Cookie Factory Virtual Model Documentation: Ignition HMI

Contents

[Introduction 2](#_Toc200016741)

[Ignition Gateway 2](#_Toc200016742)

[Description 2](#_Toc200016743)

[Connection to OpenPLC 2](#_Toc200016744)

[HMI in Ignition Designer 8](#_Toc200016745)

[Description 8](#_Toc200016746)

[Opening the HMI 8](#_Toc200016747)

[Ignition Tags 16](#_Toc200016748)

[Description 16](#_Toc200016749)

[HMI Tags 16](#_Toc200016750)

[dv 17](#_Toc200016751)

[flour\_alarm 17](#_Toc200016752)

[flour\_weight 17](#_Toc200016753)

[hopper\_alarm 17](#_Toc200016754)

[hopper\_weight 17](#_Toc200016755)

[rv\_1 17](#_Toc200016756)

[rv\_2 17](#_Toc200016757)

[rv\_3 17](#_Toc200016758)

[sugar\_alarm 18](#_Toc200016759)

[sugar\_weight 18](#_Toc200016760)

[vacuum 18](#_Toc200016761)

[vacuum\_alarm 18](#_Toc200016762)

[vacuum\_freq 18](#_Toc200016763)

[vacuum\_manual 18](#_Toc200016764)

[vacuum\_rpm 18](#_Toc200016765)

# Introduction

The purpose of this document is to aid you in the modification, emulation, or use of the Ignition human machine interface (HMI) for the Cookie Factory Virtual Model, part of CCI subtask 5.1.

While steps to open the HMI project file and to start the Ignition Gateway will be discussed in this document, it is highly recommended that you first complete at least the first 5 steps of the startup guide, created by inductive automation, before proceeding:

<https://www.docs.inductiveautomation.com/docs/8.1/getting-started/quick-start-guide>

The following document contains a general overview of what a human machine interface (HMI) is, how to configure the ignition gateway to connect to OpenPLC, how to load the project file in the Ignition designer, tags used in the HMI, what a tag is, and descriptions of the different visual components of the HMI as well as how they are influenced by each tag.

Note that this document is in no way complete, because the model itself is not complete. So, check back in the box folder for any updates to the HMI and its documentation, though I will probably let you know when an update occurs.

# Ignition Gateway

## Description

The ignition gateway is a web server that allows you to configure various components of your ignition projects, user permissions, device connections, tag diagnostics, etc. It essentially acts as a control center for all your ignition projects. We will be using the gateway to connect to our OpenPLC device.

## Connection to OpenPLC

Begin by opening your Ignition Gateway, by default it should be using port 8088 on your local machine: <http://localhost:8088>. If you need more guidance with how to open/set up the Ignition Gateway be sure to check out step 1 and 2 of the start-up guide. On start up you should see the following:

A screenshot of a computer

AI-generated content may be incorrect.

Next, we want to add a Modbus device to our service. Navigate to the configuration tab on the left.

A screenshot of a computer

AI-generated content may be incorrect.

Now you should have the configuration page open, scroll down on the left side of the screen until you reach the OPC UA section. Next, click on device connections.

A screenshot of a computer

AI-generated content may be incorrect.

Now you’ll se an empty screen, we’ll want to add an OPC UA device that represents our OpenPLC device. Select “Create new Device…”

A screenshot of a computer

AI-generated content may be incorrect.

You should see multiple device options to choose from, scroll down until you see “Modbus TCP” and select it. Scroll down and select “Next.”

A screenshot of a computer

AI-generated content may be incorrect.

Now you should be able to select your device name, the hostname/port to connect to, and adjust any advanced settings. Match the hostname/port to what is set as your hostname/port in OpenPLC runtime, by default it should be 127.0.0.1 (localhost) and 502 for the port number.

A screenshot of a computer

AI-generated content may be incorrect.

Next, we need to change the indexing setting for Ignitions Modbus addressing. Ignition uses 1-based indexing for Modbus addresses so without changing this setting all our addresses will be 1 register off their actual value. To change this, check the “Show advanced properties” box near the bottom of the page. Then scroll down to “Zero-based Addressing” and check the box to true.

A screenshot of a computer

AI-generated content may be incorrect.

After specifying all the necessary properties, scroll down to the bottom of the page and select “Create New Device.”

A screenshot of a computer

AI-generated content may be incorrect.

Congratulations! You’ve now created a new device to represent your OpenPLC runtime, if your hostname and port are set correctly you should see your new device and that its status is “Connected.” Now that we have created a new device, and it’s connected to our OpenPLC runtime instance, we need to properly address any relevant tags for our Ignition HMI. To do this click on the “Addresses” option under “More.”

A screenshot of a computer

AI-generated content may be incorrect.

Now you should see a blank addresses page that allows you to add rows or import a configuration. Click the “Choose File” button and upload the configuration file named modbus-config.csv in the directory of this document on Box. Then click “Import Configuration.”

A screenshot of a computer

AI-generated content may be incorrect.

Now you should see multiple rows of different addresses that are set to different Modbus addresses. The prefix and start/ends are set to be reminiscent of OpenPLC’s addressing system, however you should be able to change the prefix of each address as you please.

A screenshot of a computer

AI-generated content may be incorrect.

Click save at the bottom of the page and now we are ready to open the HMI in Ignition Designer.

# HMI in Ignition Designer

## Description

A human machine interface (HMI) is a system that allows people to interact and control different physical processes, systems, or machines. Typically, an HMI allows the user to see a simplified view of different systems and their relative information. Multiple HMIs may exist within a single physical facility, often encompassing different parts of facilities operations. Currently, the project file uploaded to box contains the HMI that represents the ingredient allocation process of the Cookie Factory Model.

## Opening the HMI

After installing/opening the designer launcher, as outlined in step 5 of the startup guide, you will be able to add a designer using the default gateway and your personal device as shown below. If you need more assistance with this step be sure to check out the following page:  
<https://www.docs.inductiveautomation.com/docs/8.1/launchers-and-workstation##adding-applications>

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

Select your new designer and open it. You will be asked to log in, the default log in is username: admin, and password: password, otherwise it will be the credentials you first created during Ignition installation.

A screenshot of a computer

AI-generated content may be incorrect.

Next you will see a page that has all your active projects, yours will be blank. To import the HMI for the cookie factory, select import project as shown below.

A screenshot of a computer

AI-generated content may be incorrect.

This will open your file explorer where you can select your project zip file. For the cookie factory download and select the project zip file located in box named ManufacturingModel\_CookieFactory\_2025\_2025-06-02\_1644 in the same directory as this document.   
A screenshot of a computer

AI-generated content may be incorrect.

Next you should see a screen showing all the project configurations, leave these as is and then select “Import Project” as shown below.  
A screenshot of a computer

AI-generated content may be incorrect.

Now you should see this new project in your list of projects, select “Open” next to the project name.

A screenshot of a computer

AI-generated content may be incorrect.

Congratulations! You have now opened the project file, next navigate to View->Windows->Main\_View as shown below. Double click on Main\_View.

A screenshot of a computer

AI-generated content may be incorrect.

After opening Main\_View you should see the following:

A screenshot of a computer

AI-generated content may be incorrect.

If you see any errors, such as “Trial Expired” or the tag values in red font, be sure to reset your trial in the ignition gateway or ensure you are connected to the OpenPLC runtime instance.

There are three main windows in Ignition Designer that you should direct your attention to. First there is the “Project Browser” which shows all project components from scripts to visual elements, alarms, reports, and database queries.

A screenshot of a computer

AI-generated content may be incorrect.

Next is the “Tag Browser” where you can see all the tags included in the HMI, these numbers will change as the python simulation runs in accordance with an operator’s inputs into the HMI.

A screenshot of a computer

AI-generated content may be incorrect.

Finally, you have the main window which will show you the HMI in it’s current state. Each numeric value, two state toggle, and alarm is tied to a tag. The tag values will change in accordance with the HMI and these changes will also be reflected in OpenPLC’s monitor.

A computer screen shot of a computer

AI-generated content may be incorrect.

To manipulate these values, we first must enable read/write mode and then preview mode. This will allow us to click each two-state toggle and change the values accordingly.

A screenshot of a computer

AI-generated content may be incorrect.

Congratulations! Now you should understand how to load a project into Ignition Designer and manipulate said project.

# Ignition Tags

## Description

Tags are points of data that are statically or dynamically changed. Tags can be pulled from expressions, databases, or through Open Platform Communications (OPC) addresses. In our case tags are pulled from our OpenPLC Modbus device that is connected to Ignition’s default OPC Unified Architecture (UA) server.

## HMI Tags

Although our model has dozens of tags defined in its OpenPLC program, our HMI only needs access to tags that directly influence the operators’ control over the model’s systems. For this reason, our HMI only has fourteen tags, each of these tags corresponds to at least one visual element in our HMI. Now we will go over each tag, what they represent, and what visual element they correspond to.

### dv

This tag represents the status of the diverter valve, if the value is set to 0, the valve is set to siphon sugar, if the value is set to 1, the valve is set to siphon flour.

### flour\_alarm

This tag represents the status of the flour alarm, if the value is set to 0, then the flour\_weight tag has a value greater than 1000, if the value is set to 1, then the flour\_weight tag has a value less than 1000.

### flour\_weight

This tag represents the total weight of the material in the flour silo.

### hopper\_alarm

This tag represents the status of the hopper alarm, if the value is set to 0, then the hopper\_weight tag has a value greater than 100, if the value is set to 1, then the hopper\_weight tag has a value less than 100.

### hopper\_weight

This tag represents the total weight of the material in the hopper

### rv\_1

This tag represents the rotary valve that controls the input of flour into the pneumatic conveying system. The rotary valve is open and spinning if the value of rv\_1 is 1. The rotary valve is closed and not spinning if the value of rv\_1 is 0.

### rv\_2

This tag represents the rotary valve that controls the input of sugar into the pneumatic conveying system. The rotary valve is open and spinning if the value of rv\_2 is 1. The rotary valve is closed and not spinning if the value of rv\_2 is 0.

### rv\_3

This tag represents the rotary valve that controls the output of material from the hopper into the mixer. The rotary valve is open and spinning if the value of rv\_3 is 1. The rotary valve is closed and not spinning if the value of rv\_3 is 0.

### sugar\_alarm

This tag represents the status of the sugar alarm, if the value is set to 0, then the sugar\_weight tag has a value greater than 1000, if the value is set to 1, then the sugar\_weight tag has a value less than 1000.

### sugar\_weight

This tag represents the total weight of the material in the sugar silo.

### vacuum

This tag represents the on/off state of the vacuum for the pneumatic conveying system. If the value is 1, then the pneumatic conveying system will move material. If the value is 0, then the pneumatic conveying system will not move material.

### vacuum\_alarm

This tag represents the status of the vacuum alarm. If the value is set to 0, then the vacuum is performing at a reasonable speed for the material being conveyed. If the value is set to 1, then the vacuum is underperforming in speed for the material being conveyed leading to saltation.

### vacuum\_freq

This tag represents the frequency of the vacuum’s motor. The higher the frequency the higher the RPM and vice versa.

### vacuum\_manual

This tag represents the state of the frequency control for the vacuum. If the value is set to 0, then the vacuum is set to an automatic frequency based on the material being conveyed. If the value is set to 1, then the vacuum’s frequency can be manipulated by the operator.

### vacuum\_rpm

This tag represents the rpm of the vacuum’s motor. The higher the rpm, the faster material is conveyed.

The location of all visual elements and their relevant tags can be seen below.

