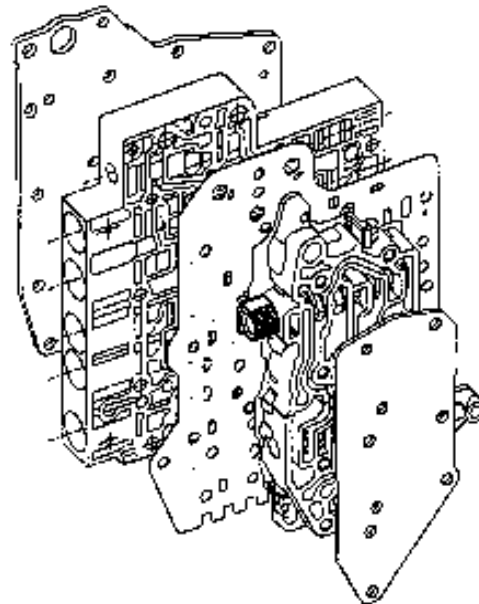


# How to Create Good Models without Writing a Single Line of Code

Technical Bulletin n°101

---



## How to contact IMAGINE:

North America      [imagine-us@amesim.com](mailto:imagine-us@amesim.com)

Europe              [imagine@amesim.com](mailto:imagine@amesim.com)

Asia                 [imagine-japan@amesim.com](mailto:imagine-japan@amesim.com)

Visit [www.amesim.com](http://www.amesim.com) for further contact information and details on other countries.

*Copyright © IMAGINE S.A. 1995-2002*

*AMESim® and AMESet® are the registered trademark of IMAGINE SA.  
All other product names are trademarks or registered trademarks of their respective companies*

*Latest update: Feb. 26<sup>th</sup>, 2002*

---

[www.amesim.com](http://www.amesim.com) – [www.amesim.com](http://www.amesim.com) - [www.amesim.com](http://www.amesim.com)

**IM****GINE**

*Copyright © IMAGINE S.A. 1995-2002*

  
**AMESim**

# How to Create Good Models Without Writing a Single Line of Code

## 1. Abstract

The story of simulation and modeling is one of qualified success and ever increasing model complexity. What is routine today was inconceivable three years ago. However, we are all painfully aware of many cases of grossly inaccurate and inefficient models.

The problem lies in the complexity that is expected in the world of the virtual prototype. Acquisition of modeling skills involves a long learning curve taking in areas such as engineering, physics, mathematics, computer science and control. Educational institutions are unable to provide this breadth of skills in undergraduate courses. The result is a series of highly repetitive classic blunders, long development times and poor models.

The theory of simulation provides two paradigms in signal port and multiport modeling. Signal port modeling is the more highly developed, but is inappropriate outside the world of control. Multiport modeling is far more general and signal port modeling can be regarded as merely a special case. The theory of bond graphs represents a bold attempt at rationalizing multiport modeling. A key feature of bond graph theory is the description of all engineering systems in terms of 9 basic elements. It has been successfully employed in a number of the more enterprising modeling courses. However, its abstract nature has limited its adoption.

[www.amesim.com](http://www.amesim.com) – [www.amesim.com](http://www.amesim.com) - [www.amesim.com](http://www.amesim.com)

**IMAGINE**

Copyright © IMAGINE S.A. 1995-2002

  
**AMESim**

In this paper the problem of reticulation of the model into a collection of submodels is addressed. A level of reticulation is proposed in terms of basic technological elements. This is higher than the bond graph level but lower than the normal multiport technological level. This approach reduces the number of elements to a manageable size but must be complemented by a 'shrink wrap' facility which is already common in signal port software. The result is an environment in which the engineer is able to create very complex models without writing code. The collection of skills required is reduced to a level reasonable for a design office technician. Coupled to modern efficient numerical solvers, the environment leads to an enormous increase in productivity.

The essence of the methodology proposed is to combine a collection of existing software and modeling techniques into a single environment. The process is illustrated with a series of examples.

## 2. Introduction

It is no longer necessary to appeal for greater use of simulation in the engineering industries. There has been a steady spread of its use from large sized industries, to medium sized industries and now to small industries. Within each industry there has been an expansion of simulation from the research department to the drawing office. [1], [2], [3], [4].

This movement could not have taken place without a record of success for simulation in the design process. The key factors have been

- Reduction of the need for testing physical prototypes,
- More thorough testing of the product,
- Convincing the potential client that the product will be worthy,
- Cutting the time to bring the product to the market place,
- A better understanding of the product.

In the automobile industry the process is well advanced. It is becoming increasingly common for companies to require their subsystem suppliers to provide a model of the subsystem before the physical subsystem is delivered.

However, despite this success there are plenty of instances of poor models of engineering systems. Below are some classic blunders. The list is far from complete but is representative of the problems that occur.

- Failure to observe a fundamental physical phenomenon such as conservation of mass.
- Failure to check model inputs to eliminate gross errors such as a stiction level less than a Coulomb friction level.
- Failure to perform adequate testing of modules within the model.
- Inappropriate level of complexity e.g. too much detail and high dynamics within a large system.
- Use of a physical formula outside its domain of validity (often leading to numerical problems)
- Domain violation in using of mathematical functions e.g. square root of a negative number. This often occurs when the integrator is iterating to convergence with an implicit method.
- Mismatch of physical units e.g. bar and Pa.
- Calling function or subroutines with incorrect arguments.
- Problems in modeling physical phenomena using discontinuities.

These problems can lead to bad results but also to excessive times to construct, debug and validate a model.

The core problem is the extensive collection of skills necessary to construct good models.

### 3. Skills necessary

Here are some skills necessary to construct a good model:

- Knowledge of the engineering system
- Knowledge of underlying physics
- Mathematical skills in manipulating the governing equations into a suitable form for the model
- Coding skills to implement the mathematical model
- Numerical skills in understanding the interaction between the model and the numerical algorithms used to perform the simulation
- Ability to interpret the results

This is an impressive collection of skills and it is not surprising that undergraduate engineering courses fail to cover all of them. Undergraduate students specializing in mathematics or physics will do no better.

In order to understand the role of these skills in the design process, it is useful to examine a variety of approaches to simulation.

### 4. The signal port approach

There are an extensive collection of signal port software packages which are widely used for simulation of engineering systems. Simulink [5], XMath [6], Easy 5 [7]. They display strong influences from analog computer practices.

In the signal port approach there are blocks, which are connected together via ports. At each port there is a one-way flow of information. This is either a single value or a vector of values. This is a very natural way of doing things with control systems. However, these software packages are often used to model the physical system that is controlled as well as the control system.

Many prebuilt blocks are provided. Sometimes it is possible to build the complete system from these blocks. This avoids much of the mathematical manipulation skills and all of the coding skills. However, knowledge of the underlying physics is required in order to express the physical system in terms of these blocks.

As the system becomes bigger, the following problems begin to occur:

- There are too many wires,
- There is too much scope for misconnections between blocks,
- The system diagram looks nothing like the physical system,
- Sometimes the blocks needed do not exist.

The first two problems are partially overcome by ability to group collections of blocks into a single icon. This can be described as a supercomponent or shrink-wrap facility.

The last problem must be solved by creating your own block. This brings back the need for mathematical manipulation and coding skills.

## 5. The multiport approach

With the multiport approach, like the signal port approach, there are blocks connected at ports. However, unlike the signal port approach, there is generally a two-way flow of information between connected ports. This is because this approach was developed to model the transmission of power. Normally the two pieces of information are described as an effort and a flow and their product represents a power.

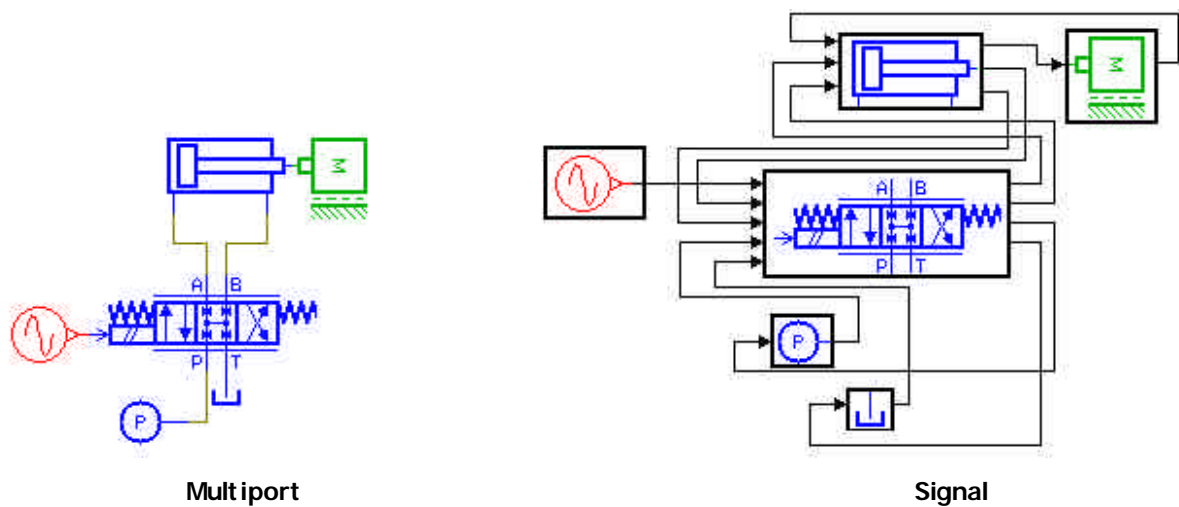


Figure 1

For systems involving transmission of power, the multiport approach is much more natural. **Erreur! Source du renvoi introuvable.** Figure 1 shows a servovalve and hydraulic jack represented by the two approaches. Note the extra wires needed in the signal port approach. Note also that for the port supplying the command signal to valve, there is one-way flow of information and so the two approaches are the same. In other words, the signal port approach is simply a special case of the multiport approach.

Multiport software includes AMESim [8], Saber [9], ITI-Sim [10] and Flowmaster [11]. They use icons of components from which the system is built. Behind each icon there is one or more submodel. Where there are standard symbols for components, icons are based on these. Thus, for example, hydraulic component icons can be based on ISO standard symbols. This imposes a division or reticulation of the complete model into submodels. This reticulation is not necessarily the best from a modeling point of view.

If no such standard symbols exist, it is necessary to create a collection of icons which define the reticulation of the system. Normally these are designed to look like the physical components. Many different reticulations are possible. The choice of which one to adopt can be difficult.

Criteria for deciding which one to adopt must include the following:

- The need to avoid implicit loops,
- At a level which makes the icon and corresponding submodel(s) as reusable as possible,
- At a level which reduces the modeling problem to a reasonable level.

The multiport approach at its best removes the need for mathematical manipulation and coding. It also eliminates the need to understand the fine detail of the physics but keeps the broader principles. Thus it is not necessary for the user to worry about the difficult problem of ensuring mass conservation is maintained during air release and cavitation conditions in a hydraulic pipe. However, it is important to consider whether wave propagation is significant in the pipe.

The traditional problem with the multiport approach is one of diversity. No matter how many icons and submodels are provided, it is not enough. This means that the user is forced to create new submodels and this means the full collection of skills again become necessary.

As an example consider a hydraulic jack. This can have:

- One hydraulic chamber or two hydraulic chambers
- No springs, one spring or two springs
- One rod or two rods
- A fixed body or movable body

So far we have 24 distinct icons. If we allowed all the different possible causalities on the 4 ports, we would have 224 submodels to cover hydraulic jacks. This is not feasible.

The rest of this paper addresses the problem of how to eliminate the diversity problem. If this goal is achieved, it will be possible to write a good model of an engineering system without writing a single line of code.

The authors believe the key to solving the problem is the choice of the reticulation used. To develop the argument we first look at the bond graph approach as a possible solution.

## 6. The bond graph approach

The idea behind bond graphs is to extract the essential characteristics of engineering systems independent of the application domain. [12], [13]. This leads to a special multiport approach which expresses all engineering systems in terms of the following 9 elements:

- C** capacitance
- I** inertia
- R** resistance (friction)
- TF** transformer
- GY** gyrator
- 0**-junction
- 1**-junction
- SE** effort source
- SF** flow source

This is a division of the system in terms of the physics. This contrasts with the division in technological terms used by common multiport software. Theoretically with 9 bond graph icons we can describe any engineering system. This promises a solution to the diversity problem. However, it does place a high stress on understanding the underlying physics of the system.

Bond graph software such as Enport [14], MS1 [15] and 20Sim [16] use this approach. Normally one or more submodel is provided for each element but often it is necessary to create additional submodels by writing code. The nature of each bond graph element imposes restrictions on the equations that can be entered. This restricts the complexity of the submodels that can be produced. This is both a strength and a weakness. The modeler is constrained to enter equations in terms of transmission of power and conservation of energy but the complexity of the model is

also constrained. This means the bond graph approach is unsuitable for high-level simulation.

If we allow an average of 6 submodels per element, we would have only 54 submodels. This is very manageable.

Disadvantages of the bond graph approach are as follows:

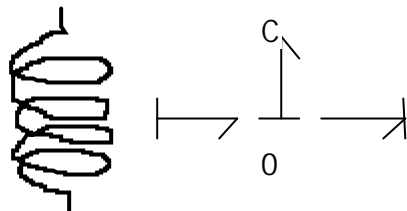
1. The classic bond graph sketch of the system looks nothing like the physical system and nothing like the system in terms of international symbols. A simulation demonstration of the model to a potential client displayed in this way is too abstract to be successful. It is necessary to introduce a 'shrink-wrap' facility to map groups of bond graph elements onto more technological icons. This is provided by some bond graph software.
2. The training necessary to become proficient in the use of bond graphs is extensive. It is too far from the technology to be comprehensible by most design office staff.
3. It does not cope well with complex phenomena such as air release and cavitation.
4. It does not cope well with some modeling techniques such as discontinuities.

Clearly the bond graph approach is an answer to the diversity problem but has too many other disadvantages. However, the approach suggests that a reticulation of the system into much smaller units has advantages. What is needed is a combination of the advantages of the bond graph approach with a much more technological representation of the system.

## 7. The basic element approach

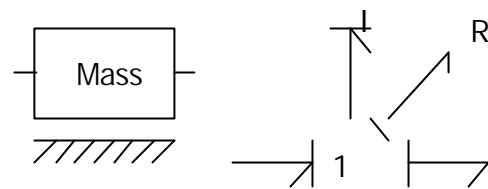
If we think of an engineering system and look for a natural division into the smallest possible technological elements, we end up with elements typified by a mechanical spring, an inertia (possibly with friction), a hydraulic orifice, etc. These will be called **basic elements**. We seek to create icons representing these elements and one or

more submodel for each icon. In Figure 2 and Figure 3, two such icons are compared with their bond graph equivalent.



Mechanical spring

Figure 2



Inertia with friction

Figure 3

It is interesting to note that each basic element icon is equivalent to from 1 to 4 bond graph elements. We will see later that the typically standard multiport technological icon can be constructed from 4 to 12 basic elements.

This suggests the following hierarchy:

- engineer
- technological element
- basic element
- bond graph element
- physics
- mathematical formulation
- code

The lowest level is the code which is furthest from the engineer. We see that the engineer is closer to the basic element level than to the bond graph. This makes expression of the system in terms of basic elements comprehensible by a much wider group of engineers and technologists.

With the AMESim software [8] this idea was pursued with the developed of a small prototype collection of basic elements to build hydraulic components. This is shown in Figure 4.

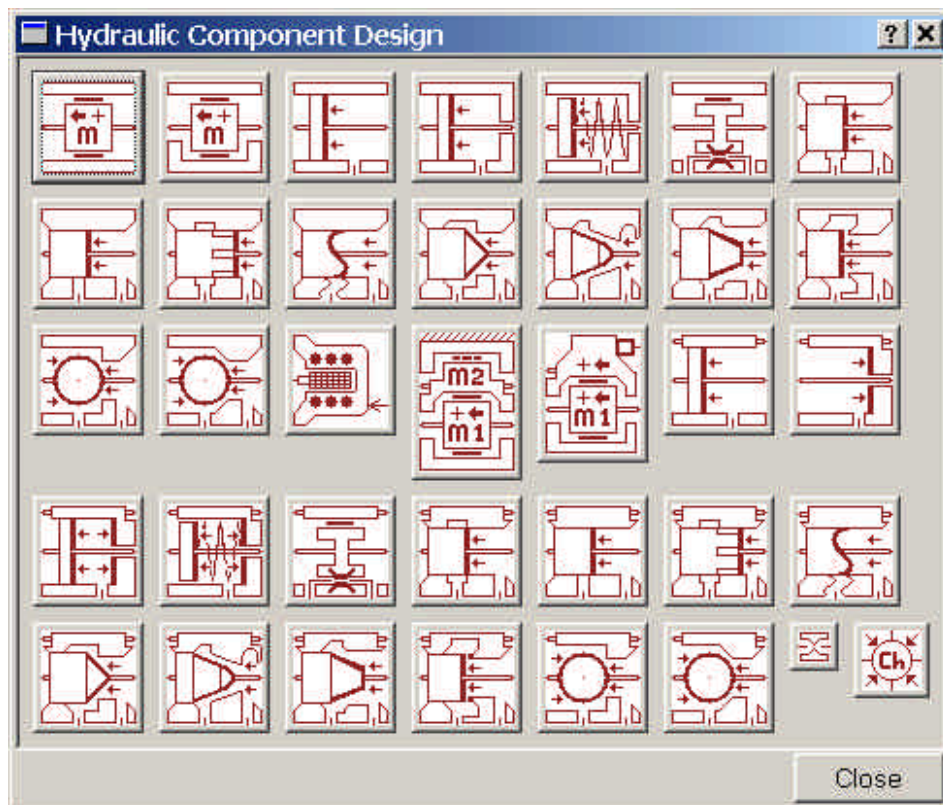


Figure 4

Taking the hydraulic jack as an example suppose we require a 2 hydraulic chamber, 2 rod hydraulic jack with 2 springs and a fixed body. Suppose further that it must be possible to attach a load from one rod only. Figure 5 shows the basic elements necessary to create this and Figure 6 shows the completed jack.

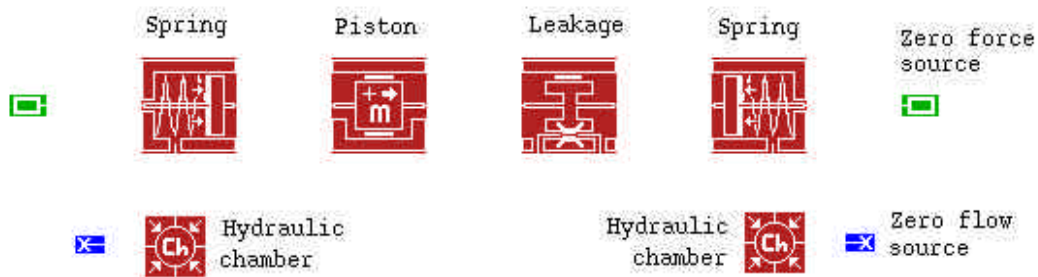


Figure 5

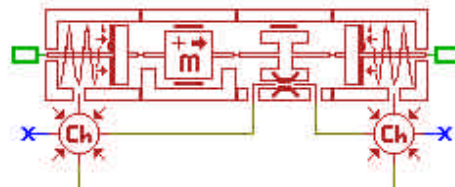


Figure 6 : an hydraulic jack

If required, this collection of icons can be shrink wrapped onto the more standard icon shown in Figure 7.

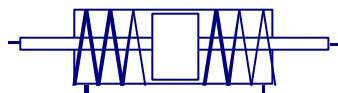


Figure 7 : jack supercomponent

This small library proved far more successful and adaptable than had been expected. Figure 8 shows an arrangement of icons needed to build a 4-way valve. Figure 9 shows the completed valve. Further examples are presented in section 8.



Figure 8

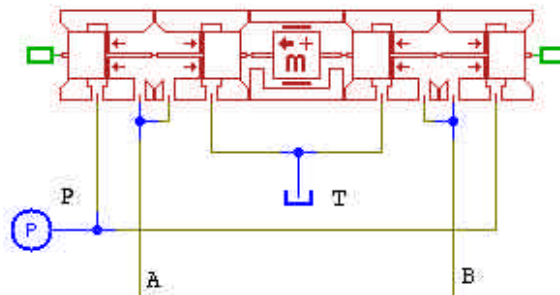


Figure 9 : 4 way valve

An important prerequisite of the basic element library is the creation of extremely well-tested, reliable and reusable submodels that a user can employ with complete confidence. The writer of the basic element library must be competent in all the modeling skills. However, the user of the basic element library is relieved of the need to write code and formulate the mathematics. Understanding of the details of the physics is not needed but decisions on assumption are necessary which imply some knowledge of physics. Understanding of the engineering system and an ability to interpret results is still important.

Experience in training design office staff to use of the basic element library suggests that it is learnt very rapidly. Generally speaking within one hour the library can be employed effectively.

## 8. Examples

This section presents three examples of the use of an extended basic element library used for industrial applications. To highlight the basic elements, the figure does not employ the shrink-wrap facility.

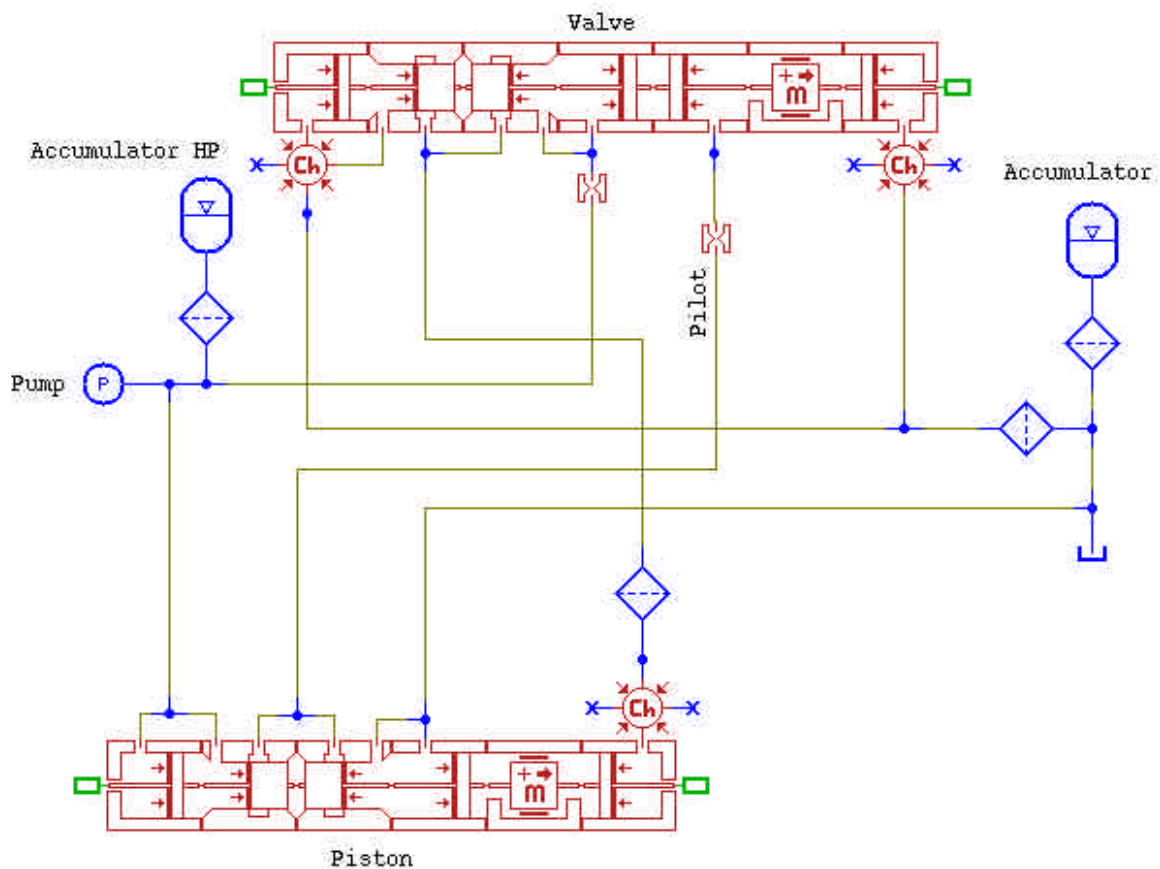


Figure 10 : Hydraulic Hammer

Figure 10 shows part of a hydraulic hammer with the hammer and control valve built from the basic element library.

Figure 11 shows part of an automatic gearbox. Components such as pressure valves, special accumulators and clutches are modeled with the Hydraulic Component Design library and the Mechanical library.

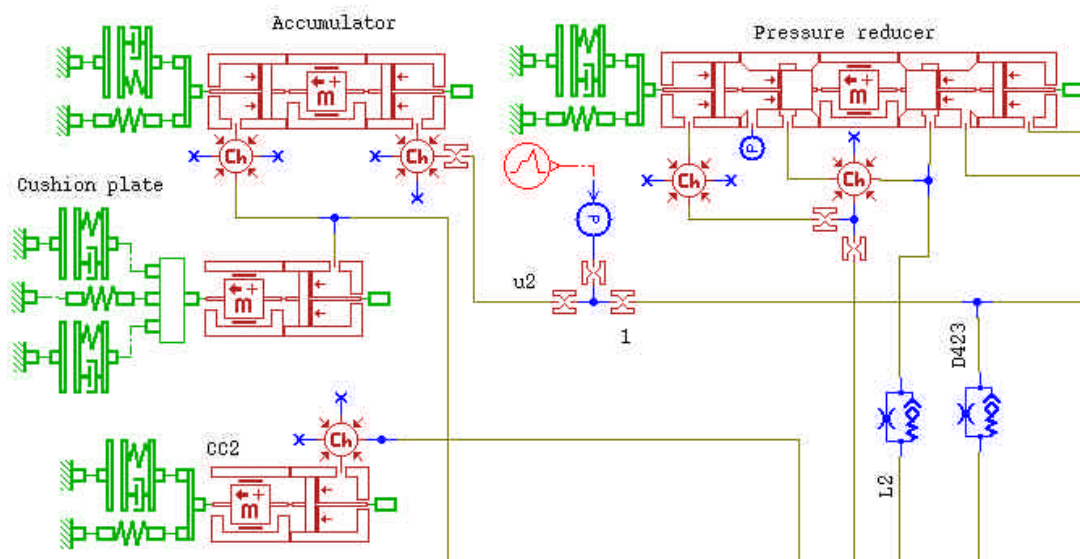


Figure 11 : automatic gearbox

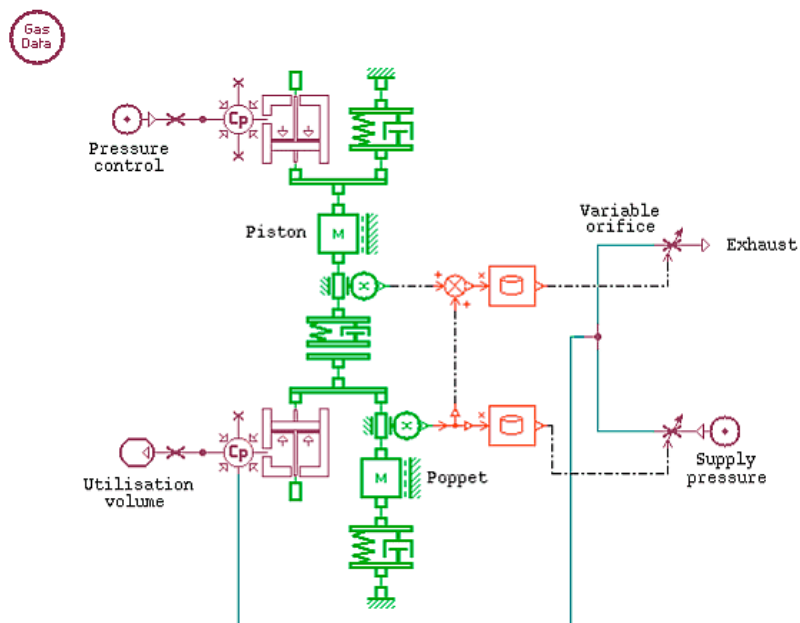


Figure 12 : Pneumatic Pressure Relay

Figure 12 shows part of a brake system. Here the pressure relay valve is modeled using the Pneumatic library and the Mechanical library.

The library is being progressively extended and other examples of applications are:

- fuel injection systems
- variable valve actuation
- lift systems
- aircraft air circuits
- aircraft fuel systems

Other libraries using the same approach are now available:

- Hydraulic Resistance,
- Network Filling,
- Thermal,
- Thermal-hydraulic,
- Thermal-pneumatic,
- Powertrain.

## 9. Numerical Aspects

One of the skills listed in section 2 was the ability to understand the interaction between the model and the numerical algorithms used to perform the simulation. Limiting the discussion to simulation where the model is either a system of

Ordinary differential equations or  
Differential algebraic equations,

with some software the user is presented with a menu of integration methods and is expected to select an appropriate method. This is completely unnecessary since algorithms exist for switching automatically between algorithms [17]. If algorithms of this type are used, consideration of numerical aspects is largely removed from the modeling process.

## 10. Conclusions

This paper has explored ways of eliminating the need for writing code in producing good models of engineering systems. The multiport approach has been used but with a special reticulation based on very small technological elements. This approach is coupled with a shrink-wrap facility, which enables groups of basic elements to be combined into a single icon.

The system developed has gone a long way towards freeing the engineer from the engineering to concentrate more on the engineering aspects of simulation.

## References

- [1] J. Habenstreit. 'Simulation et pédagogie: une rencontre du troisième type', AFCET Interfaces, March 1988.
- [2] T.J.A. De Vries. 'Conceptual Design of Controlled Electro-mechanical Systems, A Modeling Perspective', Thesis 1993, Enschede, The Netherlands.
- [3] D.G. Ullman. 'The Mechanical Design Process', McGraw Hill, New York, 1992.
- [4] OLMECO Esprit 3 Project, lead by PSA (Peugeot - Citroen).
- [5] Simulink, The MathWorks, Inc., 24 Prime Park Way, Natick, Mass. 01760, U.S.A.
- [6] XMath, Integrated Systems, Inc., 3260 Jay St., Santa Clara, CA 95054-3309, U.S.A.
- [7] Easy5, Boeing, P.O. Box 24346, M/S 7L-23, Seattle, WA 98124-0346, U.S.A.
- [8] AMESim, Imagine S.A., 5 rue Brison, 42300 Roanne, France.
- [9] Saber, Analogy Inc., 9205 SW Gemini Drive, Beaverton, Oregon 97005-7156, U.S.A.
- [10] ITI-Sim, ITI GmbH, Gostritzer Strasse 63, D-01217 Dresden, Germany.
- [11] Flowmaster, Amazon Computers Ltd., Sunrise Parkway, Linford Wood, Milton Keynes MK14 6LQ, U.K.
- [12] H.M. Paynter. 'Analysis and Design of Engineering Systems', MIT Press, Cambridge, 1961.

- [13]** D.C. Karnopp, D.L. Margolis, R.C. Rosenberg. 'System Dynamics: A Unified Approach', John Wiley & Sons, New York, 1990.
  
- [14]** Enport, Rosencode Association, Inc., 2216 E. Michigan Ave., Lansing, MI 48912, U.S.A.
  
- [15]** MS1, Lorenz Consulting S.A., Centre Socran, Parc Scientifique, Ave. Pré-Aily, B-4031 Angleur (Liege), Belgium.
  
- [16]** 20Sim, Controllab Products Inc., P.O. Box 217, 7500 AE Enschede, The Netherlands.
  
- [17]** C.W. Richards et al, 'A Type Insensitive Integrator for Fluid Power Systems,' ASME WAM, Dalls, U.S.A., November 1990.

This paper was presented by Michel Lebrun and Claude William Richards at the Fifth Scandinavian International Conference on Fluid Power at Linköping, Sweden on the 28th May 1997.