Control of an hydraulic press

Jean-Marc ROUSSEL
jean-marc.roussel@lurpa.ens-cachan.fr

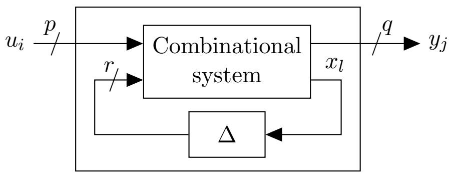
LURPA, ENS Cachan
61, avenue du Prsident Wilson
94235 CACHAN Cedex

October 16, 2013

1 Introduction

This document presents a case study made according to the algebraic synthesis method developed in LURPA. We propose to obtain the control law to implement into a Programmable Logic Controller (PLC) from its specifications given in natural language, by solving a Boolean equations system of switching functions.

We suppose that the expected control law can be expressed with recurrent Boolean equations as presented Figure 1. This generic model has p Boolean inputs (u_i), q Boolean outputs (y_j) and r Boolean state variables (x_l). These inputs and outputs correspond to the inputs and outputs of the controller for which the control laws must be designed. The state variables, used to express the sequential behavior, will be represented with internal variables of the controller.



$$\begin{cases} y_j[k] = F_j(u_1[k], \cdots, u_p[k], x_1[k-1], \cdots, x_r[k-1]) \\ x_l[k] = F_{q+l}(u_1[k], \cdots, u_p[k], x_1[k-1], \cdots, x_r[k-1]) \end{cases}$$

Figure 1: Generic model of sequential systems expressed with recurrent Boolean equations

The behavior of this model can be fully defined according to the definition of $(q+r)$ switching functions of $(p+r)$ variables. Even if this representation is very compact (the r Boolean state variables allow the representation of 2^r different states), the construction by hands of these switching functions has always been a very tedious and error-prone task [Huf54]: the model presented Figure. 1, admits 2^p inputs combinations, can send 2^q outputs combinations and can express $(2^{2^{(p+r)}})^{(q+r)}$ sequential behaviors.

Nevertheless, thanks to recent mathematical results obtained for Boolean algebras [Rud01], [Bro03], the automatic algebraic synthesis of switching functions is now possible. We propose to obtain the control law to implement into a PLC by solving a Boolean equations system of switching functions. Details of the proposed method can be found in [HRL08a] [HRL08b] [Hie09].

To avoid tedious symbolic calculus and to help the designer during the different steps of this synthesis method, a prototype software tool has been developed in Python. This tool[1] performs all the computations required for inconsistencies detection and control laws generation. This enables the designer to focus only on application-related issues. For ergonomic reasons, complementary works were also developed in order be able to represent the synthesized control law with a state model.

[1]Case studies are available: http://www.lurpa.ens-cachan.fr/-226050.kjsp

1.1 Notations used

To avoid confusion between Boolean variables and Boolean functions of Boolean variables, each Boolean variable b_i is denoted as ${}_bb_i$. The set of the two Boolean values ${}_b0$ and ${}_b1$ is denoted as: $B = \{{}_b0, {}_b1\}$. The classical two-element Boolean Algebra is $(\{{}_b0, {}_b1\}, \vee, \wedge, \neg, {}_b0, {}_b1)$.

Let $F_n(B)$ be the set of the 2^{2^n} n-variable switching functions. The Boolean Algebra of n-variable switching functions is $(F_n(B), +, \cdot, \bar{}, 0, 1)$:

- 0 and 1 are the 2 constant functions:

$$0: \begin{array}{rl} B^n & \to B \\ ({}_bb_1, \ldots, {}_bb_n) & \mapsto {}_b0 \end{array} \qquad 1: \begin{array}{rl} B^n & \to B \\ ({}_bb_1, \ldots, {}_bb_n) & \mapsto {}_b1 \end{array}$$

- $+$, $\cdot$, $\bar{}$ are three closed operations defined as follows:

$$\text{Op. } +: \begin{array}{rl} F_n(B)^2 & \to F_n(B) \\ (f,g) & \mapsto f+g \end{array} \qquad \text{Op. } \cdot: \begin{array}{rl} F_n(B)^2 & \to F_n(B) \\ (f,g) & \mapsto f \cdot g \end{array} \qquad \text{Op. } \bar{}: \begin{array}{rl} F_n(B) & \to F_n(B) \\ f & \mapsto \overline{f} \end{array}$$

$$\begin{array}{l} \text{where } \forall ({}_bb_1, \ldots, {}_bb_n) \in B^n, \\ (f+g)({}_bb_1, \ldots, {}_bb_n) = f({}_bb_1, \ldots, {}_bb_n) \vee g({}_bb_1, \ldots, {}_bb_n) \\ (f \cdot g)({}_bb_1, \ldots, {}_bb_n) = f({}_bb_1, \ldots, {}_bb_n) \wedge g({}_bb_1, \ldots, {}_bb_n) \\ \overline{f}({}_bb_1, \ldots, {}_bb_n) = \neg f({}_bb_1, \ldots, {}_bb_n) \end{array}$$

$F_n(B)$ can be equipped with a partial order relation, called *Inclusion-Relation* defined as follows:

$$x \leq y \quad \Leftrightarrow \quad x \cdot y = x$$

2 Control system specifications

Let us consider a hydraulic press with a vertical ram (Fig 2). A safety-light curtain is used to safeguard operators during the movements of the ram. A control panel allows to select the mode of operation: Manual or Automatic mode.

- In Manual mode, all the operations are carried out by pressing the corresponding push-buttons. As soon as a push-button is released, the ram movement stops.
- In Automatic mode, the cycle starts by pressing the 'Start' push-button: the ram is going down and comes back to the up position after the press operation has been done.

Figure 2: A hydraulic press and its human-machine interfaces

2.1 Inputs and outputs of the controller

The Boolean inputs and outputs of this controller are given in Fig. 3. Each movement of the ram is controlled thanks to a Boolean output ('goUp' and 'goDown'). The controller is informed of the position of the ram thanks to inputs 'up' and 'down'. The safety light curtain is connected to input 'lc' ($\mathrm{lc} =_b 1$ when the operator is not in the detection zone of the light curtain). The control panel of the press is composed of an emergency stop button (input: 'esb'), a three position center-off switch for the operating mode selection ('sbA', off: no mode selected, 'sbM') and four push-buttons (inputs: 'vpb', 'spb', 'uppb' and 'dopb').

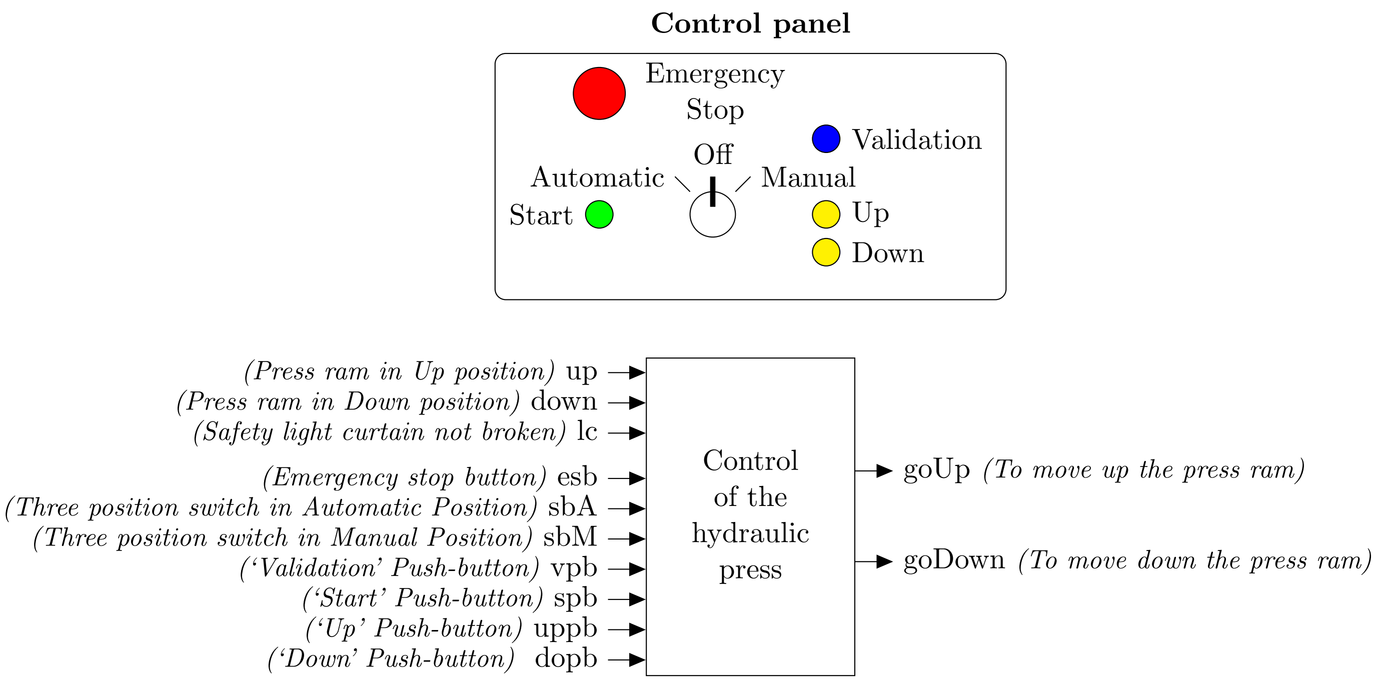


Figure 3: Inputs and outputs of the controller to design

2.2 Expected behavior

The expected behavior proposed for the synthesis of the operation modes of the control system regarding the application requirements can be expressed by the set of assertions given hereafter:

- **R1** The three modes (Automatic, Manual and Fail) are exclusive.
- **R2** While the Emergency stop button 'esb' is pressed, the press is in Failure mode.
- **R3** If the observed position of the ram is both 'up' and 'down', the press is in Failure mode.
- **R4** For leaving the Failure mode, the operator must not be in the detection zone of the light curtain.
- **R5** The press is in Automatic mode if and only if the three position center-off switch is turned on 'sbA' position.
- **R6** For reaching the Automatic mode, the press ram must be in 'up' position and the operator must not be in the detection zone of the light curtain.
- **R7** For reaching the Automatic mode, the 'vpb' push-button must be pressed.
- **R8** During the Automatic mode, the operator can be in the detection zone of the light curtain without to be in danger only if the press ram is in 'up' position.
- **R9** During the Automatic mode, if the operator is detected by the light curtain while the press ram is not in 'up' position, one has to switch into the Failure mode.
- **R10** The press is in Manual mode if and only if the three position center-off switch is turned on 'sbM' position.
- **R11** For reaching the Manual mode, the press ram must be in 'up' position and the operator must not be in the detection zone of the light curtain.
- **R12** For reaching the Manual mode, the 'vpb' push-button must be pressed.

2.3 Control laws to design

It is not possible to identify automatically how many and which state variables must be used. The designer has to fix the state variables by expertise.

For this case study, we propose to use 5 state variables: one for each output; one for each mode of operation (Automatic, Manual)[2] and one for characterizing a state where the press is in a failure mode (Fail). According to this choice, we have only 5 15-switching functions to synthesize. However, each function is a 15-variable switching functions as the control laws has 10 inputs and 5 state variables. The generic form of the control law we want to design is:

$$\begin{cases} \mathrm{auto}[k] = \mathrm{Auto}(\mathrm{up}[k], \dots, \mathrm{dopb}[k], \mathrm{auto}[k-1], \mathrm{manual}[k-1], \mathrm{fail}[k-1], \mathrm{goUp}[k-1], \mathrm{goDown}[k-1]) \\ \mathrm{manual}[k] = \mathrm{Manual}(\mathrm{up}[k], \dots, \mathrm{dopb}[k], \mathrm{auto}[k-1], \mathrm{manual}[k-1], \mathrm{fail}[k-1], \mathrm{goUp}[k-1], \mathrm{goDown}[k-1]) \\ \mathrm{fail}[k] = \mathrm{Fail}(\mathrm{up}[k], \dots, \mathrm{dopb}[k], \mathrm{auto}[k-1], \mathrm{manual}[k-1], \mathrm{fail}[k-1], \mathrm{goUp}[k-1], \mathrm{goDown}[k-1]) \\ \mathrm{goUp}[k] = \mathrm{GoUp}(\mathrm{up}[k], \dots, \mathrm{dopb}[k], \mathrm{auto}[k-1], \mathrm{manual}[k-1], \mathrm{fail}[k-1], \mathrm{goUp}[k-1], \mathrm{goDown}[k-1]) \\ \mathrm{goDown}[k] = \mathrm{GoDown}(\mathrm{up}[k], \dots, \mathrm{dopb}[k], \mathrm{auto}[k-1], \mathrm{manual}[k-1], \mathrm{fail}[k-1], \mathrm{goUp}[k-1], \mathrm{goDown}[k-1]) \end{cases}$$

This model permits to express $(2^{2^{15}})^5$ different control laws. We propose to find the control law which satisfies the expected behavior given Section 2.2 by solving a Boolean equations system of 15-variable switching functions.

[2] These mode variables must not be confused with inputs 'sbA' or 'sbM' which are the demand of operator for reaching these modes.

3 Algebraic synthesis of the control laws

The first step of the proposed method consists to formalize the expected behavior with relations between formula of 15-variable switching functions. For this case study, we have 15 specific switching functions[3]:

- The 10 switching functions (Up, Down, Lc, Esb, SbA, SbM, Vpb, Spb, Uppb, and Dopb) which characterize the behavior of the inputs of the controller. They are defined as follows:

$$\begin{aligned} \text{Up}: \quad & B^{15} \to B \\ & (\text{up}[k], \ldots, \text{goDown}[k-1]) \mapsto \text{up}[k] \end{aligned}$$

- The 5 switching functions (${}_p\text{Auto}$, ${}_p\text{Manual}$, ${}_p\text{Fail}$, ${}_p\text{GoUp}$ and ${}_p\text{GoDown}$) which characterize the previous behavior of the state variables of the controller. They are defined as follows:

$$\begin{aligned} {}_p\text{Auto}: \quad & B^{15} \to B \\ & (\text{up}[k], \ldots, \text{goDown}[k-1]) \mapsto \text{auto}[k-1] \end{aligned}$$

In our case, only 5 switching functions must be designed (Auto, Manual, Fail, GoUp and GoDown). They represent the unknowns or our problem.

Remark: As GoDown and ${}_p$GoDown represent the behavior of 'goDown' at respectively times $[k]$ and $[k-1]$, the starting of the going down of the ram corresponds to $(\text{GoDown} \cdot \overline{{}_p\text{GoDown}})$ and the stopping of the going down of the ram corresponds to $(\overline{\text{GoDown}} \cdot {}_p\text{GoDown})$.

3.1 Formalization of requirements

In order to illustrate the power of expression of relations Equality and Inclusion, all the requirements used for the synthesis of the operation modes are expressed in textual form hereafter:

- **R1** The three modes (Automatic, Manual and Fail) are exclusive.

$$\text{Auto} \cdot \text{Manual} + \text{Auto} \cdot \text{Fail} + \text{Manual} \cdot \text{Fail} = 0$$

- **R2** While the Emergency stop button 'esb' is pressed, the press is in Failure mode.

$$\text{Esb} \leq \text{Fail}$$

- **R3** If the observed position of the ram is both 'up' and 'down', the press is in Failure mode.

$$\text{Up} \cdot \text{Down} \leq \text{Fail}$$

- **R4** For leaving the Failure mode, the operator must not be in the detection zone of the light curtain.

$$\overline{\text{Fail}} \cdot {}_p\text{Fail} \leq \text{Lc}$$

- **R5** The press is in Automatic mode if and only if the three position center-off switch is turned on 'sbA' position.

$$\text{Auto} = \text{SbA}$$

- **R6** For reaching the Automatic mode, the press ram must be in 'up' position and the operator must not be in the detection zone of the light curtain.

$$\text{Auto} \cdot \overline{{}_p\text{Auto}} \leq \text{Up} \cdot \text{Lc}$$

[3]These functions are the 15 projection-functions of $F_{15}(B)$.

- **R7** For reaching the Automatic mode, the 'vpb' push-button must be pressed.

$$\text{Auto} \cdot \overline{{}_p\text{Auto}} \leq \text{Vpb}$$

- **R8** During the Automatic mode, the operator can be in the detection zone of the light curtain without to be in danger only if the press ram is in 'up' position.

$$\text{Auto} \cdot \overline{\text{Lc}} \leq \text{Up}$$

- **R9** During the Automatic mode, if the operator is detected by the light curtain while the press ram is not in 'up' position, one has to switch into the Failure mode.

$$\overline{\text{Auto}} \cdot {}_p\text{Auto} \cdot \overline{(\text{Lc} + \text{Up})} \leq \text{Fail}$$

- **R10** The press is in Manual mode if and only if the three position center-off switch is turned on 'sbM' position.

$$\text{Manual} = \text{SbM}$$

- **R11** For reaching the Manual mode, the press ram must be in 'up' position and the operator must not be in the detection zone of the light curtain.

$$\text{Manual} \cdot \overline{{}_p\text{Manual}} \leq \text{Lc}$$

- **R12** For reaching the Manual mode, the 'vpb' push-button must be pressed.

$$\text{Manual} \cdot \overline{{}_p\text{Manual}} \leq \text{Vpb}$$

3.2 Synthesis process

3.2.1 Synthesis of the operation modes

For this case study, we have started with the synthesis of operation modes. The first four studied requirements have been: R1, R2, R5 and R10:

$$\begin{cases} \textbf{R1} & \text{Auto} \cdot \text{Manual} + \text{Auto} \cdot \text{Fail} + \text{Manual} \cdot \text{Fail} = 0 \\ \textbf{R2} & \text{Esb} \leq \text{Fail} \\ \textbf{R5} & \text{Auto} = \text{SbA} \\ \textbf{R10} & \text{Manual} = \text{SbM} \end{cases}$$

For this subset of requirements, the result given by our software tool was the following inconsistency condition:

$$\mathcal{I}_0 = \text{SbA} \cdot \text{SbM} + \text{Esb} \cdot \text{SbA} + \text{Esb} \cdot \text{SbM}$$

Since requirements are declared as inconsistent, we have to give complementary information to precise our specification. By analyzing each term of this formula, it is possible to detect the origin of the inconsistency:

- $\text{SbA} \cdot \text{SbM}$: What happens if the Automatic mode and the Manual mode are simultaneously selected? We consider that it is not possible (due to the technology of the three position center-off switch) and we have added Assumption A1.

- $\text{Esb} \cdot \text{SbA}$: What happens if the Emergency stop button is activated during the Automatic mode? We consider that the Fail mode has priority on Automatic mode (for security reasons) and we have added the priority rule: $\{\text{R1,R2}\} \gg \text{R5}$.

- $\text{Esb} \cdot \text{SbM}$: What happens if the Emergency stop button is activated during the Manual mode? We consider that the Fail mode has priority on Manual mode (for security reasons) and we have added the priority rule: $\{\text{R1,R2}\} \gg \text{R10}$.

With this complementary information, Problem (1) admits as parametric solution (2):

$$\left[\begin{array}{l}
\text{Requirements:}\\
\quad\begin{cases}
\text{R1} & \text{Auto}\cdot\text{Manual}+\text{Auto}\cdot\text{Fail}+\text{Manual}\cdot\text{Fail}=0\\
\text{R2} & \text{Esb}\leq\text{Fail}\\
\text{R5} & \text{Auto}=\text{SbA}\\
\text{R10} & \text{Manual}=\text{SbM}
\end{cases}\\
\text{Priority rules:}\\
\quad\begin{cases}
\{\text{R1,R2}\}\gg\text{R5}\\
\{\text{R1,R2}\}\gg\text{R10}
\end{cases}\\
\text{Assumptions:}\\
\quad\begin{cases}
\text{A1} \quad \text{SbA}\cdot\text{SbM}=0
\end{cases}
\end{array}\right. \tag{1}$$

$$\begin{cases}
\text{Fail}=\text{Esb}+p_1\cdot\overline{\text{SbA}}\cdot\overline{\text{SbA}} & p_1\in F_{15}(B)\\
\text{Auto}=\overline{\text{Esb}}\cdot\text{SbA}\\
\text{Manual}=\overline{\text{Esb}}\cdot\text{SbM}
\end{cases} \tag{2}$$

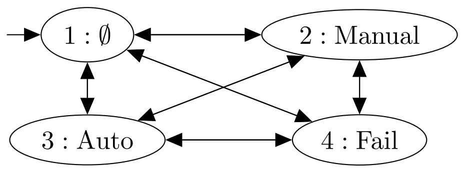
By gradually adding all the requirements and selecting the solution which minimizes the Fail mode (the parameter p_1 is fixed to 0), the complete specification of the operation modes is:

$$\left[\begin{array}{l}
\textbf{Requirements:}\\
\quad\begin{cases}
\textbf{R1} & \text{Auto}\cdot\text{Manual}+\text{Auto}\cdot\text{Fail}+\text{Manual}\cdot\text{Fail}=0\\
\textbf{R2} & \text{Esb}\leq\text{Fail}\\
\textbf{R3} & \text{Up}\cdot\text{Down}\leq\text{Fail}\\
\textbf{R4} & \overline{\text{Fail}}\cdot{}_p\text{Fail}\leq\text{Lc}\\
\textbf{R5} & \text{Auto}=\text{SbA}\\
\textbf{R6} & \text{Auto}\cdot\overline{{}_p\text{Auto}}\leq\text{Up}\cdot\text{Lc}\\
\textbf{R7} & \text{Auto}\cdot\overline{{}_p\text{Auto}}\leq\text{Vpb}\\
\textbf{R8} & \text{Auto}\cdot\overline{\text{Lc}}\leq\text{Up}\\
\textbf{R9} & \overline{\text{Auto}}\cdot{}_p\text{Auto}\cdot\overline{(\text{Lc}+\text{Up})}\leq\text{Fail}\\
\textbf{R10} & \text{Manual}=\text{SbM}\\
\textbf{R11} & \text{Manual}\cdot\overline{{}_p\text{Manual}}\leq\text{Lc}\\
\textbf{R12} & \text{Manual}\cdot\overline{{}_p\text{Manual}}\leq\text{Vpb}
\end{cases}\\
\textbf{Priority rules:}\\
\quad\begin{cases}
\{\text{R1,R2,R3,R4}\}\gg\text{R5} & \textit{(* Fail mode has priority on Automatic mode. *)}\\
\{\text{R6,R7,R8}\}\gg\text{R5} & \textit{(* Starting rules have priority on the switch position rule. *)}\\
\{\text{R1,R2,R3,R4,R8}\}\gg\text{R10} & \textit{(* Fail mode has priority on Manual mode. *)}\\
\{\text{R11,R12}\}\gg\text{R10} & \textit{(* Starting rules have priority on the switch position rule. *)}
\end{cases}\\
\textbf{Assumptions:}\\
\quad\begin{cases}
\textbf{A1} \quad \text{SbA}\cdot\text{SbM}=0 \quad \textit{(* The 2 positions of the switch button are exclusive. *)}\\
\textbf{A2} \quad {}_p\text{Auto}\cdot{}_p\text{Manual}+{}_p\text{Auto}\cdot{}_p\text{Fail}+{}_p\text{Manual}\cdot{}_p\text{Fail}=0 \quad \textit{(* Consequence of R1. *)}
\end{cases}\\
\textbf{Optimization criteria:}\\
\quad\text{Minimisation of: Fail}
\end{array}\right.$$

The solution we obtain for operation modes is:

$$
\begin{cases}
\mathrm{Fail} = \mathrm{Esb} + \mathrm{Up} \cdot \mathrm{Down} + \overline{\mathrm{Lc}} \cdot {}_{p}\mathrm{Fail} + \overline{\mathrm{Up}} \cdot \overline{\mathrm{Lc}} \cdot {}_{p}\mathrm{Auto} \\
\mathrm{Auto} = \overline{\mathrm{Esb}} \cdot \mathrm{SbA} \cdot (\mathrm{Up} \cdot \overline{\mathrm{Down}} \cdot \mathrm{Lc} \cdot \mathrm{Vpb} + {}_{p}\mathrm{Auto} \cdot (\mathrm{Up} \cdot \overline{\mathrm{Down}} + \overline{\mathrm{Up}} \cdot \mathrm{Lc})) \\
\mathrm{Manual} = \overline{\mathrm{Esb}} \cdot \mathrm{SbM} \cdot \overline{(\mathrm{Up} \cdot \mathrm{Down})} \cdot (\mathrm{Lc} \cdot \mathrm{Vpb} + {}_{p}\mathrm{Manual})
\end{cases}
$$

In an illustrative purpose, a state model representation of the synthesized operation modes management, automatically built thanks to [Gui11], is given in (Fig. 4). The transition conditions are non-trivial Boolean expressions of inputs that take into account the whole set of specifications.



$$
\begin{cases}
E_{1-2} = \neg\mathrm{esb} \wedge \mathrm{lc} \wedge \mathrm{sbM} \wedge \mathrm{vpb} \wedge (\neg\mathrm{up} \vee \neg\mathrm{down}) \\
E_{1-3} = \neg\mathrm{esb} \wedge \mathrm{lc} \wedge \mathrm{sbA} \wedge \mathrm{vpb} \wedge \mathrm{up} \wedge \neg\mathrm{down} \\
E_{1-4} = \mathrm{esb} \vee \mathrm{up} \wedge \mathrm{down} \\
E_{2-1} = \neg\mathrm{esb} \wedge \neg\mathrm{sbM} \wedge (\neg\mathrm{up} \vee \neg\mathrm{down} \wedge \neg(\mathrm{lc} \wedge \mathrm{sbA} \wedge \mathrm{vpb})) \\
E_{2-3} = \neg\mathrm{esb} \wedge \mathrm{lc} \wedge \mathrm{sbA} \wedge \mathrm{vpb} \wedge \mathrm{up} \wedge \neg\mathrm{down} \\
E_{2-4} = \mathrm{esb} \vee \mathrm{up} \wedge \mathrm{down} \\
E_{3-1} = \neg\mathrm{esb} \wedge \neg\mathrm{sbA} \wedge (\neg\mathrm{lc} \wedge \mathrm{up} \wedge \neg\mathrm{down} \vee \\
\qquad\qquad (\neg\mathrm{sbM} \vee \neg\mathrm{vpb}) \wedge (\mathrm{up} \wedge \neg\mathrm{down} \vee \mathrm{lc} \wedge \neg\mathrm{up})) \\
E_{3-2} = \neg\mathrm{esb} \wedge \mathrm{lc} \wedge \mathrm{sbM} \wedge \mathrm{vpb} \wedge (\neg\mathrm{up} \vee \neg\mathrm{down}) \\
E_{3-4} = \mathrm{esb} \vee \mathrm{up} \wedge \mathrm{down} \vee \neg\mathrm{up} \wedge \neg\mathrm{lc} \\
E_{4-1} = \neg\mathrm{esb} \wedge \mathrm{lc} \wedge (\neg\mathrm{up} \vee \neg\mathrm{down}) \wedge \\
\qquad (\neg\mathrm{sbA} \wedge \neg\mathrm{sbM} \vee \neg\mathrm{vpb} \vee \mathrm{sbA} \wedge \neg\mathrm{up}) \\
E_{4-2} = \neg\mathrm{esb} \wedge \mathrm{lc} \wedge \mathrm{sbM} \wedge \mathrm{vpb} \wedge (\neg\mathrm{up} \vee \neg\mathrm{down}) \\
E_{4-3} = \neg\mathrm{esb} \wedge \mathrm{lc} \wedge \mathrm{sbA} \wedge \mathrm{vpb} \wedge \mathrm{up} \wedge \neg\mathrm{down}
\end{cases}
$$

Figure 4: State model of modes of operation

3.2.2 Synthesis of the reactive control laws

To obtain the complete control law of the hydraulic press, the solution previously obtained for the operation modes has to be completed thanks to requirements R20 to R31.

Requirements:

$$
\begin{cases}
\textbf{R20} & \mathrm{GoUp}\cdot\mathrm{GoDown}=0\\
\textbf{R21} & \mathrm{Up}\le\overline{\mathrm{GoUp}}\\
\textbf{R22} & \mathrm{Down}\le\overline{\mathrm{GoDown}}\\
\textbf{R23} & (\mathrm{GoUp}+\mathrm{GoDown})\le\mathrm{Lc}\\
\textbf{R24} & (\mathrm{GoUp}+\mathrm{GoDown})\le(\mathrm{Auto}+\mathrm{Manual})\\
\textbf{R25} & \mathrm{Manual}\le((\mathrm{GoUp}\cdot\mathrm{Uppb})+(\overline{\mathrm{GoUp}}\cdot\overline{\mathrm{Uppb}}))\\
\textbf{R26} & \mathrm{Manual}\le((\mathrm{GoDown}\cdot\mathrm{Dopb})+(\overline{\mathrm{GoDown}}\cdot\overline{\mathrm{Dopb}}))\\
\textbf{R27} & \mathrm{Manual}\cdot(\mathrm{Uppb}\cdot\mathrm{Dopb})\le\overline{(\mathrm{GoUp}+\mathrm{GoDown})}\\
\textbf{R28} & \mathrm{Auto}\cdot(\overline{\mathrm{GoUp}}\cdot{}_p\mathrm{GoUp})\le\mathrm{Up}\\
\textbf{R29} & \mathrm{Auto}\cdot(\overline{\mathrm{GoDown}}\cdot{}_p\mathrm{GoDown})\le\mathrm{Down}\\
\textbf{R30} & \mathrm{Auto}\le(\mathrm{Up}+\mathrm{GoUp}+\mathrm{GoDown})\\
\textbf{R31} & \mathrm{Auto}\le((\mathrm{GoDown}\cdot\overline{{}_p\mathrm{GoDown}})\cdot(\mathrm{Up}\cdot\mathrm{Spb})+\overline{(\mathrm{GoDown}\cdot\overline{{}_p\mathrm{GoDown}})}\cdot\overline{(\overline{{}_p\mathrm{GoDown}}\cdot\mathrm{Up}\cdot\mathrm{Spb})})
\end{cases}
$$

Priority rules:

$$
\begin{cases}
\{\mathrm{R21},\mathrm{R23},\mathrm{R27}\}\gg\mathrm{R25} & \textit{(* Safety requirements have priority on functional requirements. *)}\\
\{\mathrm{R21},\mathrm{R23},\mathrm{R27}\}\gg\mathrm{R26} & \textit{(* Safety requirements have priority on functional requirements. *)}\\
\mathrm{R23}\gg\mathrm{R28} & \textit{(* Safety requirements have priority on functional requirements. *)}\\
\mathrm{R23}\gg\mathrm{R29} & \textit{(* Safety requirements have priority on functional requirements. *)}\\
\mathrm{R23}\gg\mathrm{R30} & \textit{(* Safety requirements have priority on functional requirements. *)}\\
\{\mathrm{R22},\mathrm{R23}\}\gg\mathrm{R31} & \textit{(* Safety requirements have priority on functional requirements. *)}
\end{cases}
$$

Assumptions:

A3 ${}_p\mathrm{GoUp}\cdot{}_p\mathrm{GoDown}=0$ *(* Consequence of R20. *)*

3.3 Equation solving

Thanks to the last mathematical results, we are able to obtain automatically the solution of this problem. This solution is:

$$
\begin{cases}
\mathrm{Fail} & =\mathrm{Esb}+\mathrm{Up}\cdot\mathrm{Down}+\overline{\mathrm{Lc}}\cdot{}_p\mathrm{Fail}+\overline{\mathrm{Up}}\cdot\overline{\mathrm{Lc}}\cdot{}_p\mathrm{Auto}\\
\mathrm{Auto} & =\overline{\mathrm{Esb}}\cdot\mathrm{SbA}\cdot(\mathrm{Up}\cdot\overline{\mathrm{Down}}\cdot\mathrm{Lc}\cdot\mathrm{Vpb}+{}_p\mathrm{Auto}\cdot(\mathrm{Up}\cdot\overline{\mathrm{Down}}+\overline{\mathrm{Up}}\cdot\mathrm{Lc}))\\
\mathrm{Manual} & =\overline{\mathrm{Esb}}\cdot\mathrm{SbM}\cdot\overline{(\mathrm{Up}\cdot\mathrm{Down})}\cdot(\mathrm{Lc}\cdot\mathrm{Vpb}+{}_p\mathrm{Manual})\\
\mathrm{GoUp} & =\overline{\mathrm{Esb}}\cdot\overline{\mathrm{Up}}\cdot\mathrm{Lc}\cdot(\mathrm{SbA}\cdot{}_p\mathrm{Auto}\cdot(\mathrm{Down}+\overline{{}_p\mathrm{GoDown}})+\mathrm{SbM}\cdot\mathrm{Uppb}\cdot\overline{\mathrm{Dopb}}\cdot(\mathrm{Vpb}+{}_p\mathrm{Manual}))\\
\mathrm{GoDown} & =\overline{\mathrm{Esb}}\cdot\overline{\mathrm{Down}}\cdot\mathrm{Lc}\cdot(\mathrm{SbA}\cdot(\mathrm{Spb}\cdot\mathrm{Up}\cdot(\mathrm{Vpb}+{}_p\mathrm{Auto})+{}_p\mathrm{GoDown}\cdot({}_p\mathrm{Auto}+(\mathrm{Vpb}\cdot\mathrm{Up})))\\
 & \qquad +\mathrm{SbM}\cdot\mathrm{Dopb}\cdot\overline{\mathrm{Uppb}}\cdot(\mathrm{Vpb}+{}_p\mathrm{Manual}))
\end{cases}
$$

4 Obtained contol laws

4.1 Representation with recurrent Boolean equations

The control laws presented hereafter was obtained by translating the expression of the unknowns according to the projection-functions into relations between recurrent Boolean equations.

$$
\begin{cases}
\mathrm{fail}[k] & = \mathrm{esb}[k] \vee \mathrm{up}[k] \wedge \mathrm{down}[k] \vee \neg\mathrm{lc}[k] \wedge \mathrm{fail}[k-1] \vee \neg\mathrm{up}[k] \wedge \neg\mathrm{lc}[k] \wedge \mathrm{auto}[k-1] \\
\mathrm{auto}[k] & = \neg\mathrm{esb}[k] \wedge \mathrm{sbA}[k] \wedge (\mathrm{up}[k] \wedge \neg\mathrm{down}[k] \wedge \mathrm{lc}[k] \wedge \mathrm{vpb}[k] \\
& \qquad \vee\mathrm{auto}[k-1] \wedge (\mathrm{up}[k] \wedge \neg\mathrm{down}[k] \vee \neg\mathrm{up}[k] \wedge \mathrm{lc}[k])) \\
\mathrm{manual}[k] & = \neg\mathrm{esb}[k] \wedge \mathrm{sbM}[k] \wedge \neg(\mathrm{up}[k] \wedge \mathrm{down}[k]) \wedge (\mathrm{lc}[k] \wedge \mathrm{vpb}[k] \vee \mathrm{manual}[k-1]) \\
\mathrm{goUp}[k] & = \neg\mathrm{esb}[k] \wedge \neg\mathrm{up}[k] \wedge \mathrm{lc}[k] \wedge (\mathrm{sbA}[k] \wedge \mathrm{auto}[k-1] \wedge (\mathrm{down}[k] \vee \neg\mathrm{goDown}[k-1]) \\
& \qquad \vee\mathrm{sbM}[k] \wedge \mathrm{uppb}[k] \wedge \neg\mathrm{dopb}[k] \wedge (\mathrm{vpb}[k] \vee \mathrm{manual}[k-1])) \\
\mathrm{goDown}[k] & = \neg\mathrm{esb}[k] \wedge \neg\mathrm{down}[k] \wedge \mathrm{lc}[k] \wedge (\mathrm{sbA}[k] \wedge \mathrm{spb}[k] \wedge \mathrm{up}[k] \wedge (\mathrm{vpb}[k] \vee \mathrm{auto}[k-1]) \\
& \qquad \vee\mathrm{sbA}[k] \wedge \mathrm{goDown}[k-1] \wedge (\mathrm{auto}[k-1] \vee (\mathrm{vpb}[k] \wedge \mathrm{up}[k])) \\
& \qquad \vee\mathrm{sbM}[k] \wedge \mathrm{dopb}[k] \wedge \neg\mathrm{uppb}[k] \wedge (\mathrm{vpb}[k] \vee \mathrm{manual}[k-1]))
\end{cases}
$$

These control laws can be automatically translated in the syntax of the Ladder Diagram language ([IEC03]) before being implemented into a PLC. The complete code is composed of only 10 rungs.

5 Conclusions

For this case study, the method we propose has allowed to find the control law for the press. The use of priority rules and optimization criteria has simplified greatly the formalization of the requirements.

References

[Bro03] Frank Markham Brown. *Boolean Reasoning: The Logic of Boolean Equations.* Dover Publications, 2003.

[Gui11] Anaïs Guignard. Symbolic generation of the automaton representing an algebraic description of a logic system. Master's thesis, ENS Cachan, July 2011.

[Hie09] Yann Hietter. *Synthèse algébrique de lois de commande pour les systèmes à évènements discrets logiques.* PhD thesis, ENS Cachan, May 2009.

[HRL08a] Yann Hietter, Jean-Marc Roussel, and Jean-Jacques Lesage. Algebraic synthesis of dependable logic controllers. In *Proceedings of 17th IFAC World Congress, 2008*, pages pp. 4132–4137, Seoul, South Korea, July 2008.

[HRL08b] Yann Hietter, Jean-Marc Roussel, and Jean-Jacques Lesage. Algebraic synthesis of transition conditions of a state model. In *Proceedings of 9th International Workshop On Discrete Event Systems (WODES'08)*, pages 187–192, Göteborg, Sweden, May 2008.

[Huf54] D. A. Huffman. The synthesis of sequential switching circuits. *J. of the Franklin Institute*, 257(3-4):161–190 and 275–303, 1954.

[IEC03] IEC 61131-3. *IEC 61131-3 Standard: Programmable controllers - Part 3: Programming languages.* International Electrotechnical Commission, 2 edition, 2003.

[Rud01] Sergiu Rudeanu. *Lattice Functions and Equations (Discrete Mathematics and Theoretical Computer Science).* Springer, 2001.