ViSim User Guide

To Create Interlock Logic

For

Programmable Interlock Platform

Control & Information Technology Group
134 W Rio Robles Drive
San Jose, CA 95134

Main: 408.750.0300
Fax: 408.750.2990

Manual Rev. 1.1
05/11

Copyright

Preface

About this manual

This manual is designed to serve as a guideline to use ViSim to create and simulate programmable interlock logic for programmable interlock on Programmable Interlock Platform. The information contained within this manual, including product specifications, is subject to change without notice. Please observe all safety precautions and use appropriate procedures when handling the Programmable Interlock Platform product and its related software.

Table of Contents

Table of Figures

Revision History

Revision	Description of changes	Date
1.0	First Release	09/2009
1.1	Minor update, address change	05/2011

1 Introduction

The ViSim is a GUI tool based on MS Office VISIO with underlying Visual Basic code. The tool enables the user to build interlock logic by interconnecting relays symbols, simulate the design, and generate a 'csv' file with the same format as produced by the GUI Web tool. This file can then be loaded into the Programmable Interlock platform.

1.1 Conventions used in this User Manual

	**Warning**	**The WARNING sign denotes a hazard to personnel. It calls attention to a procedure, practice, condition, or the like, which, if not correctly performed or adhered to, could result in injury to personnel.**
	**Caution**	**The CAUTION sign higlights information that is important to the safe operation of the platform, or to the integrity of your files. .**
	**Note**	The NOTE sign denotes important information. It calls attention to a procedure, practice, condition, or the like, which is essential to highlight.

On screen buttons or menu items appear in bold and cursive.
Example: Click ***OK*** to save the settings.

Keyboard keys appear in brackets.
Example: [ENTER] and [CTRL]

Pages with additional information about a specific topic are cross-referenced within the text.

2 Installation

2.1 Requirements

The following applications need to be installed.

1. MS Office Excel 2003 or 2007.

NOTE: Other versions of MS Office Excel are not supported at this time of writing.

2. MS Office Visio 2003.

2.2 Installation

1. Copy the ViSim folder into your local drive. This folder is available via download from the MKS website.
2. Change the security level for macros to 'Low' as follows:
   - Open blank Visio document.
   - On the menu bar: Tools – Options – Security (tab) – Macro Security…(button) : check 'Low' security radio button.

Note If you check a higher security option the macros will be blocked

Note You may need to restart the Visio for the changes to take effect.

Tutorial

This section will guide you through the process of designing, simulating and generating the csv file by using a simple example.

2.1 Create New File:

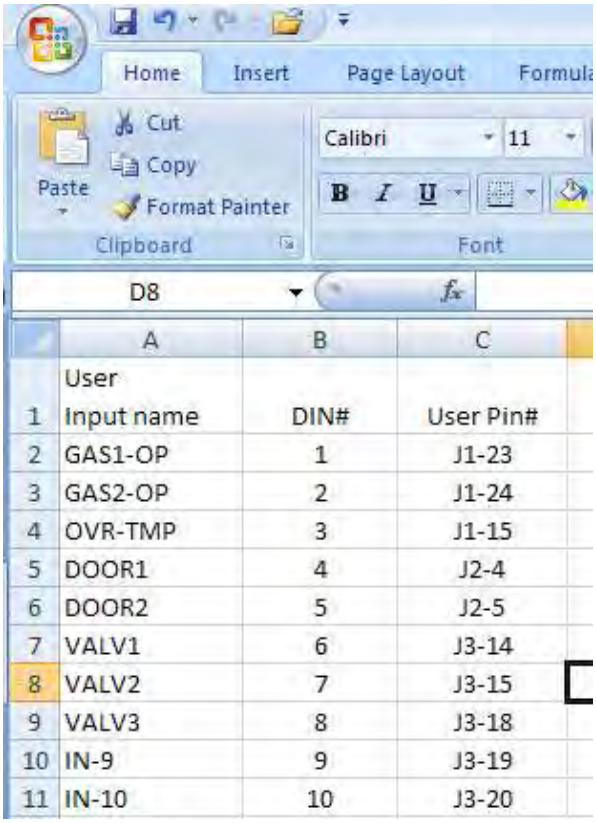
Open the ViSim folder and double-click on 'ViSim Drawing Template.vsd'. This file is a template with underlying VB code. 'Save As' this file into the same folder with your selected name.

2.2 I/O Map File:

Before drawing the logic, you need to have an Excel file with the I/O definition. By default the tool will refer to the following file:

.....\ViSim\work\IO_MAP.xlsx

The file contains two sheets, the first for the inputs, second for the outputs. Please use default labeling of the first sheet as "**In**" and the second sheet as "**Out**".
The sheet structure is shown below: The first column contains the input user name (or signal name), and the second column contains the index of this input in the Programmable Interlock. The third column contains the location of this input on the Distribution board. Please use this similar structure for the second sheet.



	A	B	C
	User		
1	Input name	DIN#	User Pin#
2	GAS1-OP	1	J1-23
3	GAS2-OP	2	J1-24
4	OVR-TMP	3	J1-15
5	DOOR1	4	J2-4
6	DOOR2	5	J2-5
7	VALV1	6	J3-14
8	VALV2	7	J3-15
9	VALV3	8	J3-18
10	IN-9	9	J3-19
11	IN-10	10	J3-20

Figure 1 Example of IO Map Excel Spreadsheet file.

2.3 Draw The Logic:

From the ‘Drawing’ stencil on the left of the page, drag two ‘INPUT’ one RELAY and one OUTPUT symbols and drop them on the page. Drag the LINE symbol and drop it on the page. Then stretch the edges of the line with the mouse left click being hold down and tie them to the symbol pins as shown in the following figure. You’ll need to drag the LINE 3 times.

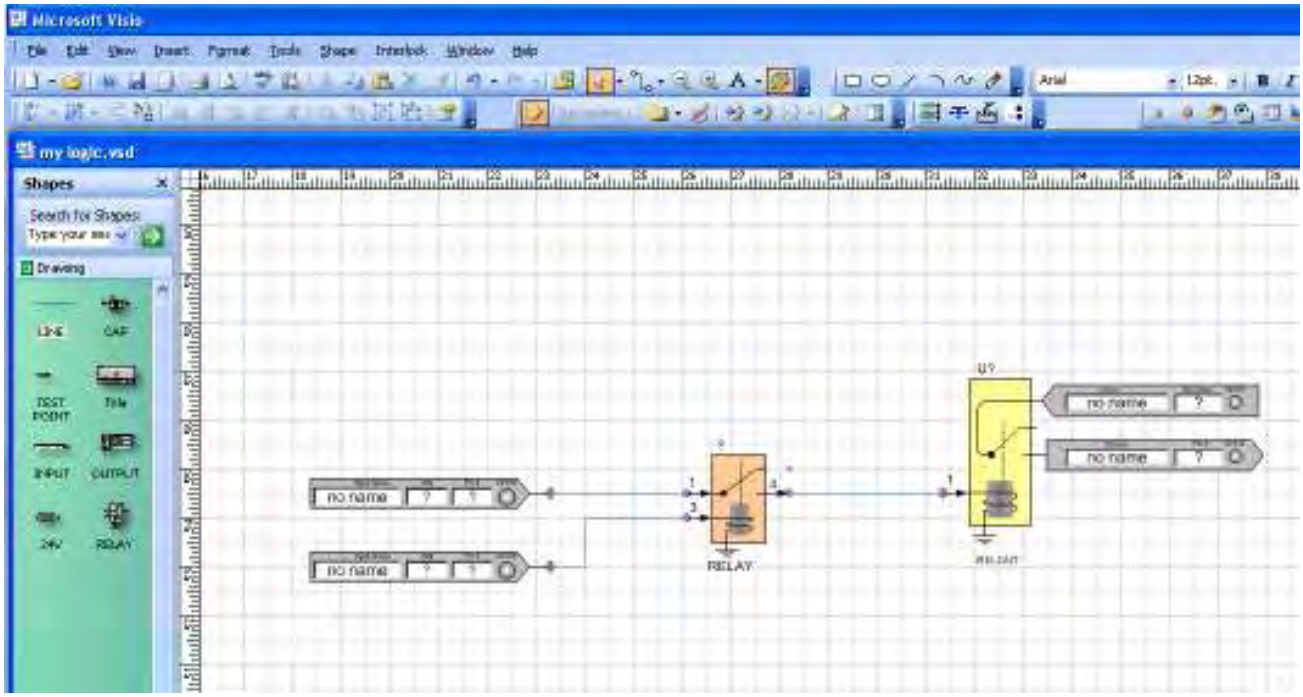


Figure 2 Simple one AND gate logic example with ViSim

2.4 Assign Inputs and Output.

Place the mouse pointer close to the INPUT symbol, until you see a small menu item as shown on the figure below.

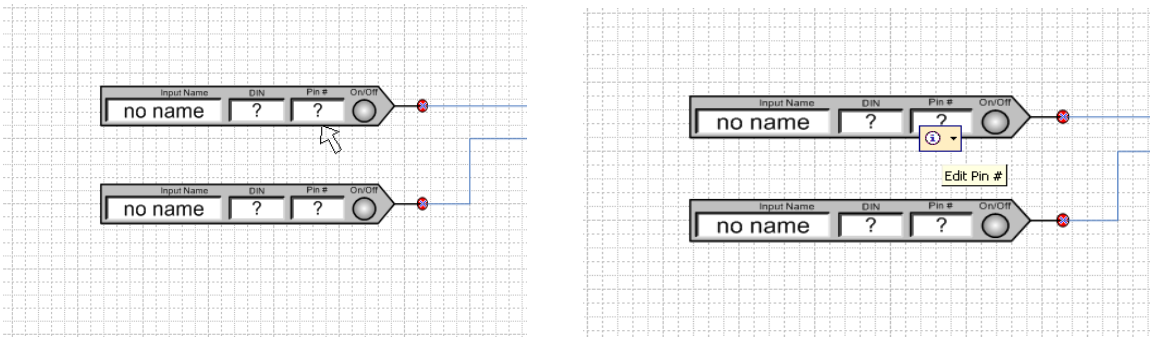


Figure 3 Selecting Input using drop down menu in ViSim

NOTE: If the input symbols are place closed to each other, the input menu might not be easily accessible by moving the mouse over the input symbol. User can change input signal by selecting the input symbol with menu bar->Interlock->Edit I/O as shown below.

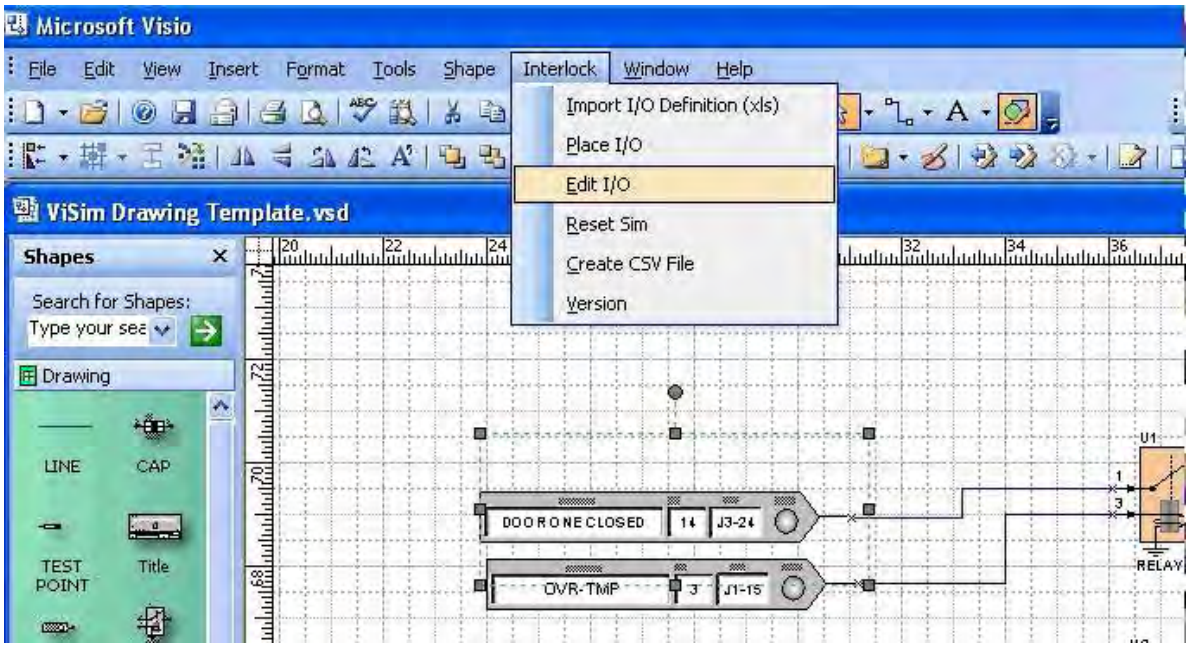


Figure 4 Selecting Input using drop down menu in ViSim

Click on this menu item and then click again on the ‘Edit Pin #’ sub item.

You will get a form as shown on the right. The data within this form is taken from the IO_MAP.xlsx file. Click on any one of the items in the list, and then click the ‘Assign’ button. The form will be closed and the selected data will appear on the INPUT symbol. Do the same for the second INPUT symbol and the OUTPUT symbol.
At this point the I/O is assigned to the input and output symbols.

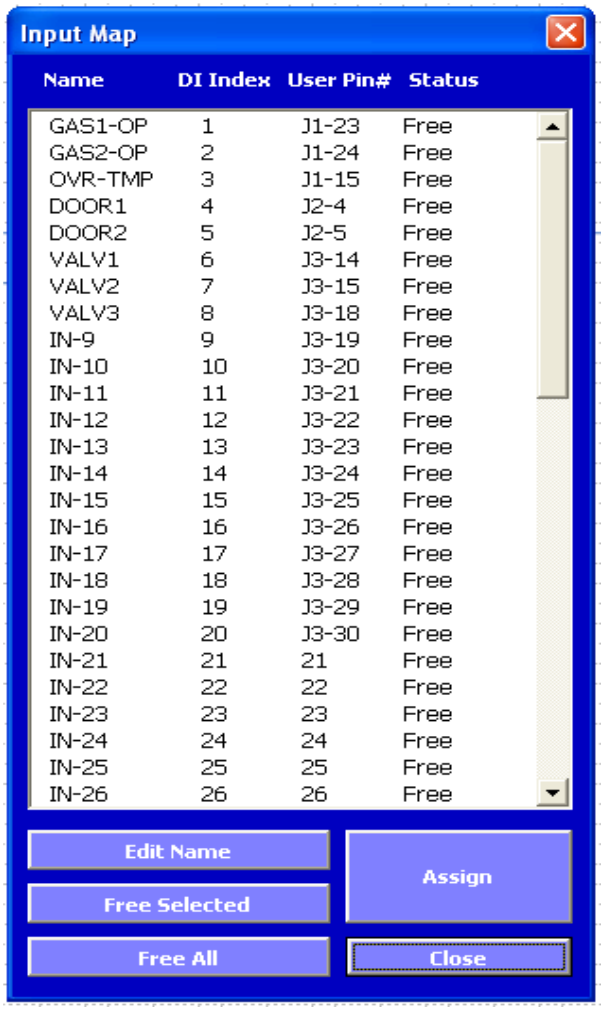


Figure 5 IO selection from spreadsheet IO file

2.5 Generate CSV (Coma-Separated Value) File.

Click on the Visio tool bar icon as shown in the following figure.

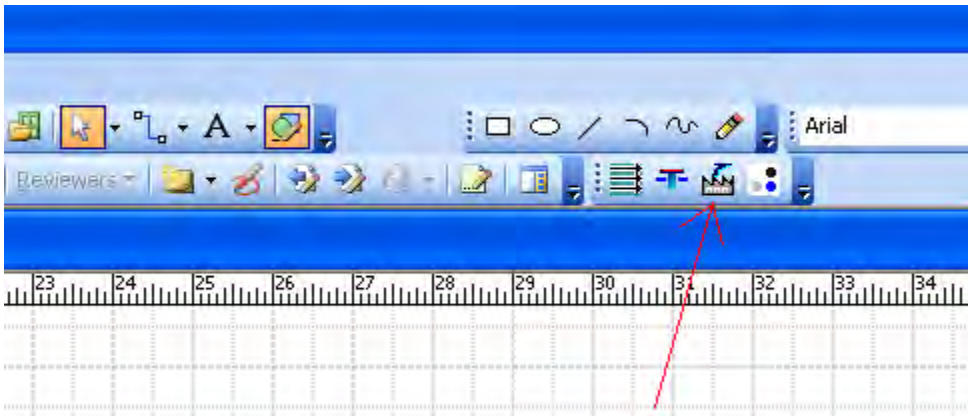


Figure 6 Generating CSV file from Visio ViSim Tool.

This will activate the netlist generator and the csv file generator. At the end of netlist generation a message 'Connections Complete' will appear on the screen. Press OK. Then a message 'csv File creation completed' will appear. Press OK again.
At this point you should see the following on the Visio page. Note that the symbols are designated, and also note that the small red circles around the pins disappeared to indicate the connection has been made.

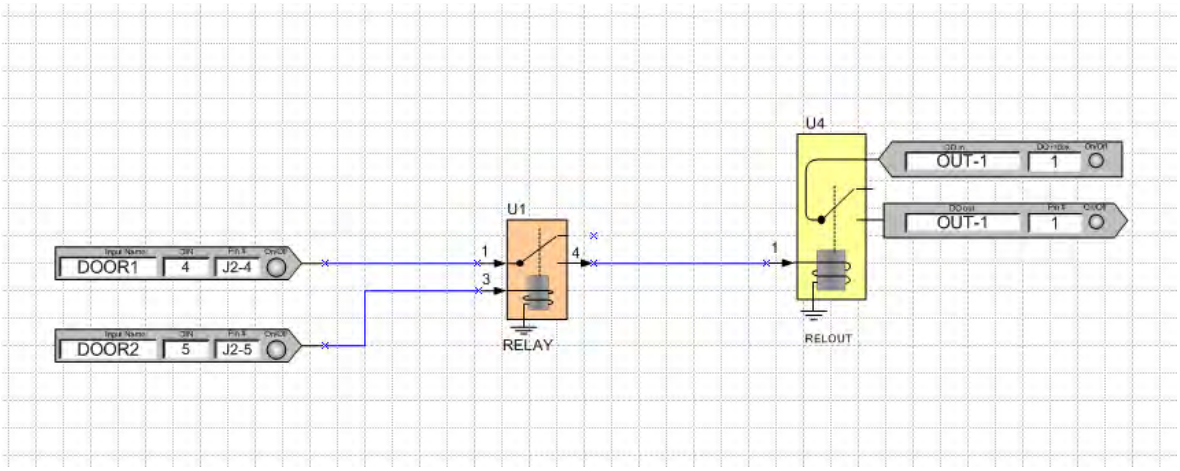


Figure 7 Complete interlock sample design ready for simulation

Note: All input symbols used in interlock design has to be assigned with signal name. Leaving a connected input symbol with “no name” will result in error. Error path will be highlighted with red wire to the un-assigned input signal as shown below

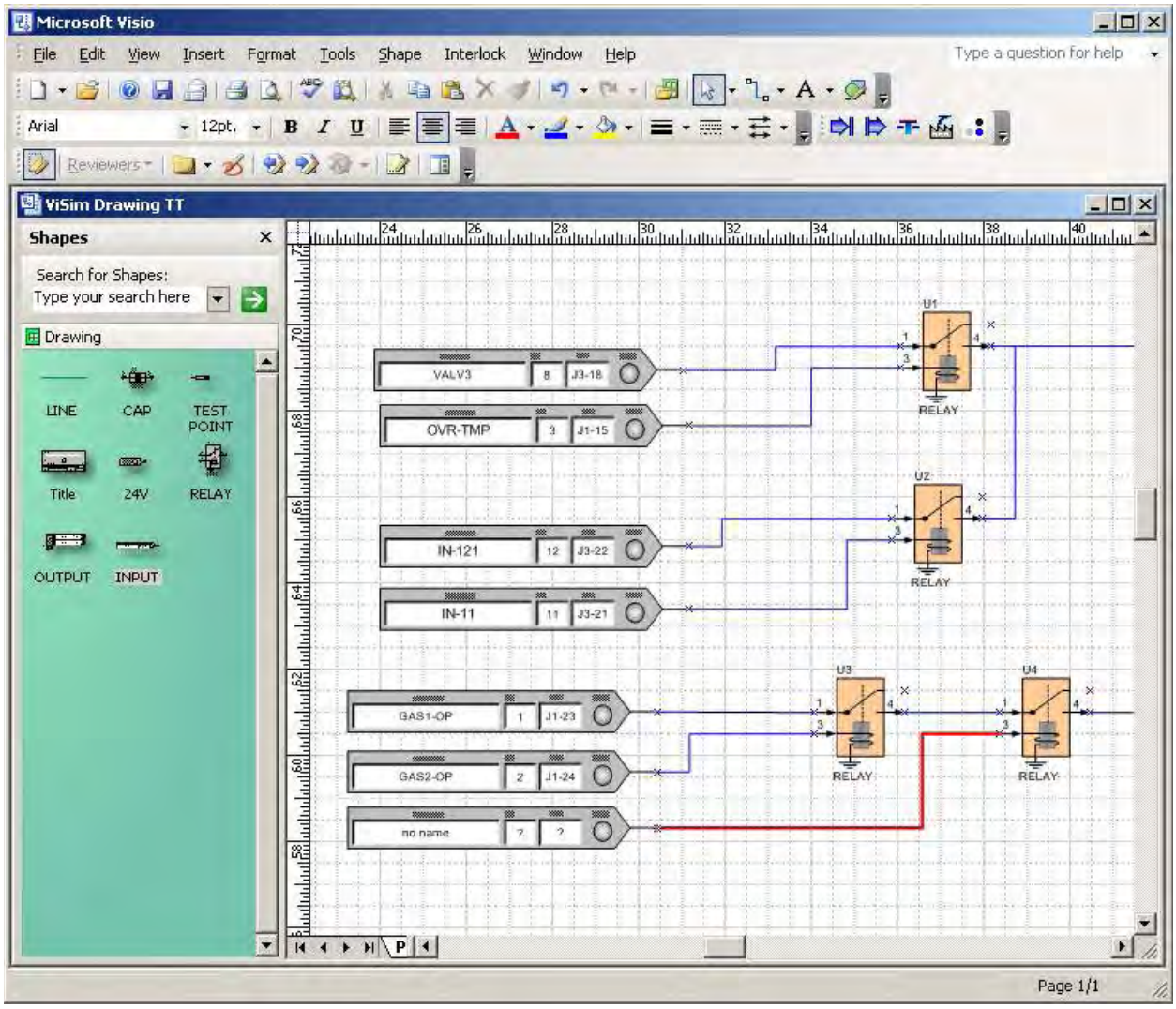


Figure 8 Complete interlock sample design ready for simulation

2.5.1 Logic for the example

The example above will generate the below logic:

OUT1 = IN4 and IN5

Figure 9 Equivalent Logic for the simple example above.

2.5.2 The Default CSV File:

The created file is: …\ViSim\work\intlkcfg_user.csv

The following figure shows the logic definition section in this file, where we have one AND term composed of the inputs 4 and 5. This AND term is assigned to output 1.

114	64	Input 64			
115	65	Watchdog			
116	IN	O1	O2	O3	O4
117	4	A1	-	-	-
118	5	A1	-	-	-
119	OUTFB	O2	O3	O4	O5
120	OUTNUM				
121					
122					

Figure 10 CSV file generated by ViSim Tool for example above

2.6 Simulate the Design

Turn on and off the inputs by moving the mouse pointer as shown in the figure to activate the drop down menu.

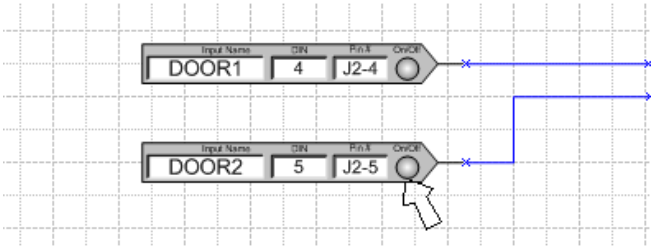


Figure 11 Button to toggle inputs for simulation.

Then click ON or OFF to simulate the output

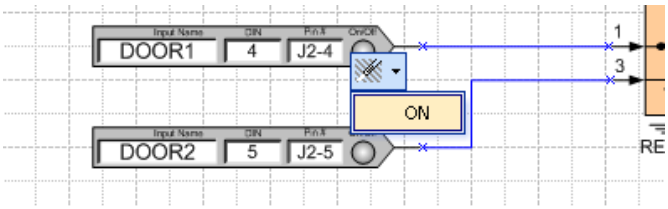


Figure 12 Toggle an input ON or OFF to simulate programmable logic design

Result when both inputs are ON:

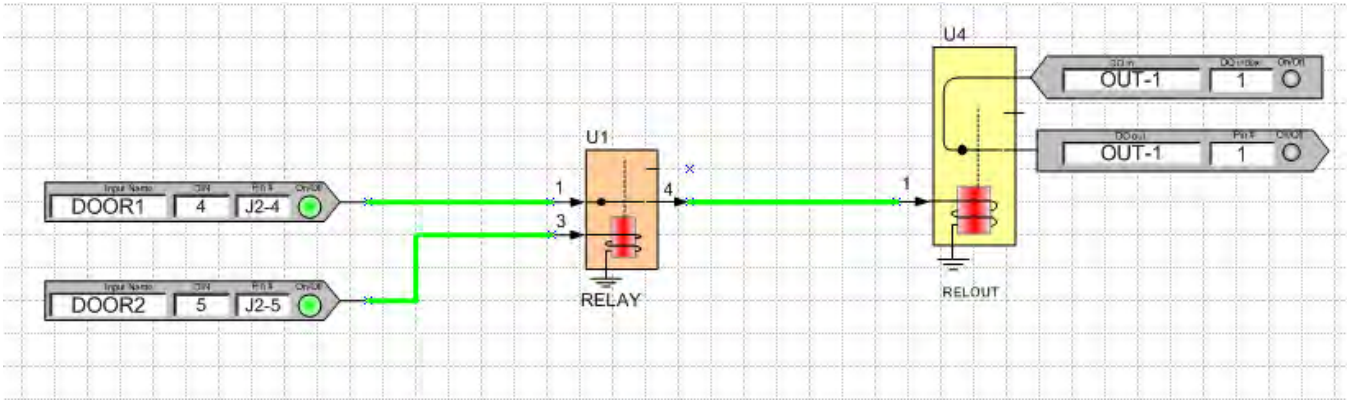


Figure 13 Result of simulation when logic is TRUE and activates the output.

To reset the simulator click on the tool bar icon as shown below:

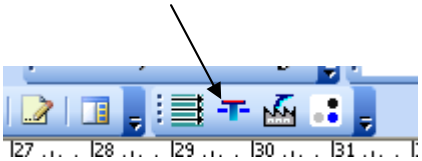


Figure 14 Reset simulation condition on ViSim tool.

3 Logic Structure

3.1 Combined AND/OR Logic Term

Relays can be connected to form various logic structures. The following example shows a combination of AND and OR terms.

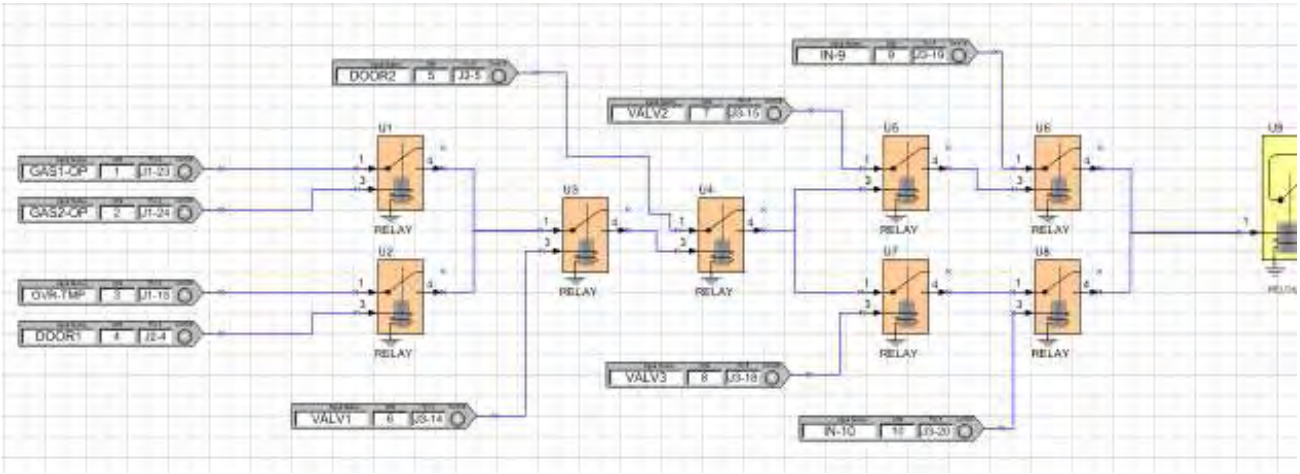


Figure 15 More complex logic structure

The logic term represented by this structure would be:
Out1 = (1 and 2 or 3 and 4) and 5 and 6 and (7 and 9 or 8 and 10)

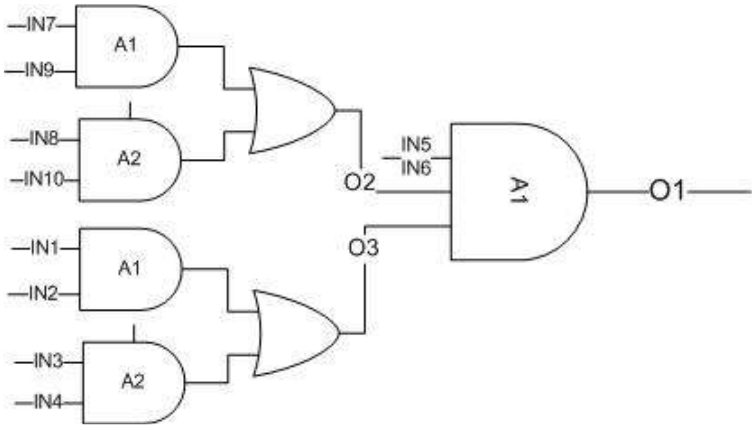


Figure 16 Equivalent logic gate for figure 14

Note

The above logic depicts the equivalent logic equation for interlock logic condition, not the actual TTL gate level for timing analysis purposes.

The resulting ‘csv’ is on the right.

Note that the tool uses other free outputs (actual or virtual in this case O2 and O3) to form the logic parts and feed them back to output 1.

114		64 Input 64			
115		65 Watchdog			
116	IN	O1	O2	O3	O4
117	1	-	-	A1	-
118	2	-	-	A1	-
119	3	-	-	A2	-
120	4	-	-	A2	-
121	5	A1	-	-	-
122	6	A1	-	-	-
123	7	-	A1	-	-
124	8	-	A2	-	-
125	9	-	A1	-	-
126	10	-	A2	-	-
127	OUTFB	O2	O3	O4	O5
128	2	A1	-	-	-
129	3	A1	-	-	-
130	OUTNUM				
131					

Figure 17 ViSim generated CSV file for more complex logic design above

3.2 Latch Structure

The following structure realizes a latch function. Given that the upper AND term (1 * 2 * 5 * 6) is true, when the inputs 3 and 4 (the latch condition) turns ON, the relay U5 will be energized for a short period as dictated by the capacitor, and then due to the feedback on U5 (pin 4 to pin3) the relay will be kept energized. Once the upper condition turns false, the relay U5 will be de-energized.

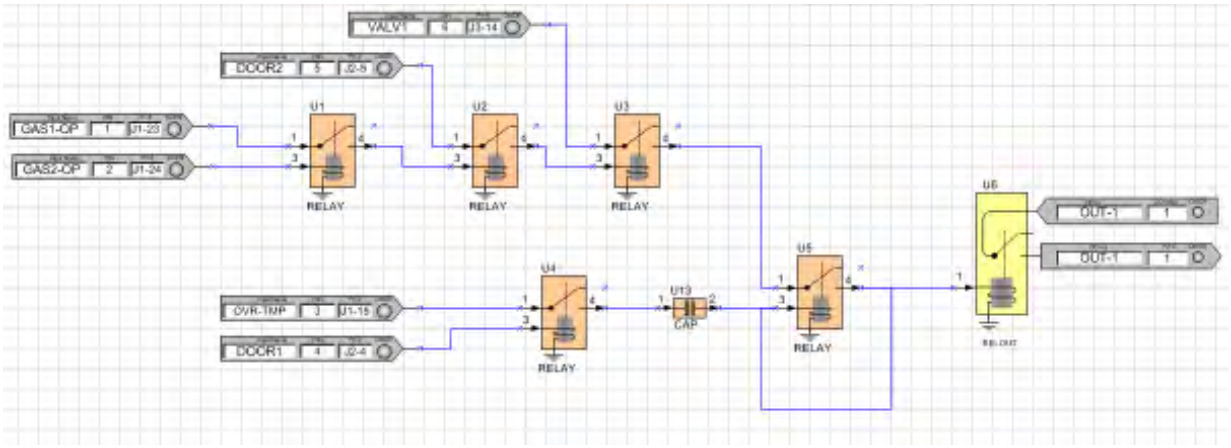


Figure 18 Example of Latch and Clear Interlock design

This is translated into the csv file as shown on the right:

The latch condition is assigned to and term #1 (A1). The latched function is assigned to A3. The clear condition is the same latched function inverted by using a free output (in this case O2) and then fed back to O1 as term #2.

115	65	Watchdog		
116	IN	O1	O2	O3
117	1	A3	A1	-
118	2	A3	A1	-
119	3	A1	-	-
120	4	A1	-	-
121	5	A3	A1	-
122	6	A3	A1	-
123	9	-	-	-
124	10	-	-	-
125	OUTFB	O2	O3	O4
126	2	A2	-	-
127	OUTNUM	A1->L	INV	
128		A2->C		
129				

Figure 19 CSV file with Latch condition

Below is the truth Table for **A1->L** and **A2->C** condition.

Interlock Output Current State	AND gate 2 Output	AND gate 1 Output	Interlock Output **FINAL** STATE
FALSE	FALSE	FALSE	FALSE
XX	FALSE	TRUE	TRUE
XX	TRUE	XX	FALSE
TRUE	FALSE	XX	TRUE

4 Loading ViSim CSV file to Programmable Interlock Platform

At the time of this writing, Programmable Interlock Platform only supports loading of csv through Webpage. Please refer to the user manual for connecting to the unit for Webpage access.

4.1 Network Configuration

The Programmable Interlock Platform network settings configure the 100 BaseT Ethernet ports. The following are factory-default Ethernet settings.

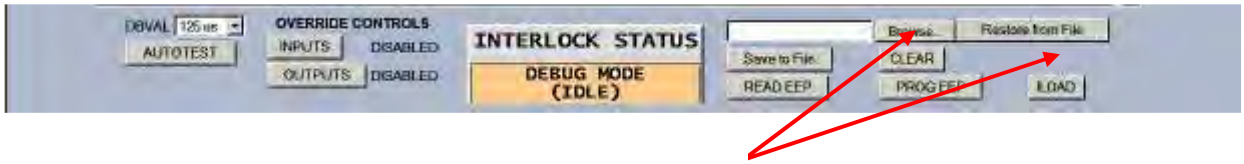
Parameter	Setting
IP-Address	192.168.0.X
Subnet mask	255.255.255.0
Default Gateway	None

4.2 Connecting to a computer via TCP/IP

Open a web browser window and enter the IP address of the unit. There is a slight delay as the unit transfers content to your local browser. The main Device page is displayed. If you are unable to load the page, check the units switch settings and ensure the IP address of the PC has is set as DHCP. Also check the Ethernet cable connections to your PC and to the Programmable Interlock unit.

4.3 Loading CSV file to Unit

When complete with ViSim simulation, and CSV file is generated. On the unit's Webpage Interlock Tab, Click **Browse…**button to load the modified configuration file then click the **Restore from File** button on the Interlock Webpage to load the changed name.



Browse and Restore from File buttons

Figure 20 Browse to the CSV file then Click Restore from File to load logic to unit

Caution	**Logic will not be stored into Non-Volatile memory until user click on PROG EEP button**
Note	Interlock has to be in IDLE Mode before it can be programmed. Reset unit or power toggle to put unit in IDLE Mode before programming

4.4 Program Logic into Non-Volatile Memory

After logic from CSV file is loaded, verify that the correct interlock and Inputs/.Outputs names are correct per ViSim design as shown in the figure below

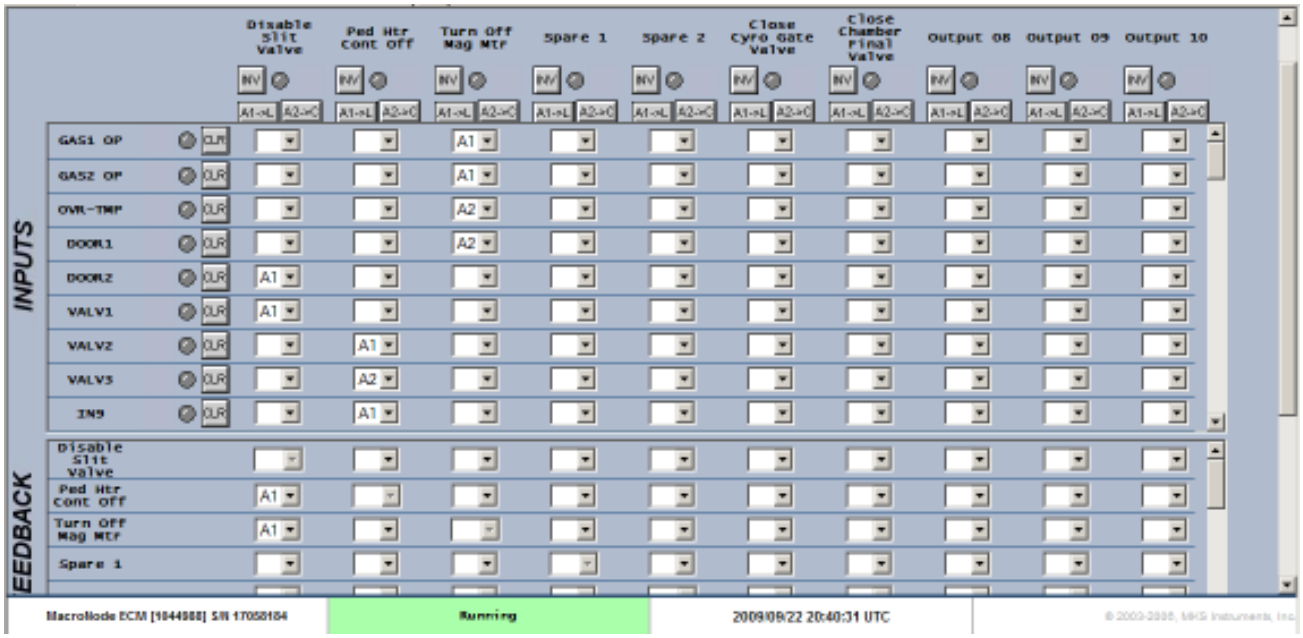
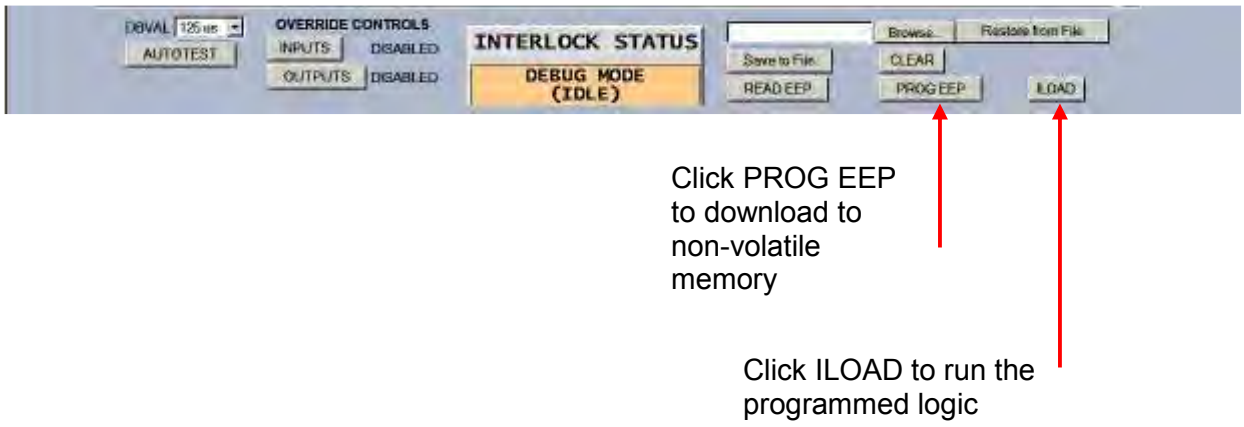
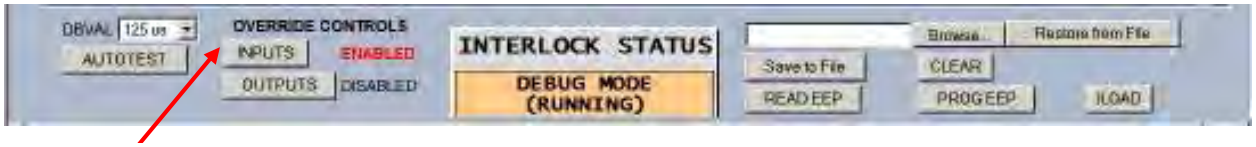


Figure 21 Interlock Webpage after ViSim CSV file is loaded.

1. Once logic is verified, click on **PROG EEP** button to download the logic to non-volatile memory, then click **ILOAD** button to run the download programmed logic.



2. Interlock Status will show Debug (Running). To Test the logic on the platform, click on **INPUT** in the OVERRIDE CONTROLS to allow manual inputs driving.



Click to enable manual Input driving for debugging purposes.

	**Caution**	**Logic will not be stored into Non-Volatile memory until user click on PROG EEP button**
	**Caution**	Refresh Webpage and READ EEP again to make sure updated page is shown instead of reading stale page from system cache.