# Summary of Qualitative Physics using Dimensional Analysis

R. Bhaskar and Anil Nigam
IBM Thomas J. Watson Research Center, P.O. Box 704,
Yorktown Heights, NY 10598, USA

## Introduction

Dimensional analysis as a method for solving problems in qualitative physics without prior knowledge of physical laws that govern the operation of such devices. Once dimensional analysis has identified the dimensionless groups (π-groups) that govern a system, the next step is to use these groups as the basis for a qualitative model.

π-calculus is a formal framework for qualitative reasoning. Instead of modeling the system as static equations, by employing π-calculus, we can treat the physical variables as processes and the π-groups as channels through which these processes communicate. Changes to one variable (process) propagate through these channels, affecting other connected processes.

This way, we can build a computer programming algorithm that model the operations of the device without prior knowledge of how the devices work, by following the π-calculus framework.

## Oversimplified analogy

I would like to start this summary with over-simplified analogy story, which we can call as "Coffee Latte Analogy". My grandma made her own pre-mix of coffee latte using coffee, sugar, and milk powder. Her recipe stated for every premix she needs 1 table spoon (tbs) of coffee, 1 (tbs) of sugar, and 1 (tbs) of milk powder.

For simplification, she calls her pre-mix recipe as coffee:sugar:milk powder, as all composition are in tbs. This comparison can be called as (oversimplified) *dimensionless* variables. Because, all the ingredients share same measurement (tbs).

When I make my coffee exactly with pre-mix recipe, it can be called as (oversimplified) *intra-regime partials*, as all the parts come from recipe (*regime*).

After we got the recipe, we started to be creative and change the water temperature to boil the coffee latte, as she didn't put it in her notes. This can be called as (oversimplified) *inter-regime partials*, as we have the ingredients from the recipe (*dimensionless variables*), but we add some changes (*variables*) from outside the recipe.

One day, I made this recipe in my boyfriend's place. He has a fancy coffee maker. And instead of boiling like usual, I used the pre-mix recipe with this coffee machine. This can be called as (oversimplified) *inter-ensembles partials*, as we same (*contact variables)* ingredients from the recipe (*dimensionless variables*), but we do it in different house (*ensembles*).

Now, I would extend this logic into proper physics scenarios, with one of the simplest physic scenarios, determining period of oscillation of a pendulum.

## Basic Mathematical Concepts and Symbol Relevant to This Topic

Before going deep, I would like to remind 4 basic mathematical theorem and symbols.

1. The principle of Dimensional Homogeneity
   $A = B + C$
   The left and right side of =, should have same value. Therefore, $\mathrm{A}$ value is equivalent to the sum of $\mathrm{B}$ and $\mathrm{C}$.

2. π in this research is not the π of circle. π in here representing dimensionless number. Because π is dimensionless, 2π is $\mathrm{Constanta}\ (\mathrm{C})$.

3. ∝ is representing direct proportionality.
   $\mathrm{A} \propto \frac{1}{B}$, means when A increased, the B will be decreased.
   $\mathrm{A} \propto \mathrm{B}$, means when A increased, the B will be increased as well.

4. Conservation of Dimensions: Every term in a physically valid equation must have the same dimensions. You cannot add apples and oranges; you can only add variables with same unit of measurements.

## Early Theorem, Buckingham-*π Theorem*, and Halls' *Theorem*

1. *The Early Theorem* is applying mathematical relationships between variables.
   a. The most basic format of The Early Theorem is:
   b. $y = C.x^{a1}.x^{a2} \dots .x^{an}$ , with y is the variable we want to define, $\mathrm{C}$ is dimensional Constanta, $\mathrm{x}$ is independent variable, and a is exponent of $\mathrm{x}$.

2. *Buckingham π Theorem* gives us the theoretical basis for why we can determine variables without knowing the original physics formula. It gives us systematic method to determine how many dimensionless combination can be acquired from a

group of variables.
The most basic formats of Buckingham π Theorem is
$\pi = (n - r)$ , with Regime π comes from the number of variables (n) minus the number of basic dimensions (r).

3. Buckingham π Theorem only turn physical phenomenon in mathematical symbols. It does not tell us their physical roles nor it guarantees that each number contains only one variable that is not the basis. *Hall's Theorem* takes the result and gave us the definition of physical role of the variable in each regime. Hall's Theorem takes dimensional analysis from being a tool for modelling problems in engineering to a method for problem solving in artificial intelligence.

## Dimensionless Analysis Proof of Concept, using the formula of *Pendulum's Period of Oscillation*

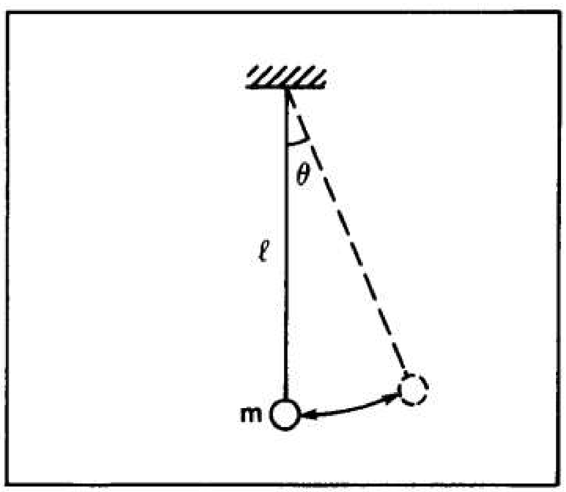


Fig. 1. A simple pendulum.

m = mass units [M] (say G-mass),
l = length units [L] (say CM),
g = acceleration units [$LT^{-2}$] (say CM per second squared),
θ = angle of the pendulum oscillation (no dimensions [ ]).
t = oscillation time (the time needed for one complete swing) [T] (say seconds)

1. The variables available in the Fig.1, can be sum up as:
$t = f(m, l, g, \theta)$

2. By inspection it is clear that mass only available on the right side of the equation, so we can omit the $\mathrm{m}$ from the right side as well:
   $\mathrm{t} = \mathrm{f}(\mathrm{l}, \mathrm{g}, \theta)$

3. $\theta$ is dimensionless, so it can be entered only as product, and therefore can be omitted from this equation. Which means, only time, length and gravitation are relevant in this calculations.
   $\mathrm{t} = \mathrm{f}(\mathrm{l}, \mathrm{g})$

4. Based on Buckingham π Theorem, we can determine there are 1 π regime, from:
   $\mathrm{n}$ = 3 [number of variables: T, L, g]
   $\mathrm{r}$ = 2 [number of basic dimensions: T, L. Because g is ($T/L^2$), this formula only have 2 dimensions T and L].
   $\pi = (\mathrm{n} - \mathrm{r}) = (3 - 2) = 1$. Therefore, in this formula, there is only 1 regime(π).

5. From The Early Theorem, Step 3 and Step 4, we can get the formula:
   $\pi = \mathrm{t} \,.\, l^a . \, g^b$, from Buckingham π Theorem in step 4, we can know the regime is 1, therefore:
   $\pi = \mathrm{t} \,.\, l^a . \, g^b$

6. We can get this replace the formula in step 4 and 5 with its respective symbol.
   $\pi = \mathrm{t} \,.\, l^a . \, g^b$ . Remember from step 4, that g is $\mathrm{T}/\mathrm{L}^2$, so it can be changed into
   $L^0 T^0 = \mathrm{T}. \mathrm{L}^a. (\frac{L}{T^2})^b$ , which can be further simplified into:
   $L^0 T^0 = T. L^a. L^b. T^{-2b}$, which means:
   $L^0 = L^{a+b}$ and $T^0 = T^{1-2b}$.
   So from T, we can get: $0 = (1 - 2b)$, which means b = -1/2.
   And from L, we can get: $0 = (a + b)$, which means $(a = -2b)$, which resulting a = ½

7. If we replace $a$ and $b$ from $\pi = \mathrm{t} \,.\, l^a . \, g^b$ (Step 5), into ½ and -1/2 (Step 6), we can get:
   $\pi = \mathrm{t} \,.\, l^{1/2} . \, g^{-1/2}$, which equivalent to :
   $$\pi = \mathrm{t} \,.\, \sqrt{\frac{l}{g}}$$
   From this notation, we can summarize that $t \propto \sqrt{\frac{l}{g}}$

8. We can proof the formula, by replace it with its symbol.
   $T \propto \sqrt{\frac{L}{\frac{L}{T^2}}}$ , By cross out the L, we can get $T \propto \sqrt{T^2}$ . Therefore, $T = T$

## The $\pi$-calculus

- $\Pi$-calculus is conceptual machine for reasoning the dimensions numbers. Machine in here is not in the sense of computer, but logical thinking framework. $\Pi$-calculus provides a formal framework for qualitative reasoning.
- *Basis* is the set of variables that repeated in each $\Pi$. Basis plays significant role in constructions of regimes. All basis variables (r), should accomplished these criteria: (1) Every dimension that occurs in the dimensional representation of the n variables characterizing the system must occur in the dimensional representation of one or more basis variables. (2) The dimensional representations of the basis variables should be linearly independent.
- Each dimensionless number ($\Pi$), refers to a particular physical aspect of the system, which we called as *regime*. Regime is the result of dimensionless analysis.
- A collection of regimes is called *ensemble*.
- If same variables available in both $\Pi_1$ and $\Pi_2$, it is called as *contact variables* or *pivot*.
- *Coupling* refers to the interconnectedness between different components or subsystems within a larger system. Coupling occurs when two or more "*ensembles*" (*sets of regimes*) share the same variables (*contact variables*).
- *Component / Subsystem* is a separate part or functional unit of a larger device or system. Each component has its own physical function and can be modeled separately using one or more "ensembles" (dimensionless sets of regimes).
  Instead of trying to model the entire complex system at once, we can break the system into multiple subsystems/components to analyze each component separately and then connect the models through coupling variables. The purpose of breaking a system down into components/subsystems is to simplify the analysis.
- Relationship between regimes $\Pi$ might fall into one of this type:
  - *If $Z_j$ is in the basis and occurs in $\Pi_i$, then use intra-regime partials*
    *Intra-regime analysis*: Analysis within a regime, for examining how the variables within a regime related to each other
  - *If $Z_j$ is in the basis but not in $\Pi_i$, then reason using chains of inter-regime partials*
    *Inter-regime analysis*: Analysis across regimes, to see how different regimes are related to one another through contact variables
  - *If $Z_j$ is not in the basis, then use appropriate inter-regime partial linking $\Pi_i$ and $\Pi_j$*
    *Inter-ensemble analysis*: Analysis across ensembles, to reason about the behavior of a device or system consisting of coupled components or subsystems.

## How π-calculus Works, using The Pressure Regulator's Mechanism

The objective of this example (Fig.2) is to demonstrate how we reason across coupled ensembles.

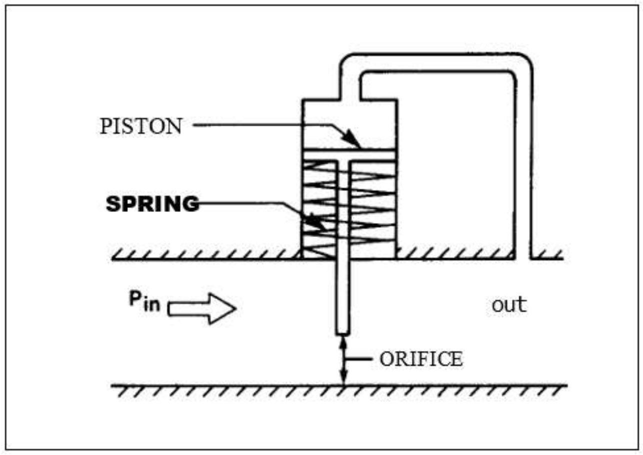


Fig. 2. The pressure regulator.

Pressure regulators are being used to maintain a constant pressure at the output. To simplify the process, we can say it consist of 2 sides, a pipe with an orifice and a spring valve assembly. Therefore, we can modelled pressure regulator as inter-ensemble model with 2 components connected with contact variables (Fig 3).

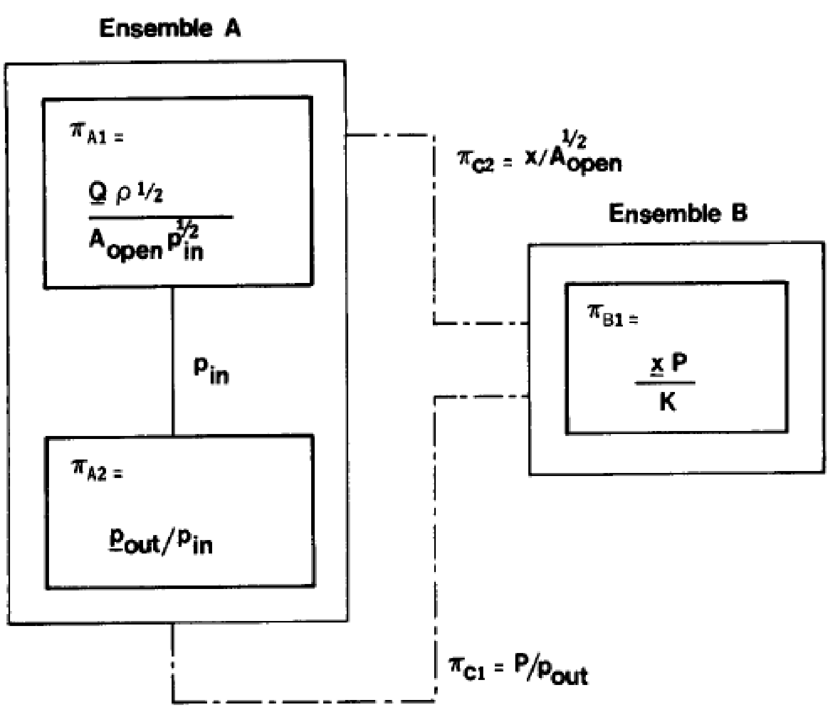


Fig.3 . Inter-ensemble analysis of the pressure regulator.

**First component is a pipe with an orifice**, which is a familiar system in fluid mechanics. The pertinent quantities are as follows:

| | | |
|---|---|---|
| outlet pressure | $p_{out}$ | $[ML^{-1}T^{-2}]$, |
| orifice flowrate | $Q$ | $[L^3T^{-1}]$, |
| inlet pressure | $p_{in}$ | $[ML^{-1}T^{-2}]$, |
| orifice opening | $A_{open}$ | $[L^2]$, |
| fluid density | $\rho$ | $[ML^{-3}]$. |

Using $(p_{in}, A_{open}, \rho)$ as the basis we can obtain the following $\Pi$s

$$\Pi_{A1} = \frac{Q\rho^{1/2}}{A_{open}p_{in}^{1/2}}, \qquad \Pi_{A2} = \frac{p_{out}}{p_{in}}.$$

From this ensemble the intra-regime partials $\partial Q/\partial p_{in}$, $\partial Q/\partial A_{open}$, and $\partial p_{out}/\partial p_{in}$ are all positive. Hence the inter-regime partial $[\partial p_{out}/\partial Q]^{p_{in}}$ is also positive. If the input pressure $p_{in}$ increases, $\Delta p_{in} > 0$, then from the intra-regime partials we can infer that the flowrate $Q$ and outlet pressure $p_{out}$ will increase. Similarly if the orifice opening $A_{open}$ decreases then we can conclude that $Q$ will increase. Lastly, since the inter-regime partial $[\partial p_{out}/\partial q]^{p_{in}}$ is positive, an increase in $Q$ will lead to an increase in $p_{out}$.

**The second component is spring valve ensemble**. In the spring valve, pressure is applied to a piston that is connected to a spring. The quantities (variables) that characterize this system are:

| | | |
|---|---|---|
| spring displacement | $x$ | $[L]$, |
| pressure | $P$ | $[ML^{-1}T^{-2}]$, |
| spring constant | $K$ | $[MT^{-2}]$. |

From these quantities, we can obtains $\Pi_{B1} = xP/K$.

There are 2 basis here, P and K. There are 3 dimensions that appear (L,M,T), but the rank in dimensional matrix is only 2. Because, we combined $[MT^{-2}]$ together, because it appears both in P and K, with same exponents.

A model of pressure regulator is coupled of a pipe with an orifice ensemble and spring valve ensemble. The information needed for coupling the ensembles comes in two flavors, topology and geometric constraints. Coupling regimes are closely tied to the connections between components and thus are ratios of pertinent quantities with identical dimensionality modulo exponent. In this example there are two coupling regimes:

$$\Pi_{C1} = P/p_{out}, \qquad \Pi_{C2} = x/A_{open}^{1/2}$$

The regime $\Pi_{C1}$ comes from the connection that transmits the outlet pressure in the pipe to the piston in the spring valve assembly. Thus $\partial P/\partial p_{out}$ is positive; so an increase in $p_{out}$ leads to an increase in $P$. The second coupling regime, $\Pi_{C2}$, encodes the geometric constraint that motion of the piston affects the orifice opening; more specifically as the spring is compressed, the orifice reduces. This behavior is captured by the partial $\partial x/\partial A_{open}$ which is positive.

As the goal of pressure regulator is to maintain outlet pressure $P_{out}$ at constant value $P_*$. The key point is that the system has an **active feedback loop** that corrects the initial change to return to the desired state. This is the whole purpose of the regulator.

Based on the logic we had so far, we can sum out that pressure regulator will exhibit the following behaviors:

> An increase in $p_{in}$ leads to an increase in $p_{out}$ (from $\Pi_{A2}$). This increase in $p_{out}$ leads to an increase in $P$ in the spring valve ensemble (from coupling regime $\Pi_{C1}$). The increase in $P$ causes $x$ to decrease (from $\Pi_{B1}$). This time using the coupling regime $\Pi_{C2}$, decrease in $x$ leads to a decrease in $A_{open}$. Now in the pipe orifice ensemble, this decrease in $A_{open}$ leads to a decrease in $Q$. Finally through the inter-regime partial $[\partial p_{out}/\partial Q]^{p_{in}}$, the decrease in $Q$ leads to a decrease in $p_{out}$. Thus we have derived the feedback behavior, i.e. an increase in $p_{out}$ eventually leads to a decrease in $p_{out}$. This might seem like a contradiction but is not; we are talking about changes in $p_{out}$ at temporally distinct points.[15]

## Regimes as Representation:

How the theorems, we discussed above can be used when building programming algorithm?

1. **Regime as physical process**: Dimensionless numbers represent physical processes. A dimensionless number like $\pi = t\,.\sqrt{\frac{g}{l}}$ (in pendulum's period of oscillation) is not just a mathematical number/equation, but has physical meaning. In the case of a pendulum, this number represents the relationship between time, length, and gravity that determines the oscillation.
2. **In-Principle Reducibility**: Dimensional analysis does not require that numerical information be substituted with nonnumerical, qualitative information. Instead, an ensemble of regimes with the appropriately chosen variables contains all the physical information that a set of laws and geometrical constraints contain.
3. **Conservation of dimensions**: Conservation of Dimensions ensures that each formula generated by this method is dimensionally consistent, a fundamental requirement for physical validity. Conservation of dimensions is to ensure that the results of dimensional analysis able properly represented a real physical process.
4. **Power vs Generality**: Dimensional analysis is beneficial to allow programmers enhance a system without requiring depth understanding of physical laws and formulas. However, because of this generality, this method cannot provide numerical values and can only provide qualitative results (positive or negative signs) or proportionality.

# Flowchart of Simple Pressure Regulator Logic based on Dimensional Analysis in Python

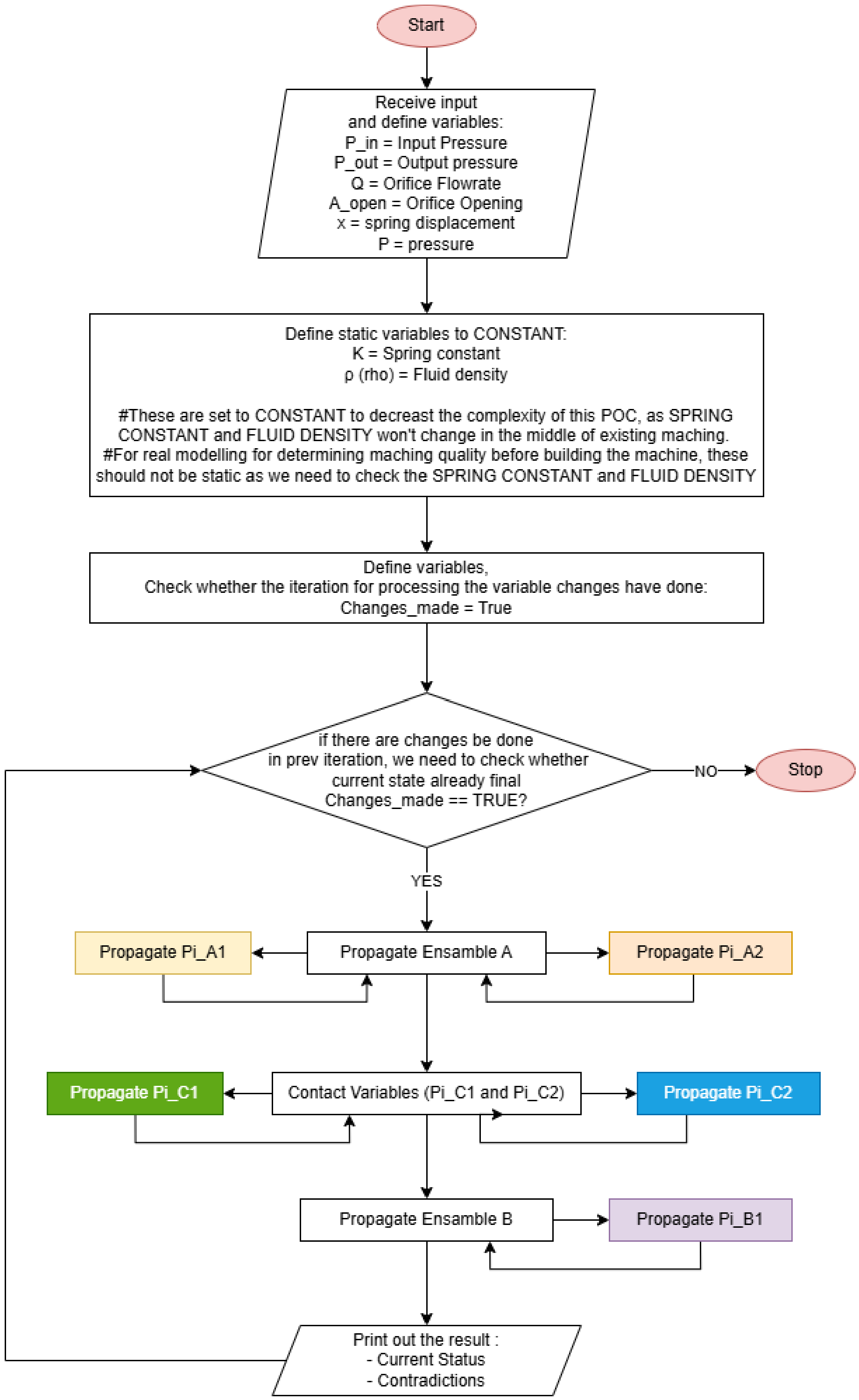


Fig4. Algorithm of Pressure Regulator's Mechanism, inherited from π-calculus

Fig4 shows how the algorithm that we can build to simulate the mechanism of pressure regulator's algorithm, by using π-calculus, without prior knowledge on how to operate it. From "How π-calculus Works, using The Pressure Regulator's Mechanism", we able to build the flow of pressure regulator, which shown in Fig 3.

Fig3 consisted of 2 ensembles: Ensembles_A and Ensembles_B. Ensembles_A consisted of 2 regimes ($\pi_{a1}$ and $\pi_{a2}$). While, Ensembles_B consisted of 1 regimes ($\pi_{b1}$). Ensembles_A and Ensembles_B are connected by pivot variables, $\pi_{c1}$ and $\pi_{c2}$.

Therefore, we can start the program by receiving all the value (INCREASE, DECREASE, CONSTANT, UNKNOWN) of the qualitative variables ($\pi_{in}$, $\pi_{out}$, Q, $A_{open}$, x, P). Variable that representing spring constant (K) and Fluid density (rho/ρ) are hard-coded into CONSTANT to decrease the complexity of this program. Because, a machine spring and fluid density would not suddenly change in the middle of the process. But, in the future, when we used this system to model a machine prior building it, of course we need to make it input-able as well. Additional, UNKNOWN value is added to cover the scenario where user did not pre-defined a variable.

Then, the program will iterate starting from Ensembles_A, which value will be retracted from $\pi_{a1}$ (Propagate Pi_A1 – Fig.7) and $\pi_{a2}$ (Propagate Pi_A2 – Fig.8). Then, we will retract value of pivot variables $\pi_{c1}$ (Propagate Pi_C1 – Fig.10) and $\pi_{C2}$ (Propagate Pi_C2 – Fig.11). With the value from pivot variables, we can connect Ensembles_A with Ensembles_B, which value came from $\pi_{b1}$ (Propagate Pi_B1 – Fig.9).

The algorithm of each propagation discussed above are listed here:

**Propagate Pi_A1 (Fig.7)**
This function evaluates the relationship between flow rate (Q), fluid density (rho), valve opening area ($A_{open}$), and inlet pressure ($P_{in}$). The numerator variables Q and rho (But, we have made rho static in CONSTANT, so only Q has power to change the value) increase the value of the regime, while the denominator variables $A_{open}$ and $P_{in}$ reduce it. For example, when $P_{in}$ increases while others remain constant, the overall regime value decreases because $P_{in}$ appears in the denominator.

**Propagate Pi_A2 (Fig.8)**
This function links outlet pressure ($P_{out}$) and inlet pressure ($P_{in}$) through a division relationship. If both pressures change in the same direction with the same magnitude, the regime becomes constant. If only one changes, the result reflects the direction of that change relative to the other.

**Propagate Pi_B1 (Fig.9)**

This function describes the spring valve subsystem by relating displacement ($x$), internal pressure ($P$), and spring constant ($K$). Since $K$ is treated as constant in the current program, the regime outcome depends mainly on the qualitative changes of $x$ and $P$. For example, if $P$ increases, the force on the spring increases, resulting in a change in $x$.

**Propagate Pi_C1 (Fig.10)**

This coupling regime connects the pipe ensemble (Ensemble_A) and the spring ensemble (Ensemble_B) by using pivot variables:pressure ($P$) with outlet pressure ($P_{out}$). A change in one of these pressures is directly propagated to the other to ensure consistency between both ensembles.

**Propagate Pi_C2 (Fig.11)**

This coupling regime connects the Ensemble_A and Ensemble_B, by employing pivot variables:spring displacement ($x$) with the valve opening area ($A_{open}$). When, $A_{open}$ and $x$ are inversely proportionate.

For future enhancement or changing in this coding to accommodate other mechanism, we only need to change propagate functions and the sequences that the propagate functions been called in solve_pressure_regulator.

To accommodate, the proportionality between nominator and denominator, I have also prepared function:determine_product_status (Fig.6) (to find the value of 2 qualitative variables with multiplication relationship between them) and function:determine_division_status (Fig5) (to find the value of 2 qualitative variables with division relationship between them). We can enhanced this in the future by changing the variables (status1, status2) received by function:determine_product_status and function:determine_division_status, into array or dictionary, instead.

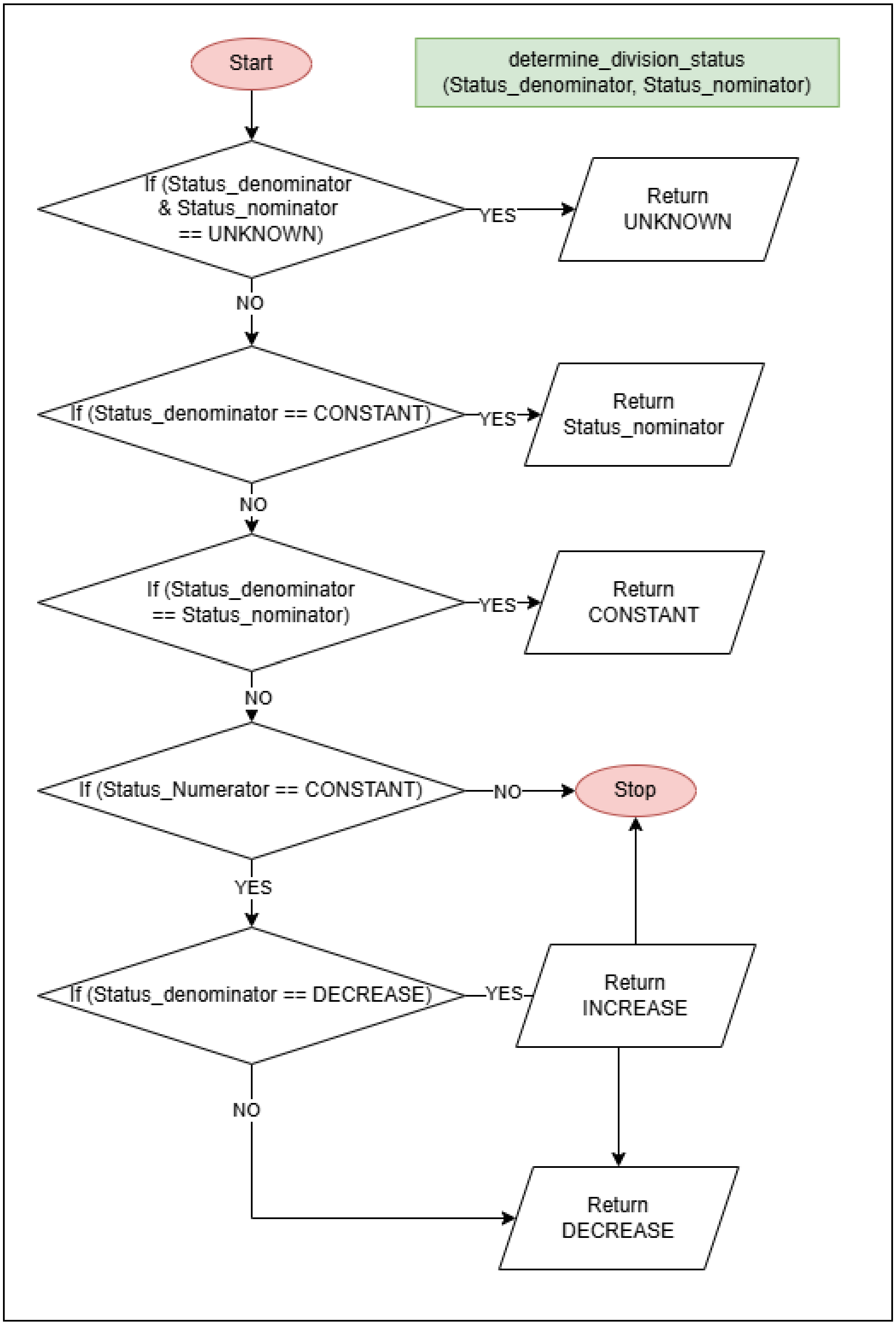


Fig5.Division Function

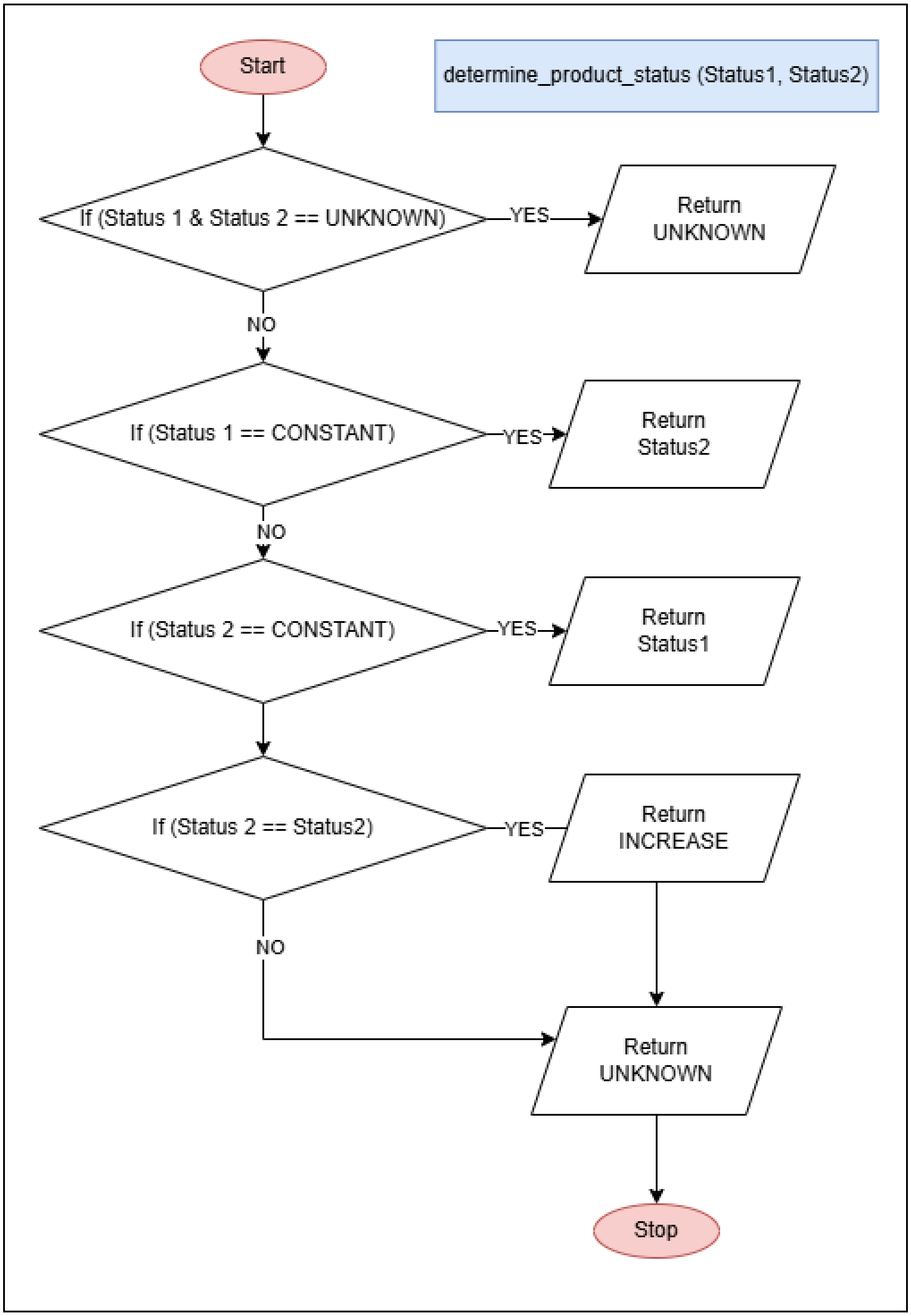


Fig6.Multiplication Function

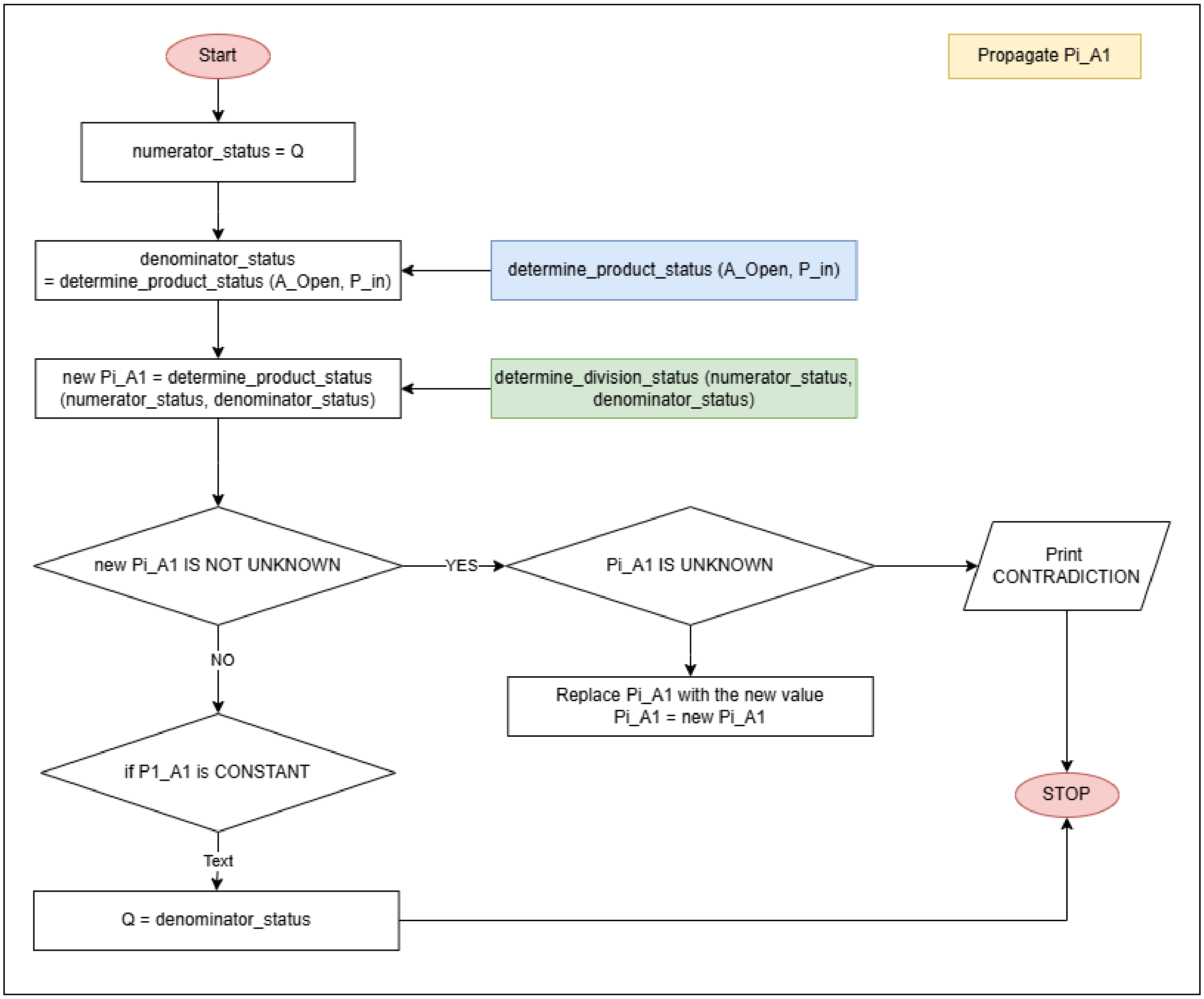
Start
Propagate Pi_A1
numerator_status = Q
denominator_status
= determine_product_status (A_Open, P_in)
determine_product_status (A_Open, P_in)
new Pi_A1 = determine_product_status
(numerator_status, denominator_status)
determine_division_status (numerator_status,
denominator_status)
new Pi_A1 IS NOT UNKNOWN
YES
Pi_A1 IS UNKNOWN
Print
CONTRADICTION
NO
Replace Pi_A1 with the new value
Pi_A1 = new Pi_A1
if P1_A1 is CONSTANT
STOP
Text
Q = denominator_status

Fig7.Pi_A1

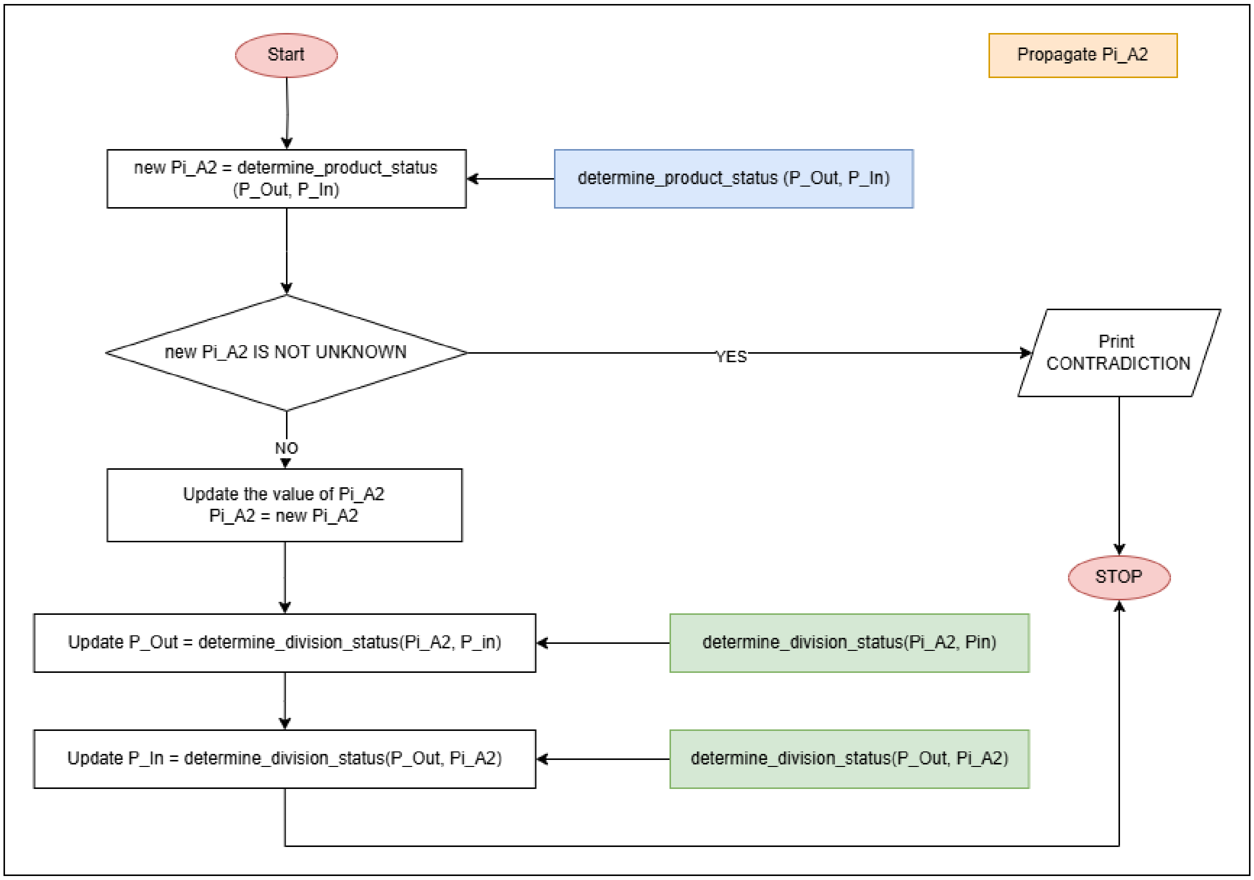
Start
Propagate Pi_A2
new Pi_A2 = determine_product_status (P_Out, P_In)
determine_product_status (P_Out, P_In)
new Pi_A2 IS NOT UNKNOWN
YES
Print CONTRADICTION
NO
Update the value of Pi_A2
Pi_A2 = new Pi_A2
STOP
Update P_Out = determine_division_status(Pi_A2, P_in)
determine_division_status(Pi_A2, Pin)
Update P_In = determine_division_status(P_Out, Pi_A2)
determine_division_status(P_Out, Pi_A2)

Fig8.Pi_A2

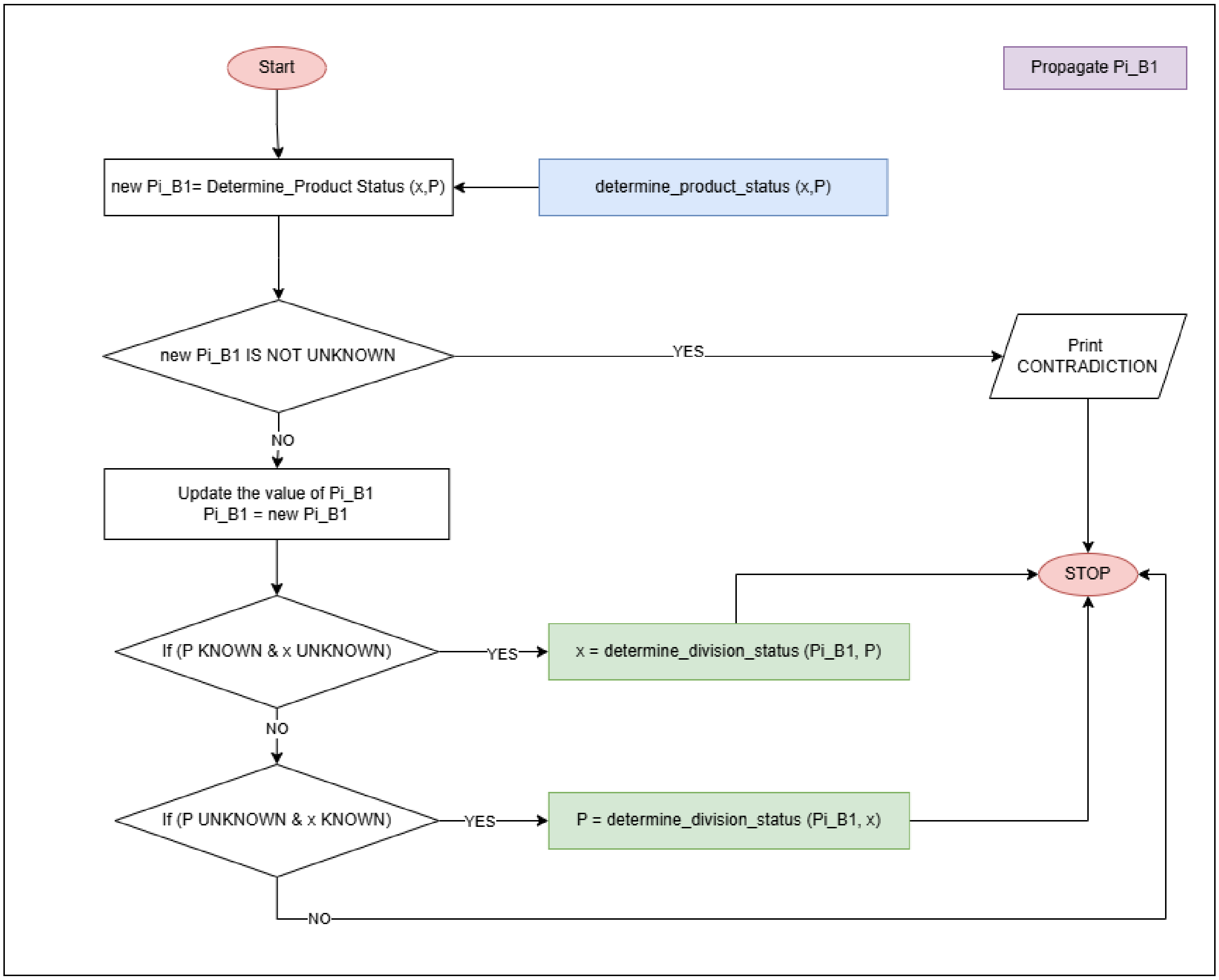
Start
Propagate Pi_B1
new Pi_B1= Determine_Product Status (x,P)
determine_product_status (x,P)
new Pi_B1 IS NOT UNKNOWN
YES
Print CONTRADICTION
NO
Update the value of Pi_B1
Pi_B1 = new Pi_B1
STOP
If (P KNOWN & x UNKNOWN)
YES
x = determine_division_status (Pi_B1, P)
NO
If (P UNKNOWN & x KNOWN)
YES
P = determine_division_status (Pi_B1, x)
NO

Fig9.Pi_B1

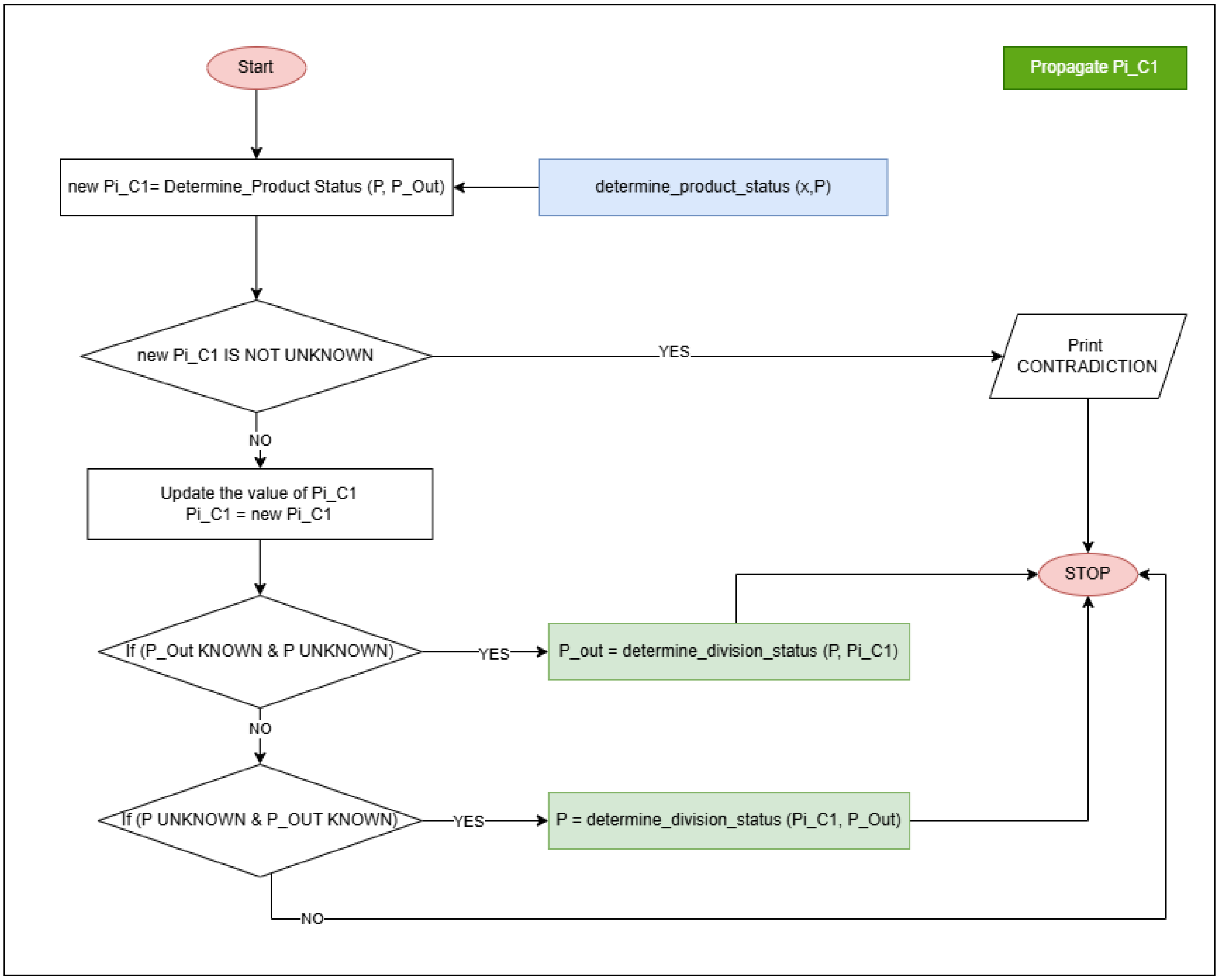
Propagate Pi_C1
Start
new Pi_C1= Determine_Product Status (P, P_Out)
determine_product_status (x,P)
new Pi_C1 IS NOT UNKNOWN
YES
Print CONTRADICTION
NO
Update the value of Pi_C1
Pi_C1 = new Pi_C1
STOP
If (P_Out KNOWN & P UNKNOWN)
YES
P_out = determine_division_status (P, Pi_C1)
NO
If (P UNKNOWN & P_OUT KNOWN)
YES
P = determine_division_status (Pi_C1, P_Out)
NO

Fig10.Pi_C1

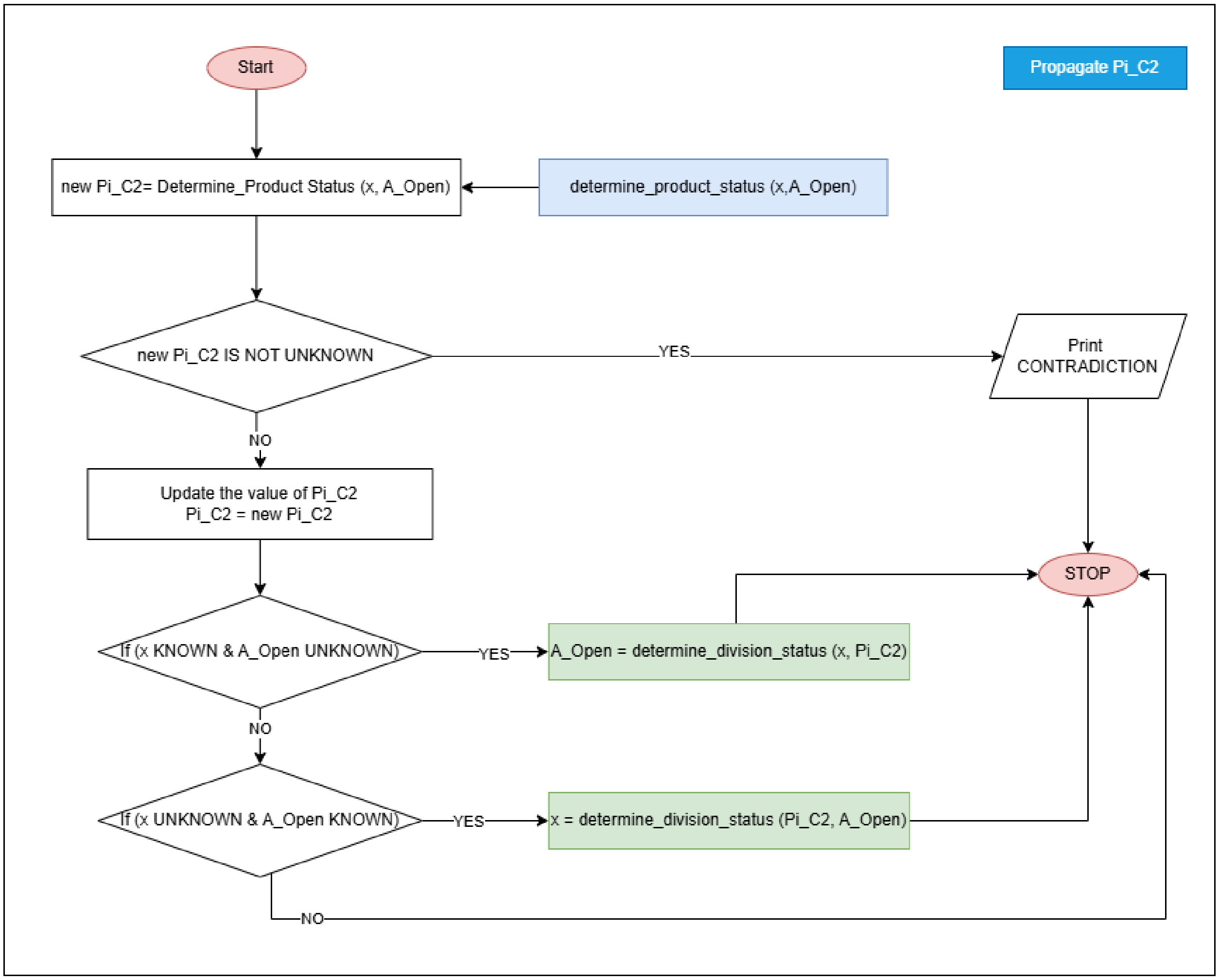
Start
Propagate Pi_C2
new Pi_C2= Determine_Product Status (x, A_Open)
determine_product_status (x,A_Open)
new Pi_C2 IS NOT UNKNOWN
YES
Print
CONTRADICTION
NO
Update the value of Pi_C2
Pi_C2 = new Pi_C2
STOP
If (x KNOWN & A_Open UNKNOWN)
YES
A_Open = determine_division_status (x, Pi_C2)
NO
If (x UNKNOWN & A_Open KNOWN)
YES
x = determine_division_status (Pi_C2, A_Open)
NO

Fig11.Pi_C2

## Functions Implemented

**1. Input Handling**

- Accepts qualitative values (INCREASE, DECREASE, CONSTANT, UNKNOWN) for system variables: $P_{in}$, $P_{out}$, $Q$, $A_{open}$, $x$, $P$
- Pre-sets **spring constant (K)** and **fluid density (ρ)** to CONSTANT (hard-coded).

**2. Propagate Functions (Ensemble A – Pipe system)**

- **Propagate Pi_A1** – Fig.7: relates $Q$, $\rho$, $A_{open}$, $P_{in}$.
- **Propagate Pi_A2** – Fig.8: relates $P_{out}$, $P_{in}$.

**3. Propagate Functions (Coupling Regimes / Pivot variables)**

- **Propagate Pi_C1** – Fig.10: couples $P \leftrightarrow P_{out}$.
- **Propagate Pi_C2** – Fig.11: couples $x \leftrightarrow Ao_{pen}$.

**4. Propagate Functions (Ensemble B – Spring system)**

- **Propagate Pi_B1** – Fig.1: relates $x$, $P$, $K$.

**5. Supporting Functions for Qualitative Algebra**

- **determine_product_status** – Fig.6: evaluates multiplication relationship of two qualitative states.
- **determine_division_status** – Fig.5: evaluates division relationship of two qualitative states.

**6. Main Algorithm (solve_pressure_regulator)**

- Iterates through Ensemble A, pivot variables, and Ensemble B.
- Propagates qualitative changes step-by-step until results stabilize or contradictions are found.

## Future Enhancements

**1. Input Flexibility**

- Make spring constant (K) and fluid density (ρ) user-input instead of fixed CONSTANT.
- Allow different initial conditions for better system modeling.
- Currently only allow positive numbers in all variables, allow flexibility.

**2. Extended Variable Handling**

- Generalize determine_product_status and determine_division_status to handle arrays/dictionaries of variables.
- Support proportionality among multiple numerator and denominator variables.

**3. Enhanced Propagation**

- Change this code into other machines (e.g., thermal, electrical) by creating new propagate functions.
- Extend pivot logic to connect more than two ensembles.
- Introduce **enhanced propagation rules** so the system can check contradictions not only on the π-variables (Pi-*), but also on the core physical variables (e.g. $P_{in}$, $P_{out}$, $Q$, $A_{open}$, $x$, $P$).
  This would allow the solver to detect inconsistencies earlier and provide richer diagnostics about the system's behavior.

**4. Contradiction Tracking**

- Store and report a list of contradictions instead of stopping at the first conflict.

- Provide explanations of contradiction causes.

**5. Simulation Expansion**

- Move beyond static qualitative reasoning to handle temporal changes (dynamic feedback loops).
- Allow iteration to show transient vs steady-state qualitative behavior.

## DEMO

```
PS D:\SE4GD\JobSearch\CoverLetter\tempere -maysere> python.exe .\code3.py
Select the scenario you want to run:
1. Scenario 1: P_out constant, P_in increasing
2. Scenario 2: P_in constant, P_out decreasing
3. Scenario 3: P_in increasing, P_out increasing
4. Scenario 4: P_in increasing, P_out is not determined
5. Scenario 5: P_in constant, P_out increasing
6. Manual Input
Make your choice (1/2/3/4/5/6): 6
Enter the status for the following variables (I, D, C, U)
Status of P_in: I
Status of P_out: U
Status of Q: C
Status of A_open: I
Status of x: U
Status of P:
--- INITIAL STATUS ---
{'P_in': 'Increased', 'P_out': 'U', 'Q': 'Constant', 'A_open': 'Increased', 'x': 'U', 'P': 'U', 'Pi_A1': 'Constant', '
Pi_A2': 'U', 'Pi_B1': 'U', 'Pi_C1': 'U', 'Pi_C2': 'U'}
--------------------
Iteration 1:
CONTRADICTION FOUND ON PI_A1: User defined PI_A1 IS 'Constant'. It is CONTRADICTED the calculation result 'Decreased'.
{'P_in': 'Increased', 'P_out': 'U', 'Q': 'Constant', 'A_open': 'Increased', 'x': 'Increased', 'P': 'U', 'Pi_A1': 'Cons
tant', 'Pi_A2': 'U', 'Pi_B1': 'U', 'Pi_C1': 'U', 'Pi_C2': 'U'}
Iteration 2:
CONTRADICTION FOUND ON PI_A1: User defined PI_A1 IS 'Constant'. It is CONTRADICTED the calculation result 'Decreased'.
{'P_in': 'Increased', 'P_out': 'U', 'Q': 'Constant', 'A_open': 'Increased', 'x': 'Increased', 'P': 'U', 'Pi_A1': 'Cons
tant', 'Pi_A2': 'U', 'Pi_B1': 'U', 'Pi_C1': 'U', 'Pi_C2': 'Constant'}
Iteration 3:
CONTRADICTION FOUND ON PI_A1: User defined PI_A1 IS 'Constant'. It is CONTRADICTED the calculation result 'Decreased'.
{'P_in': 'Increased', 'P_out': 'U', 'Q': 'Constant', 'A_open': 'Increased', 'x': 'Increased', 'P': 'U', 'Pi_A1': 'Cons
tant', 'Pi_A2': 'U', 'Pi_B1': 'U', 'Pi_C1': 'U', 'Pi_C2': 'Constant'}
```

Fig12.Example of running script

Fig12 shows running sample of the attached code.

## Acknowledgements

I would like to sincerely thank Professor Eric Coatanéa for providing the opportunity to work on this assignment and for sharing the reference materials that inspired this study. I am also grateful to my pen-pal (Eepy), who studied mechanical engineering and working as technical sales, for discussing with me the concepts I initially found difficult and for helping me better understand the qualitative reasoning framework. This work has not only been a valuable academic exercise but also a personal opportunity to reconnect and learn collaboratively.

## Proof of Manual Work

All reasoning steps, derivations, and algorithm sketches presented in this summary were developed manually. I prepared handwritten notes and draft calculations throughout the process before typing the final version, while diagrams were drawn in draw.io (Fig.14). A photo of my handwritten notes is included as evidence that this work was carried out by hand and not generated by an automated language model (Fig.13).

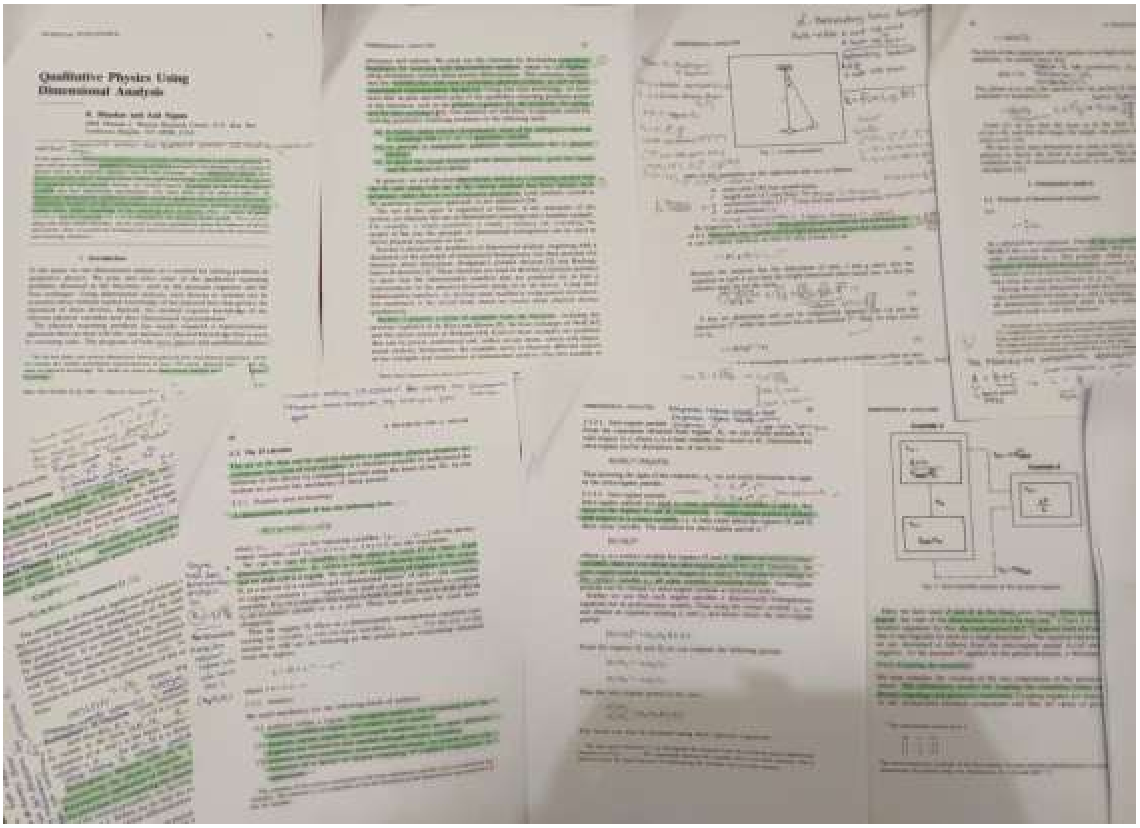


Fig13. Some of the handwritten notes

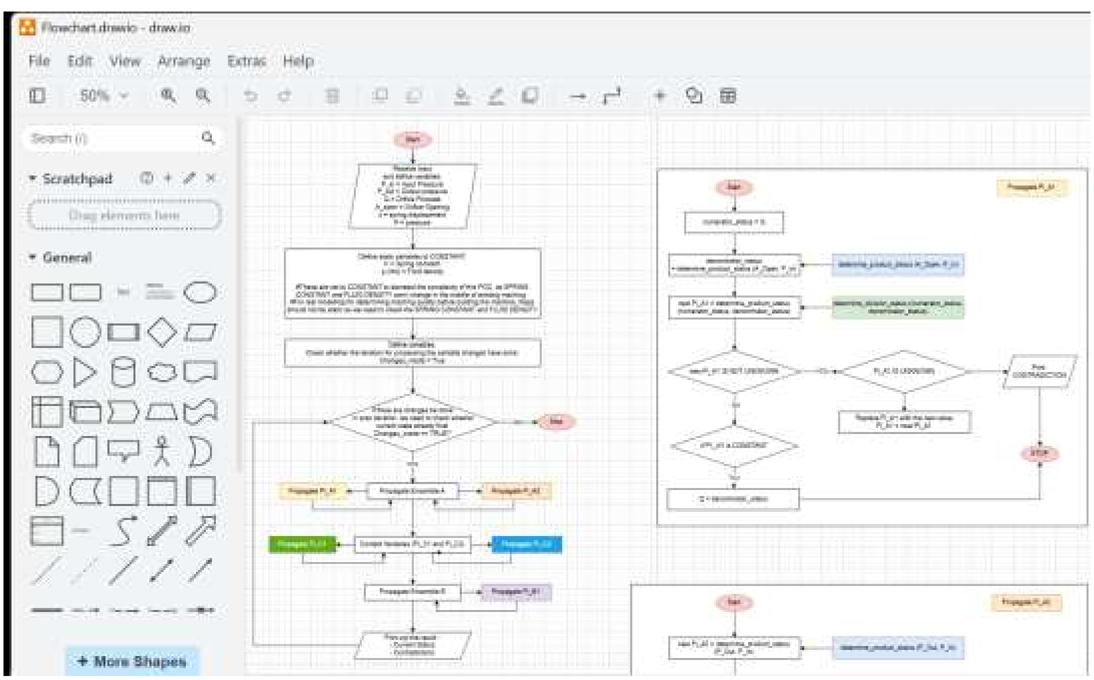


Fig14. Screenshot of draw.io diagrams