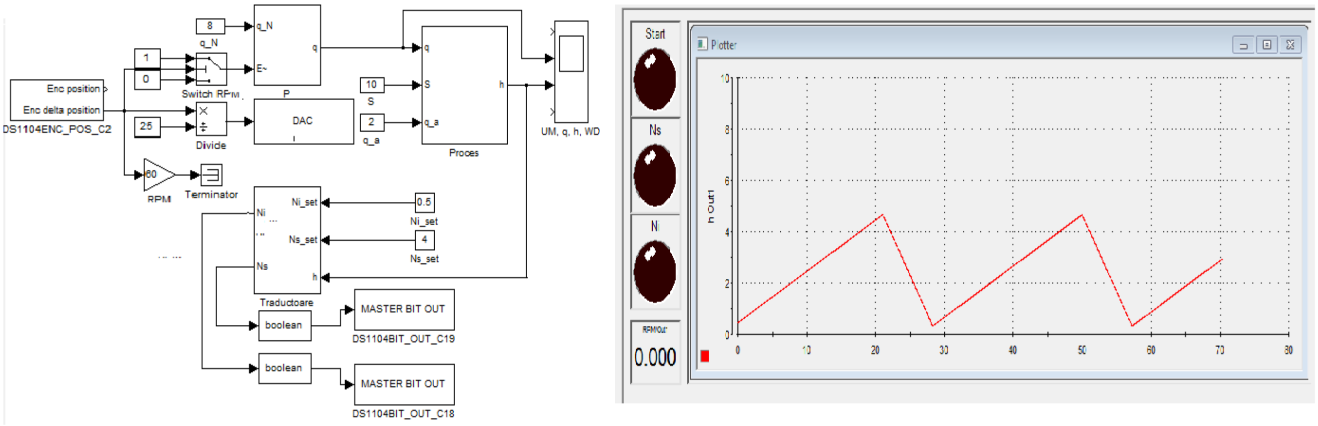

Fig.2.3. HIL model for simulation on dSPACE platform

2.3. SIL-HIL modeling and simulation of the hydro process control

In fig.2.4. the internal controller scheme is presented, and the level transducers are simulated. In this case the software will function through the following blocks:

Water level translators using analog comparison

Command STR_E by using a flip switch S/R

Command filling with a *time-out-delay relay*

Command STR_P using a logic function: STR_P=STR_E*!UM

Command STOP using a logic function: STOP=Ni+Z+WD

Command WD by a time-on-delay STR_P and canceled by *Tv*

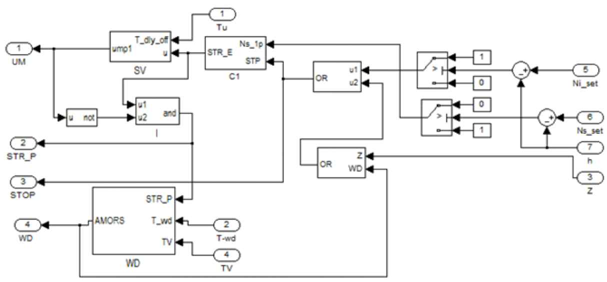

Fig. 2.4. Simulated controller

The software will be built according to algorithm in fig.2.5, in Ladder language of the Moeller-Easy PLC. Three loops are distinguishable: START/STOP, FILLING and test by the WATCHDOG. Each loop is programmed in Ladder with high efficiency, using only a few instructions, as shown in the following figure:

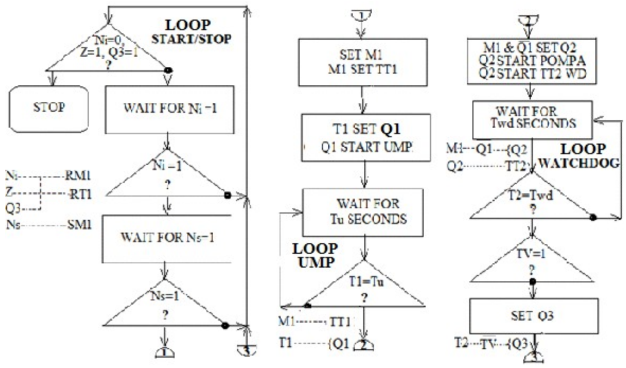

Fig.2.5. Algorithm and LADDER program

In fig.2.6, the connection of the PLC running the described program to the Simulink-DSPACE platform, that will contain the simulation of the process, the interface with the PLC, the pump motor and HIL simulation results.

Conclusions

- The concept of RP, MIL, SIL and HIL are presented and justified.
- The HIL simulation is applied on the hydro system in order to design an embedded system control.
- The results are useful for RP and can be extended of other class of applications.

Reference:

1. T. Hwang, J. Rohl, K. Park, J. Hwang, K. H. Lee, K. Lee, S.-J. Lee, Y.-J. Kim, Development of HIL Systems for active Brake Control Systems, *SICE-ICASE*, 2006.
2. J. Du, Y. Wang, C. Yang, and H. Wang, Hardware-in-the-loop simulation approach to testing controller of sequential turbo charging system, *Proceedings of the IEEE*,

2007.
3. Pop E., Bubatu R., *Teoria Sistemelor. Educatie prin e-Learning*, Editura Universitas, 2012.
4. W. Leonhard, *Control of Electrical Drives,* Springer, 1996.
5. E. Pop, Automatizari in industria miniera, EDP-Bucuresti, 1993
6. Pop E. - *Kalman Filter Design, Modeling, Simulation and Practical Applications*, Proccedings of the 12th International Conference on Applied Computer and Applied Computational Science (ACACOS `13), WSEAS, ISBN: 978-1-61804-171-5, April 2013, Kuala Lumpur, Malaysia
7. Pop E., Patrascoiu N. - *Practically Algorithm, Modeling and Simulation of Luenberger Observer*, Proccedings of the 12th International Conference on Applied Computer and Applied Computational Science (ACACOS `13), WSEAS, ISBN: 978-1-61804-171-5, April 2013, Kuala Lumpur, Malaysia
8. Pop E., Patrascoiu N. - *Simulation and Design Algorithm for Identifiers. Applications*, Proccedings of the 12th International Conference on Applied Computer and Applied Computational Science (ACACOS `13), WSEAS, ISBN: 978-1-61804-171-5, April 2013, Kuala Lumpur, Malaysia
9. Avram A., Pop E., Barbu C. – *VLSI Embedded Solution for Multi-Drive Conveyors Control*, Proceedings of the International Conference on Applied Computer Science (ACS), WSEAS, Malta, September 15-17, 2010, ISSN: 1792-4863, ISBN: 978 960 474 225 7