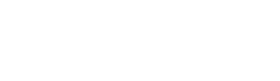
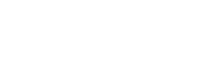
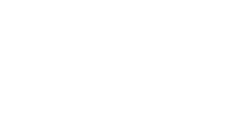
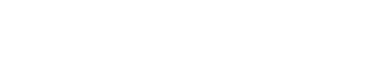
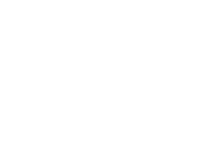
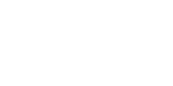
**Task No 01**

* **Block** **diagram** **of** **the** **feedback** **control** **system**

1. A general Block Diagram Stream 1 Composition



|  |  |
| --- | --- |
| Blending Tank |  |
|  |



|  |  |
| --- | --- |
| Controlled Variable Signal | Composition Sensor/transmitter |
|  |

SP  error Controller

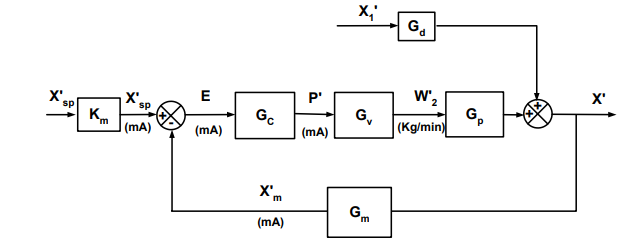
Controller Output

Stream 2 Actuator (Valve) Flow Rate

Output Composition

Controlled Variable Measurement

2. Block Diagram with the transfer Functions and signal types



**b)** **Transfer** **function** **for** **each** **block.**

𝑊 (𝑡), 𝑊𝑡), 𝑋1(𝑡), 𝑋(𝑡) 𝑎𝑟𝑒 𝑡𝑖𝑚𝑒 𝑑𝑒𝑝𝑒𝑛𝑑𝑎𝑛𝑡 𝑣𝑎𝑟𝑖𝑎𝑏𝑙𝑒𝑠.

(

2

𝑊 , 𝑋2 𝑎𝑟𝑒 𝑡𝑖𝑚𝑒 𝑖𝑛𝑑𝑒𝑝𝑒𝑛𝑑𝑎𝑛𝑡 𝑣𝑎𝑟𝑖𝑎𝑏𝑙𝑒𝑠.

1

1. Mass Balance (Mixture)

𝑑𝑚 = 𝑊 + 𝑊 (𝑡) − 𝑊𝑡) → [𝑑𝑚 = 0 𝑓𝑜𝑟 𝑐𝑜𝑛𝑠𝑡𝑎𝑛𝑡 𝑣𝑜𝑙𝑢𝑚𝑒 𝑎𝑛𝑑 𝑑𝑒𝑛𝑠𝑖𝑡𝑦] → 𝑊 + 𝑊 (𝑡) − 𝑊𝑡) = 0

(

𝑑𝑡 𝑑𝑡

1 2 1 2 (

2. Mass balance (Solute)

Taking the right-hand side as a function in the time dependent variables: 𝑓(𝑋1(𝑡),𝑊 (𝑡), 𝑋(𝑡)), we

2

linearize it about steady state using a first order Taylor series expansion:

𝑓(𝑋1(𝑡), 2(𝑡),𝑋(𝑡)) 𝑓(𝑋 ,𝑊 ,𝑋) + 𝜕𝑓 | 𝑋′1(𝑡) + 𝜕𝑓 | 𝑊′2(𝑡) + 𝜕𝑓 | 𝑋′(𝑡) 1 𝑡 𝑋1,𝑊 ,𝑋 2 𝑡 𝑋1,𝑊 ,𝑋 𝑡 𝑋1,𝑊 ,𝑋

1 2

̅ ̅ ̅

𝑑𝑋 𝑑𝑊 𝑑𝑋

( ) ( ) ( )

2 2 2

̅ ̅ ̅ ̅ ̅ ̅ ̅ ̅ ̅

Where 𝑋′1(𝑡),𝑊′2(𝑡),𝑋′(𝑡) 𝑎𝑟𝑒 𝑡ℎ𝑒 𝑑𝑒𝑣𝑖𝑎𝑡𝑖𝑜𝑛𝑠 𝑓𝑟𝑜𝑚 𝑠𝑡𝑒𝑎𝑑𝑦 𝑠𝑡𝑎𝑡𝑒 𝑎𝑡 𝑎 𝑡𝑖𝑚𝑒 = 𝑡

We define the deviation variables for our time dependent process variables:

𝑋′1(𝑡) = 𝑋1(𝑡) − 𝑋1 𝑊′2(𝑡) = 𝑊 (𝑡) − 𝑊 𝑋′(𝑡) = 𝑋(𝑡) − 𝑋

̅

̅

2 2

̅

We perform the derivations to solve the linear expansion:

𝑓(𝑋 ,𝑊 ,𝑋) = 𝑊 𝑋 + 𝑊 𝑋2 − (𝑊 + 𝑊 )𝑋 = 𝜌𝑉 𝑑𝑋 = 0

̅

𝑑𝑡

1 2 1 1 2 1 2

̅ ̅ ̅ ̅ ̅ ̅ ̅

2

1

𝑑𝑋

2

̅ ̅ ̅

̅

𝑑𝑊

2

̅ ̅ ̅

̅

1 2

𝑑𝑋

2

̅ ̅ ̅

𝜕𝑓 | = 𝑊 1(𝑡) 𝑋1,𝑊 ,𝑋

𝜕𝑓 | = 𝑋2 − 𝑋 2(𝑡) 𝑋1,𝑊 ,𝑋

𝜕𝑓 | = −𝑊 − 𝑊 (𝑡) 𝑋1,𝑊 ,𝑋

The left-hand side of the equation can be rewritten in terms of derivation variables as follows:

𝑑𝑋 𝑑(𝑋′ + 𝑋) 𝑑𝑋′ ̅ 𝑑𝑋′ 𝜌𝑉 𝑑𝑡 = 𝜌𝑉 𝑑𝑡 = 𝜌𝑉 [ 𝑑𝑡 + 𝑑𝑡 ] = 𝜌𝑉 𝑑𝑡

̅

𝑑𝑋

𝑡 𝑡 𝑡 𝑡

( ) ( ) ( ) ( )

Combining the side derivations and substituting them in the expansion and the solute balance equation:

𝑑𝑋′

𝑡

( )

( )

′

̅ ̅

1 1 2

𝜌𝑉 𝑑𝑡 = 𝑊 𝑋′1(𝑡) + 𝑋2 − 𝑋 𝑊 2(𝑡) − (𝑊 + 𝑊 )𝑋′(𝑡)

Applying Laplace Transform:

′

𝜌𝑉(𝑠𝑋′(𝑠) − 𝑋(0)) = 𝑊 𝑋′1(𝑠) + (𝑋2 − 𝑋)𝑊′2(𝑠) − (𝑊 + 𝑊 )𝑋′(𝑠)

̅ ̅

1 1 2

(𝜌𝑉𝑠+𝑊 + 𝑊 )𝑋′(𝑠) = 𝑊 𝑋′1(𝑠) + (𝑋2 − 𝑋)𝑊′2(𝑠)

1 2 1

̅ ̅

We then can write transfer functions for the response of the controlled variable to changes in the load variable and the manipulated variable:

1. Load/Output Transfer Function: (by setting the manipulated variable to a constant)

(𝜌𝑉𝑠+𝑊 + 𝑊 )𝑋′(𝑠) = 𝑊 𝑋′1(𝑠) + (𝑋2 − 𝑋)𝑊′2(𝑠)

1 2 1

̅ ̅

(𝜌𝑉𝑠+𝑊 + 𝑊 )𝑋′(𝑠) = 𝑊 𝑋′1(𝑠)

̅

1 2 1

𝑊 ′

1

̅

1 2

1

𝑊 + 𝑊

̅

1 2

𝜌𝑉

̅

1 2

𝑊 𝐾

( )

𝑋

𝐺𝑑𝑣 = 𝑋′1 𝑠) = (𝜌𝑉𝑠 + 𝑊 + 𝑊 ) = (𝑊 + 𝑊 𝑠 + 1) = 𝑠 𝑑𝑣1

+

(

𝑊 𝜌𝑉 𝑑𝑣 𝑊 +𝑊 𝑊 +𝑊

1

and



=

̅ ̅

1 2 1 2

𝐾 =

Where

3

̅̅̅

̅

̅

̅

̅

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | 𝑾𝟏**(kg/min)** | 𝑾𝟐**(kg/min)** | 𝑿𝟏 | 𝑿𝟐 | 𝒉**(m)** | 𝑿 | 𝝆(𝒈/𝒄𝒎𝟑) |
| **Group** **3** | 750 | 275 | 0.25 | 0.66 | 1.6 | 0.36 | 1 |

2. Manipulated Variable/Output Transfer Function: (by setting the load to a constant)

(𝜌𝑉𝑠+𝑊 + 𝑊 )𝑋′(𝑠) = 𝑊 𝑋′1(𝑠) + (𝑋2 − 𝑋)𝑊′2(𝑠)

1 2 1

̅ ̅

(𝜌𝑉𝑠+𝑊 + 𝑊 )𝑋′(𝑠) = (𝑋2 − 𝑋)𝑊′2(𝑠)

1 2

̅ ̅

(𝑋 − 𝑋)

̅

̅

1 2

2

̅

1 2

̅

1 2

𝑠 + 1

𝐺 = = = =

̅

𝑋′(𝑠) (𝑋2 − 𝑋) 𝑊 + 𝑊 𝐾 𝑣

𝑚𝑣 𝑊′2(𝑠) 𝜌𝑉𝑠 + 𝑊 + 𝑊 𝜌𝑉 𝑠 + 1 𝑊 + 𝑊

(𝑋2−𝑋) 𝜌𝑉 𝑚𝑣 𝑊 +𝑊 𝑊 +𝑊

and



=

̅ ̅

1 2 1 2

̅

𝐾 =

Where

Calculating the Transfer Function constants, using values for Group1:

𝐾𝑣 = 𝑊 𝑊 𝑊 = = 0.73

1

+

̅

1 2

𝐾 𝑣 = (𝑋2 − 𝑋) = 0.66 − 0.36 = 2.9 × 10−4 𝑚𝑖𝑛/𝑘𝑔

𝑊 + 𝑊

̅

̅

750 + 275

𝑉𝑖𝑞𝑢𝑖𝑑 = 𝜋 𝐷2ℎ = 𝜋 × 22 × 1.5 = 3𝜋 𝑚3

4 4 2

𝜌 = 1𝑐𝑔3 × 11𝐾𝑔 × 106𝑐𝑚3 = 1000 𝐾𝑔/𝑚3

𝑚 000 1𝑚

3

= = 4.595 min

𝐺 = =

The transfer functions to relate load and manipulated variable to the output variable:

𝐾𝑣 0.73

𝑑𝑣 𝑠 + 1 4.595𝑠 + 1

𝐺 = =

𝐾 𝑣 2.9 × 10−4 𝑚𝑣 𝑠 + 1 4.595𝑠 + 1

Transfer Function for the composition measurement device:

The device receives input in terms of concentration and outputs and electrical signal.

∆𝑒𝑙𝑒𝑐𝑡𝑟𝑖𝑐 𝑠𝑖𝑔𝑛𝑎𝑙 −𝜃𝑠 20 − 4 −1𝑠 16 −𝑠 −𝑠 𝑚 𝑐𝑜𝑚𝑝𝑜𝑠𝑖𝑡𝑖𝑜𝑛 𝑠𝑝𝑎𝑛 0.5 0.5

𝐺 = 𝑒 = 𝑒 = 𝑒 = 32𝑒

Transfer Function for the final control element (a valve):

𝐾 = 1.2 = 250𝑚𝑖𝑛.𝑝𝑠𝑖

300 𝑘𝑔

∆𝑝𝑟𝑒𝑠𝑠𝑢𝑟𝑒 𝑠𝑖𝑔𝑛𝑎𝑙 𝐾 15 − 3 250 187.5

𝐺 = = × =

𝑣 ∆𝑒𝑙𝑒𝑐𝑡𝑟𝑖𝑐 𝑠𝑖𝑔𝑛𝑎𝑙 𝜏𝑣𝑠 + 1 20 − 4 0.08335𝑠 + 1 0.08335𝑠 + 1

Transfer function for the controller:

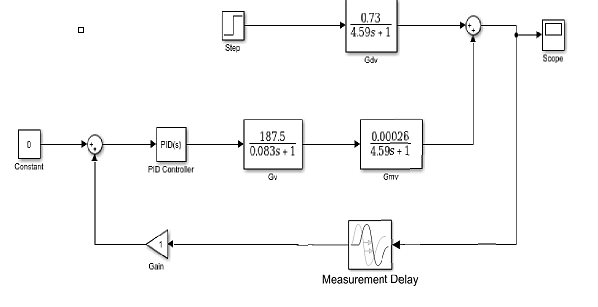
𝐺 = 𝐾 (1 + 𝜏𝐼𝑠 ) (PI) 𝐺 = 𝐾 (1 + 𝜏𝐼𝑠 + 𝜏𝐷𝑠) (PID)

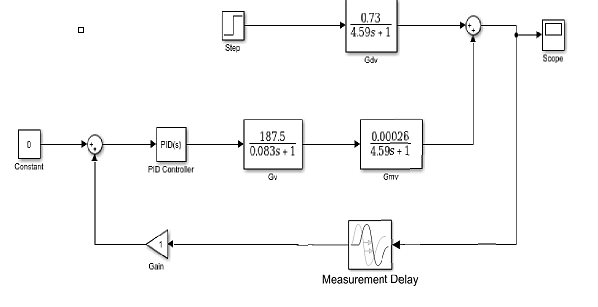
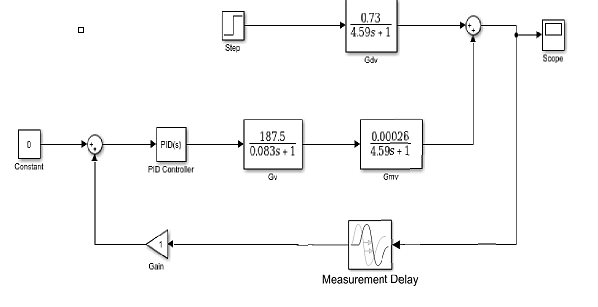
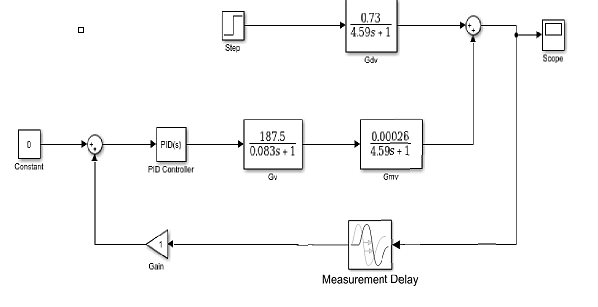
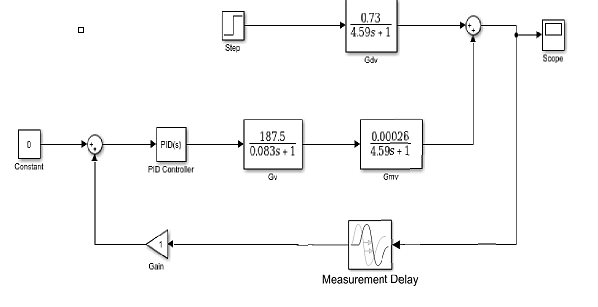
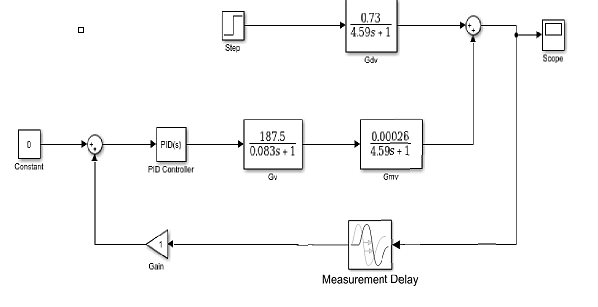
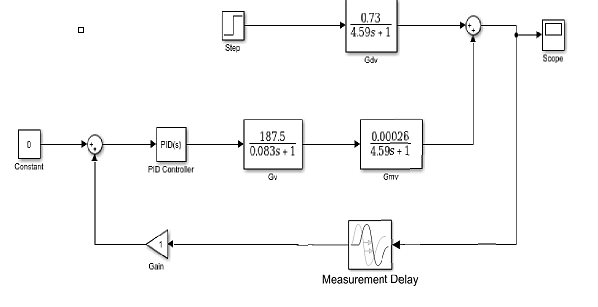
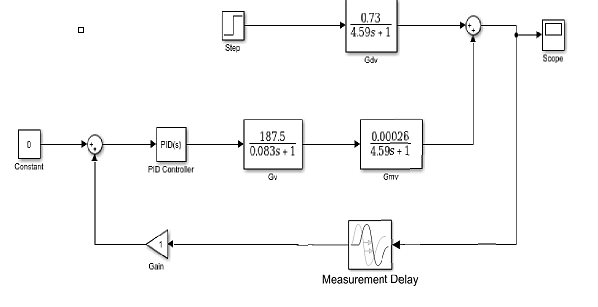
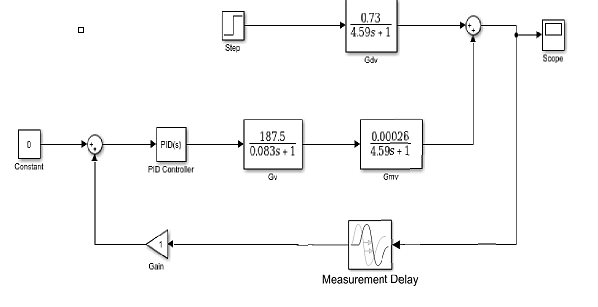
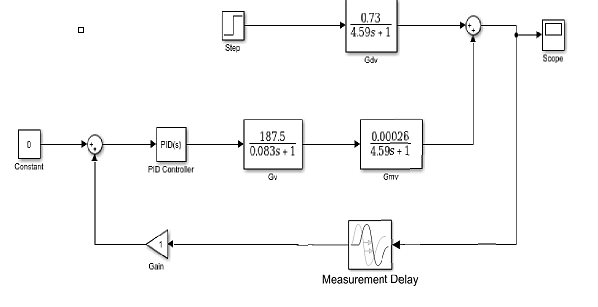
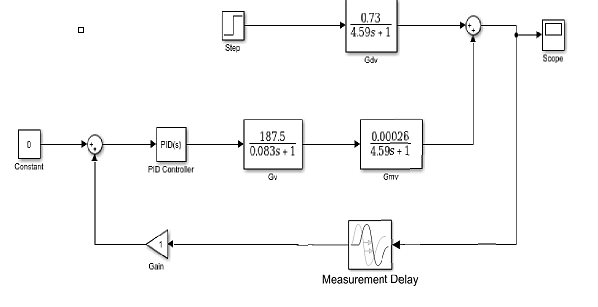
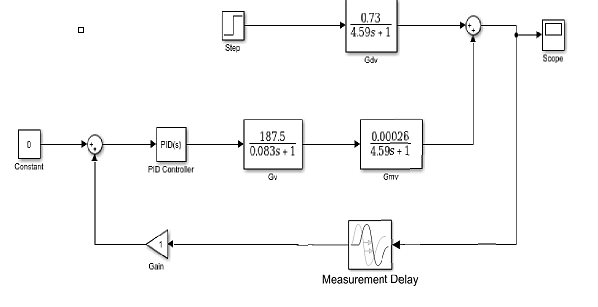
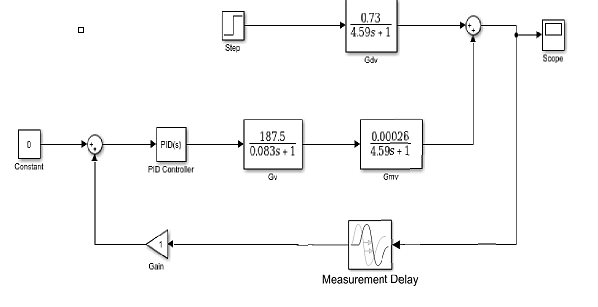
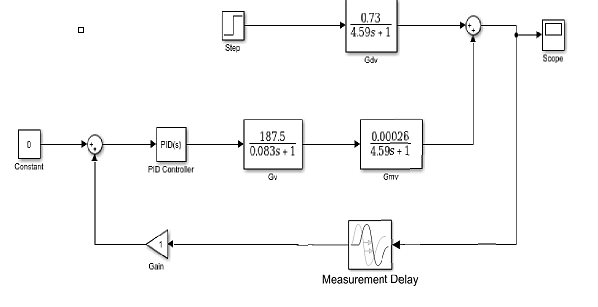
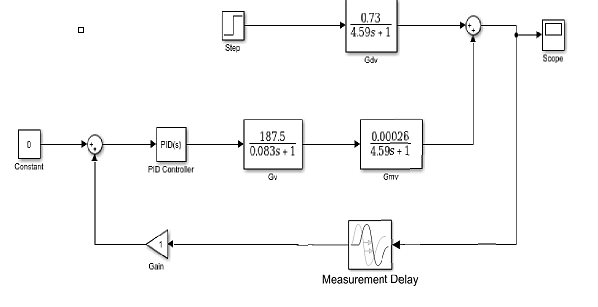
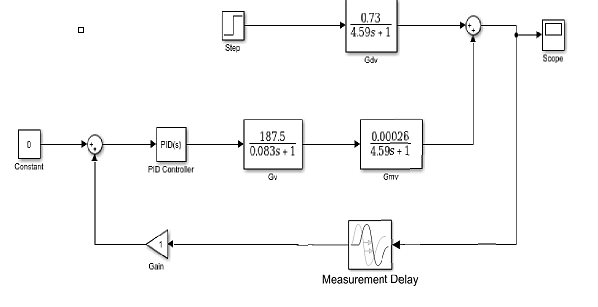
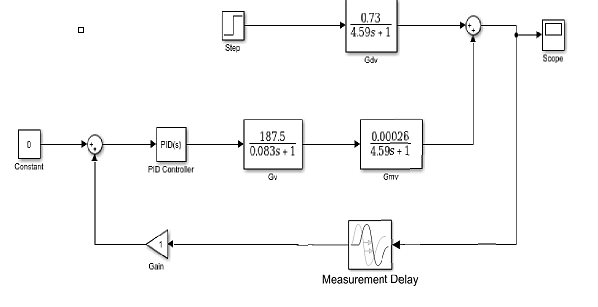
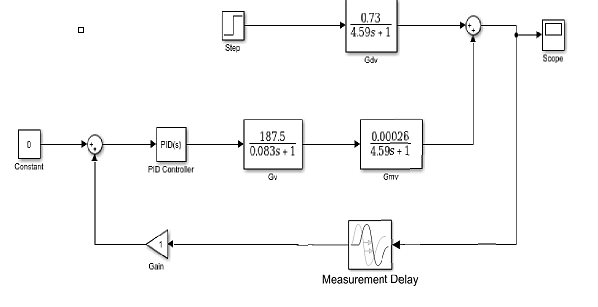
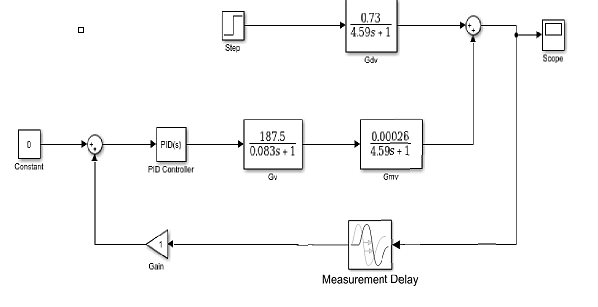
1

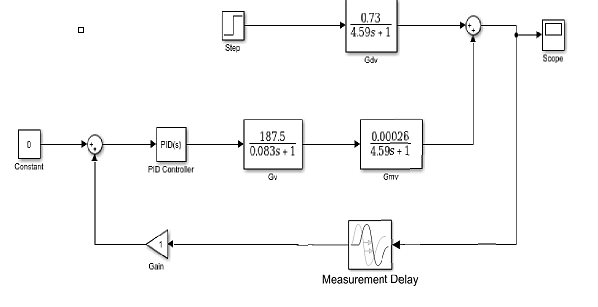
1

**c)** **Use** **Simulink** **to** **determine** **the** **PI** **controller** **settings/** **parameters** **using** **the** **step** **response** **method** **and** **Ziegler-Nichols** **settings.**

We simulate the process on Simulink:







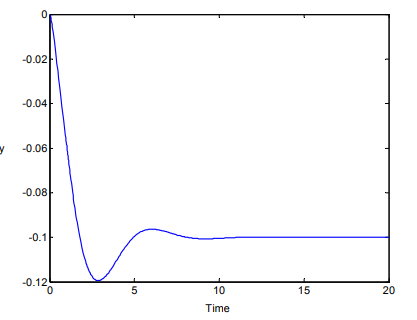
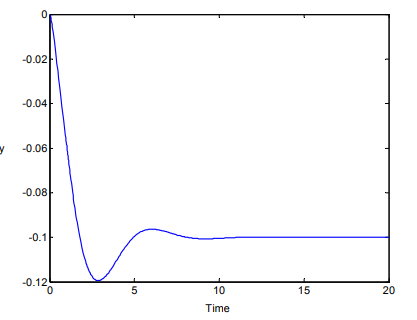
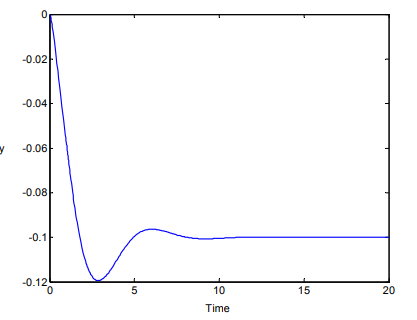
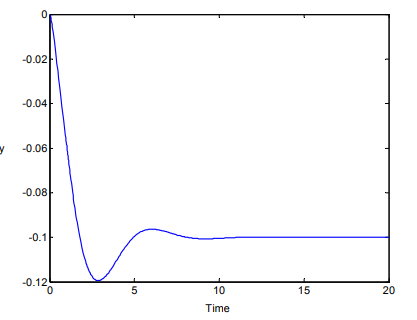
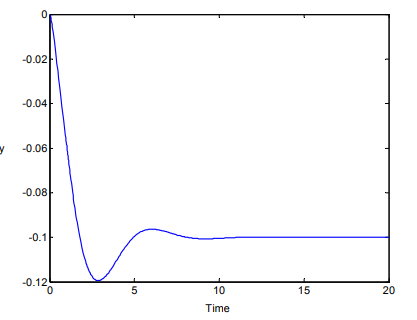
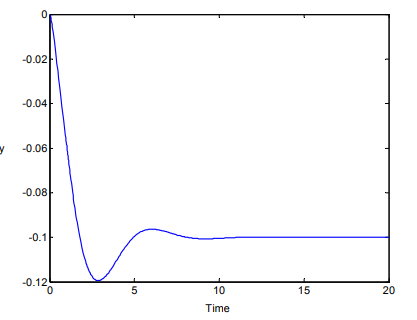
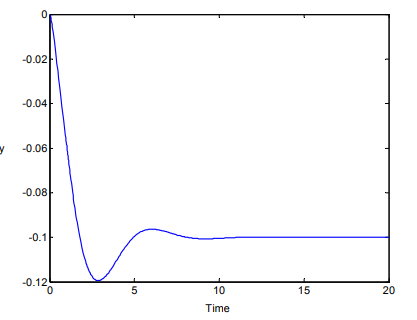
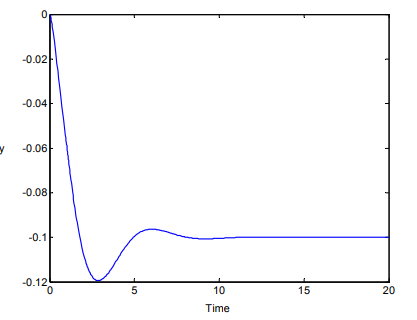
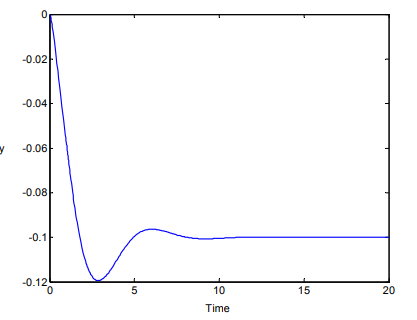
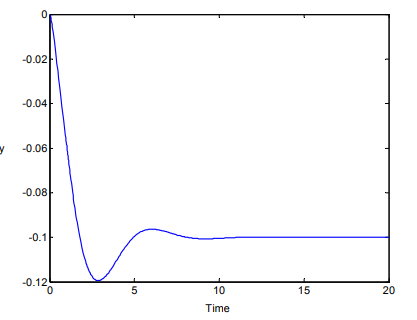
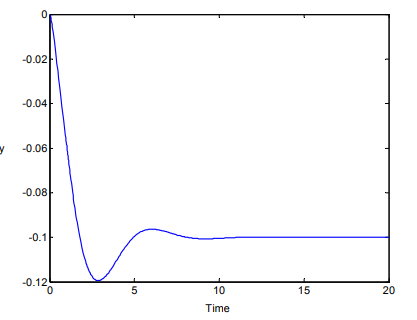
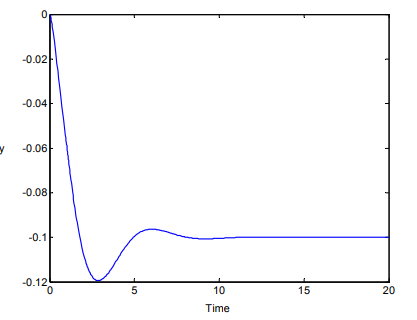
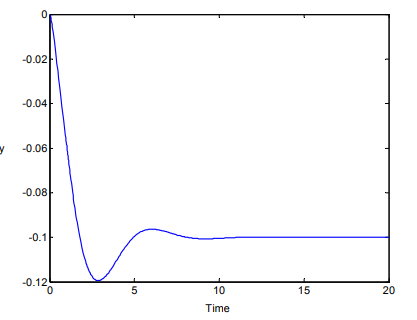
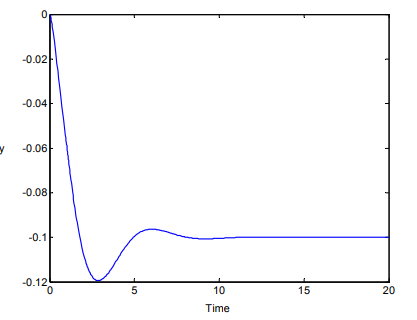
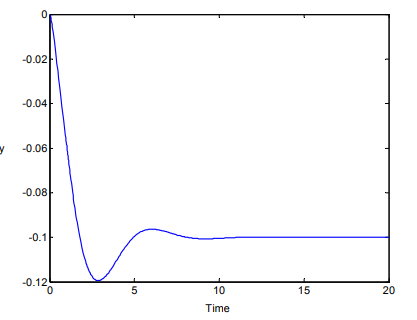
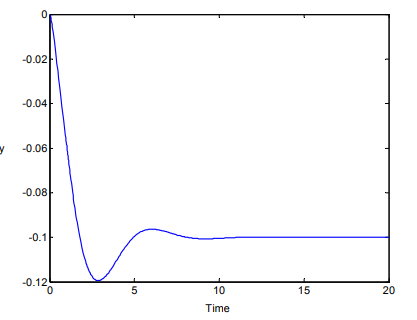
5

**c) The simulation results for the task c:**

**d)** **Simulate** **the** **closed** **loop** **response** **for** **the** **determined** **PI** **controller** **settings/parameters** **and** **a**

**step** **disturbance** **of** **+0.2** **in** 𝑿𝟏**.**

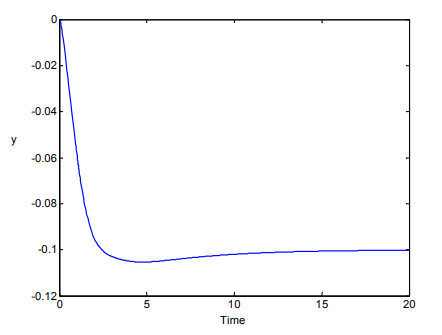
               

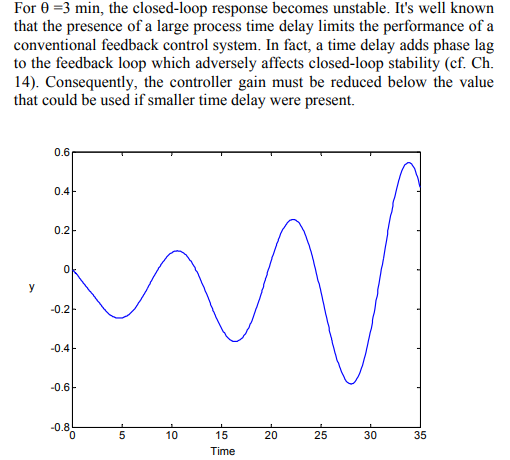
6

**e)** **Attempt** **to** **obtain** **improved** **control** **by** **adding** **derivative** **action** **to** **your** **PI** **controller** **of** **part** **(c).**

**Try** **several** **values** **of** **derivative** **time** 𝜏𝐷**.** **Which** **one** **gives** **the** **best** **results** **for** **a** **step** **disturbance** **of**

**+0.2** **in** 𝑿𝟏**?**





So, since we are only adding the derivative action to our already tuned PI controller, with:

𝐾 = 0.45 × 𝐾𝑢 = 2.1915 𝜏𝐼 = 1.2 = 3.3316 𝑚𝑖𝑛

𝑃

𝑢

we use different values for 𝜏𝐷 to compare their effects on the process response, we also included one graph for the PI controller to notice the effect of adding a derivative component:

We notice that introducing a derivative action to the process reduced the overshoot. With the overshoot being lower the higher the value of the derivative time constant. We also notice that starting from

𝜏𝐷 = 0.6 the process response becomes less dampened and oscillations become more apparent, so, we try to choose a value less than 0.6.

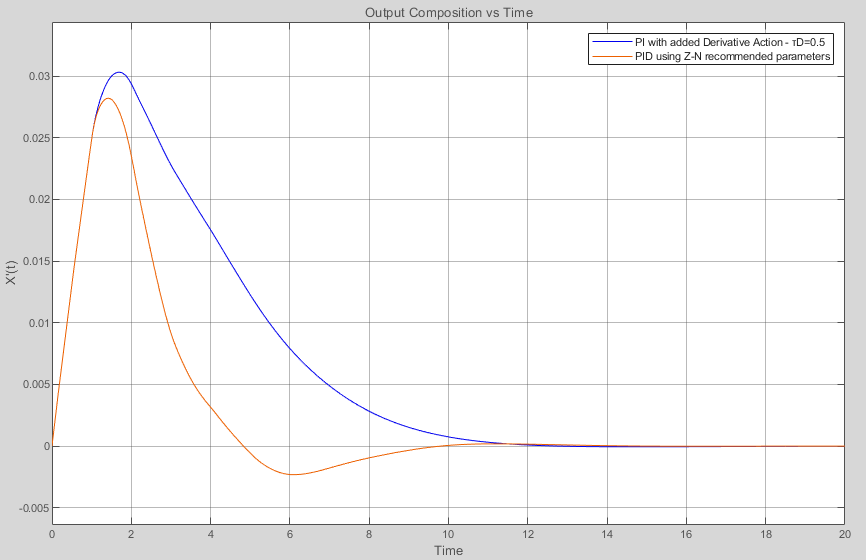
7

The above graph provides a close look at 0.2 ≤ 𝜏𝐷 < 0.6 and we notice that there are two things to take into consideration when choosing a value for 𝜏𝐷: the decrease in overshoot, and the sluggishness of the process. A good choice to would be decrease the overshoot as much as possible while not making the process too sluggish. We choose 𝜏𝐷 = 0.5 𝑚𝑖𝑛 because it gives a good decrease in the overshoot while still maintaining a fast response.

The chosen Tuning Parameters:

|  |  |  |
| --- | --- | --- |
| **PI** **controller** **with** **added** **derivative** **action** | | |
| 𝑲𝒄 | 𝝉𝑰 | 𝝉𝑫 |
| 2.1915 | 3.3316 𝑚𝑖𝑛 | 0.5 𝑚𝑖𝑛 |

8



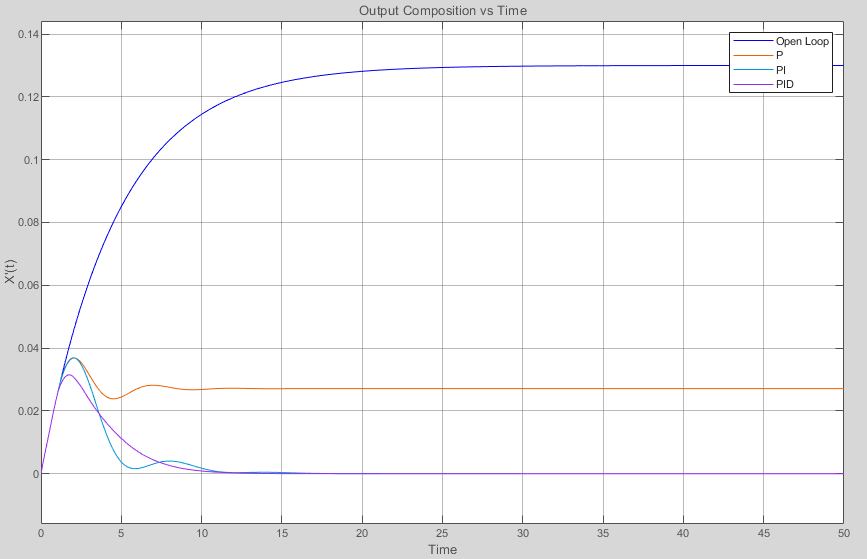
|  |  |  |
| --- | --- | --- |
| **PID** **controller** **using** **Ziegler-Nichols** **parameters** | | |
| 𝑲𝒄 | 𝝉𝑰 | 𝝉𝑫 |
| 0.6 × 𝐾𝑢 = 2.922 | 𝑃/2 = 1.999 𝑚𝑖𝑛 | 𝑃/8 = 0.49975 min |

It is important to note that one main benefit of using a PID controller is that it allows for the use of a

higher proportional gain and a lower Integral time constant which has the effect of reducing the overshoot and the oscillations if tuned well. The Ziegler-Nichols method provides a way to optimize the parameters of a PID controller as well. The table gives the following parameters for a PID controller using the same 𝐾𝑢 and 𝑃we found in **part** **(c)**:

The following sketch graph shows the comparison between only adding a derivative action to an already tuned PI controller and using a PID controller which was tuned for recommended parameters using Ziegler-Nichols continuous cycling method for a PID controller. We notice that re-tuning the Controller greatly enhanced the controlling action by further decreasing the overshoot and the speed of the controlling action.

9



Finally, we present this graph to compare the effect of different controlling configurations on our system

with a step change of +0.2 in 𝑋 :

1

We notice that introducing a proportional controller greatly decreased the disturbance effects on the system, but it did not eliminate it entirely. Adding an integrator to the controller eliminated the offset and forced the controlled variable back to its set point, and adding a derivative component eliminated the oscillations while maintaining a fast and smooth response.

References

[1] D. E. Seborg, T. F. Edgar, D. A. Mellichamp and F. J. Doyle, Process Dynamics and Control, Wiley, 2017.

10