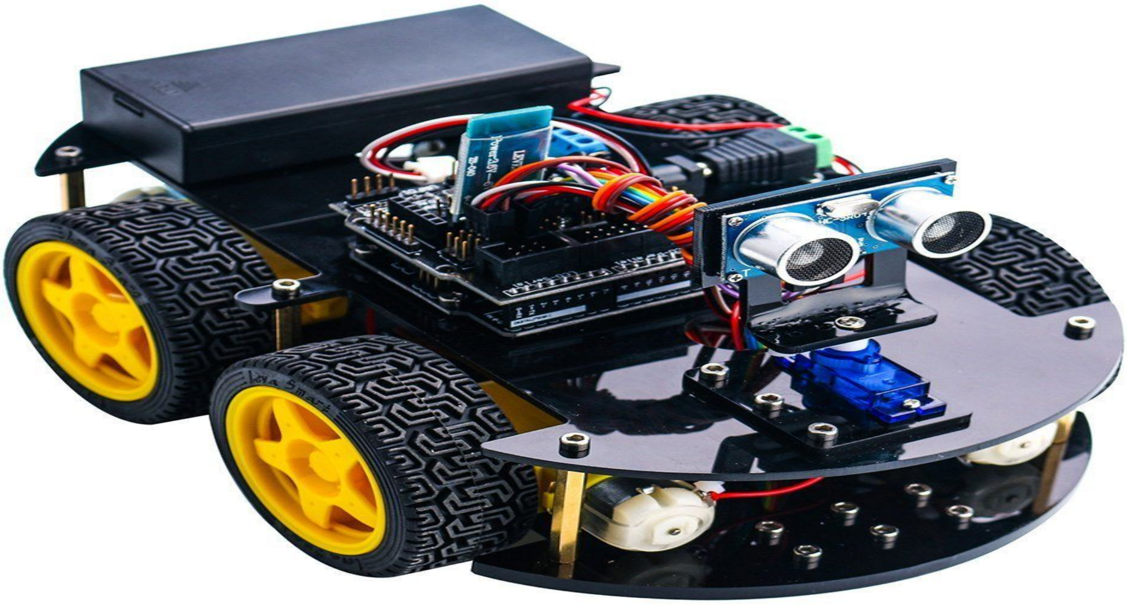


Microcontroller Laboratory Exercise

CONTROL OF A MOBILE PLATFORM IN LABVIEW ENVIRONMENT

Lab Report

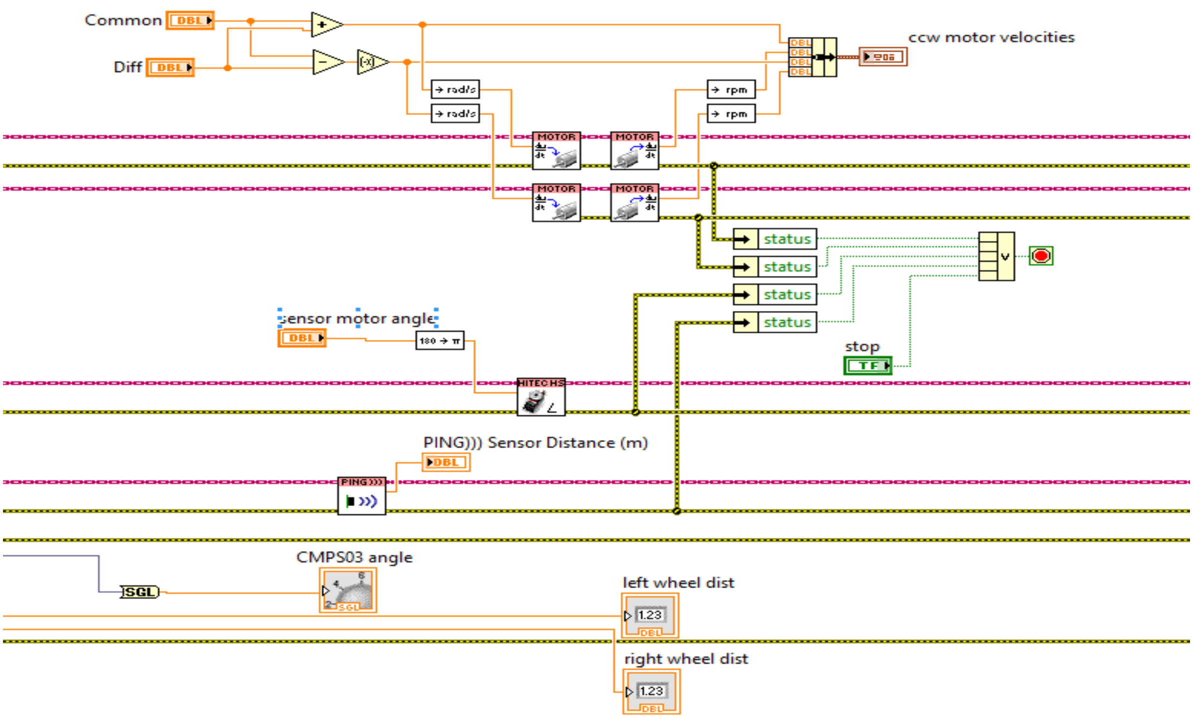
Students' Name: Das Anjan Kumar (D4DQA)

Yousha Mohammed Shamin Yeasher (WBDDO4)

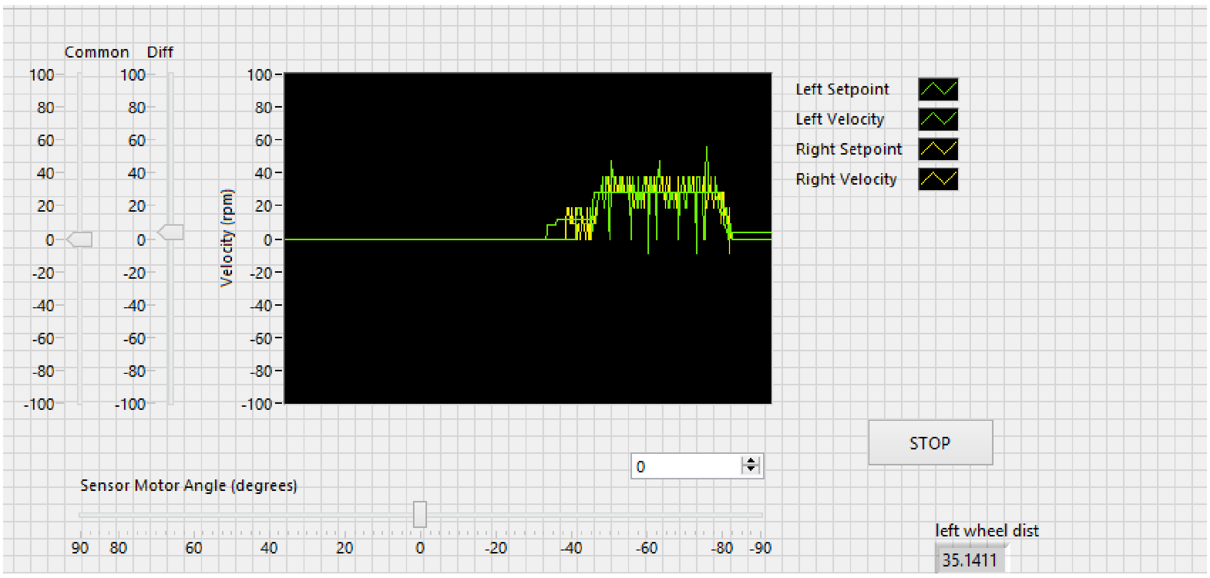
Lab Instructor: Paczolay Gabor

Task 5.1:

In this experiment we will be working on the VI connected to the Mobile Robot. The initial connection of the VI is as such that, there are many connections in the VI But we will be focusing on only the important ones.



We can see from the above screenshot that there are CMPs03 is the output of the given arc lengths in the ROBOT itself. We can also see the motor and the left and right wheel distance as well. When we run the robot we can see that the robot wheels are moving which can be controlled from the Block diagram of the VI. We can also control the direction of the sensor and the velocity of the wheels independently. Below we can see the screen shot of the control panel from the Block diagram of the VI.



Task 5.2:

For this task we will measure the value of the 2e, to measure the value of the three equations from the lab guide, we will

$$x_{i+1} = x_i + \Delta L \cos\left(\theta_i + \frac{\Delta\theta}{2}\right)$$

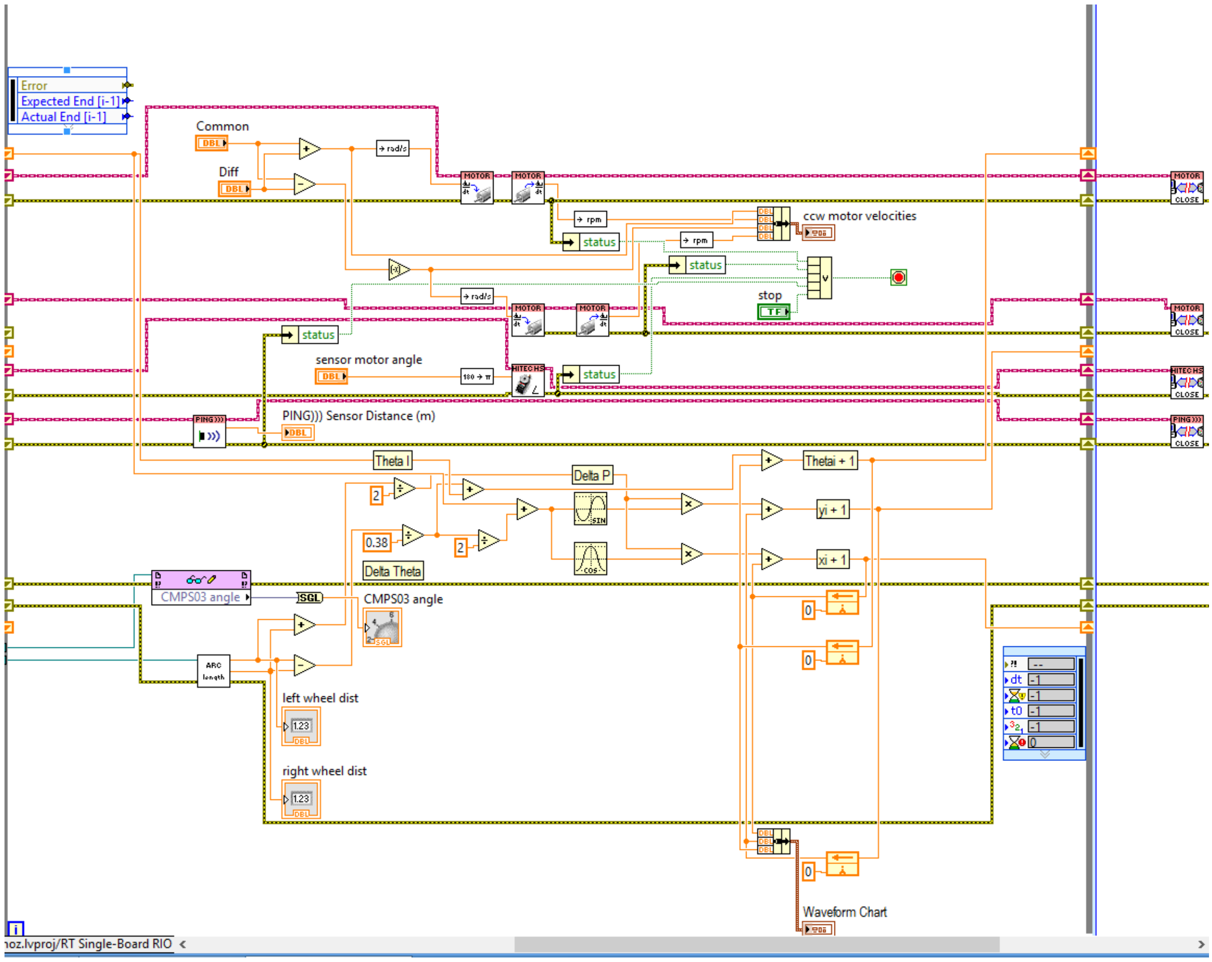
$$y_{i+1} = y_i + \Delta L \sin\left(\theta_i + \frac{\Delta\theta}{2}\right)$$

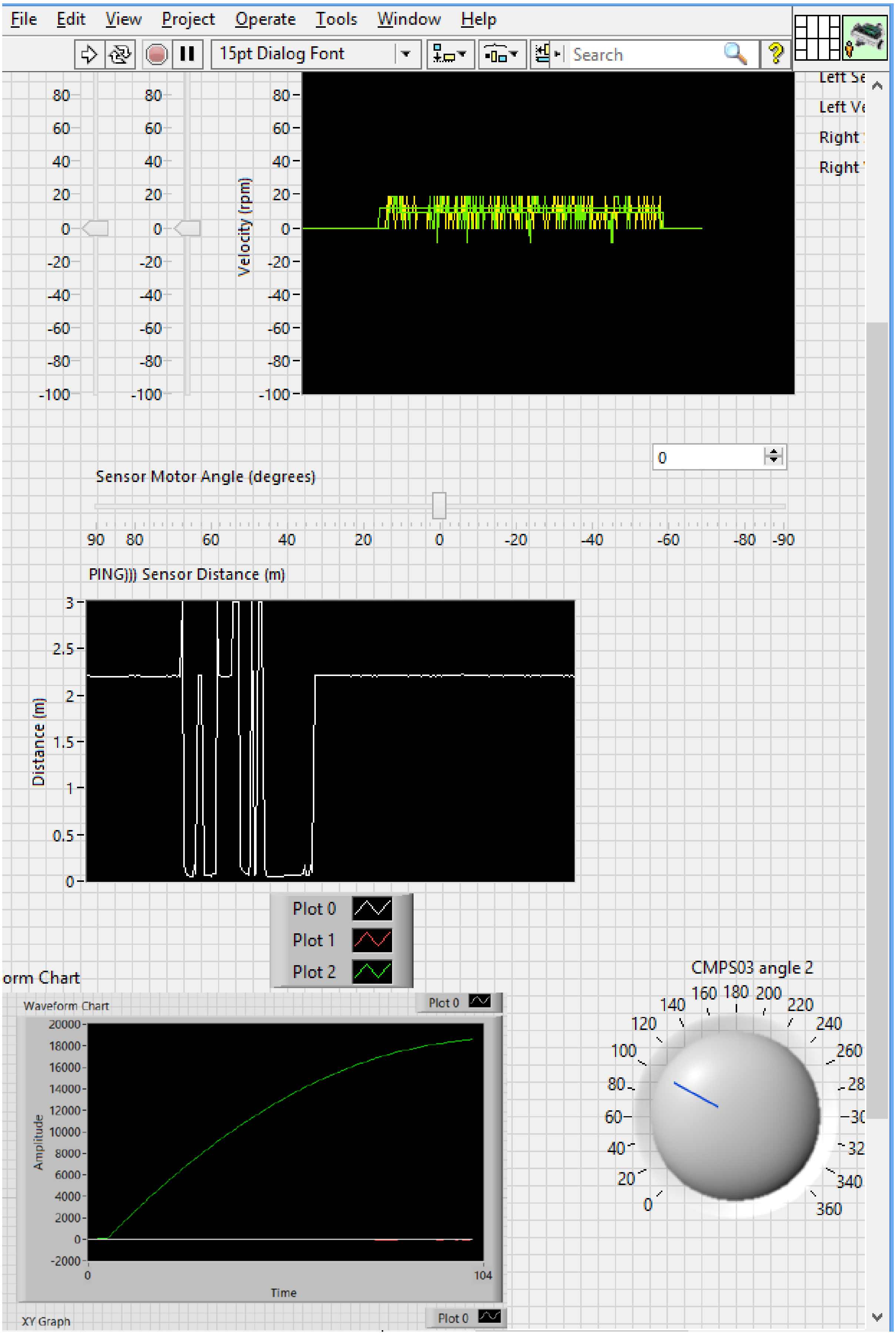
$$\theta_{i+1} = \theta_i + \Delta\theta = \theta_i + \frac{\delta_R - \delta_L}{2e},$$

According to the equations we will need to set up the diagram in the front panel of the VI and after that we will need to put the value in a array.

We can use general and regular functions from the Front panel function palette. According to the functions we will get the output of the xi , yi and thetai in an array. We will show that array in a waveform generator.

In the below screens shot we can see the front panel of the desinged functions.





File Edit View Project Operate Tools Window Help
15pt Dialog Font
Search
Velocity (rpm)
Sensor Motor Angle (degrees)
PING))) Sensor Distance (m)
Distance (m)
Plot 0
Plot 1
Plot 2
orm Chart
Waveform Chart
Amplitude
Time
CMPS03 angle 2
XY Graph

Task 5.3:

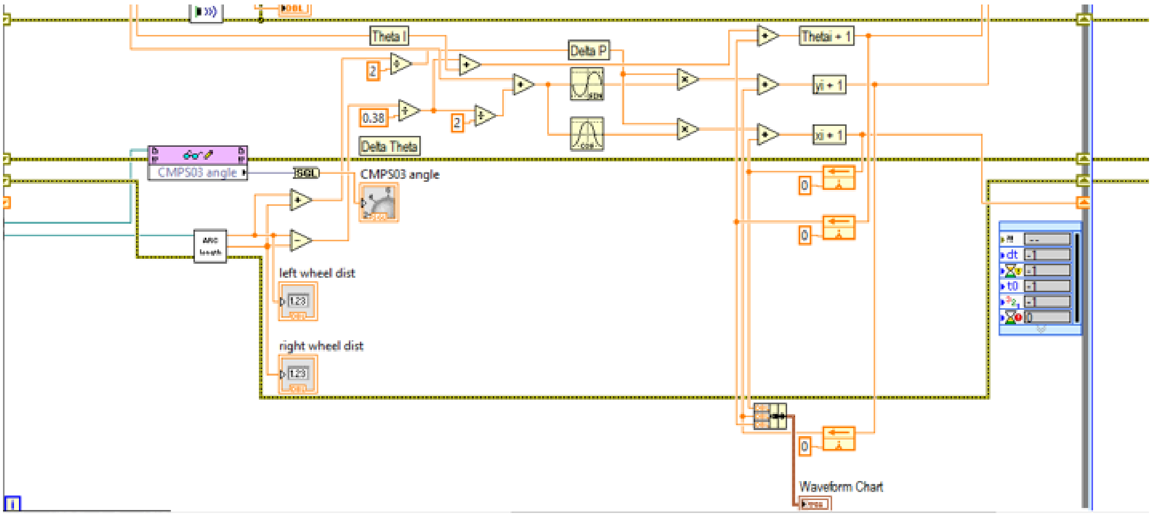
In this task we will measure the position and orientational error using the dead lock reckoning after a motion sequence.

In navigation, dead reckoning is the process of calculating current position of some moving object by using a previously determined position, or fix, and then incorporating estimates of speed, heading direction, and course over elapsed time.

In this task we will continuously take data from the thetai which will be used to determine the measurement and orientational error from the dead reckoning readings.

$$\theta_{i+1} = \theta_i + \Delta\theta = \theta_i + \frac{\delta_R - \delta_L}{2e},$$

The connections of the thetai is given in the screen shot below:



Task 5.4:

In this task we will increase the diameter of the wheels and go through the same steps as before in 5.2 and 5.3.

where r is the instantaneous radius of rotation. Denote by $\rho = \frac{1}{r}$ the instantaneous curvature. Their expressions read

$$r = e\frac{\delta_R + \delta_L}{\delta_R - \delta_L} \qquad \rho = \frac{1}{e}\frac{\delta_R - \delta_L}{\delta_R + \delta_L}, \tag{4.7}$$

The radius of the wheels of the mobile robot can be a big factor in case of calculating the positional and orientational error of the robot. Error accumulation may be caused by inaccurate knowledge of a parameter like e, the sliding of wheels or inaccurate knowledge of wheel diameters which allows calculating arc lengths _L and _R based on rotation angles of wheel axes.

As we increase the radius of the wheels, we can notice change in the positional and orientational error.

Task 5.5:

In this task we will implement the Kalman-filter for the estimation of orientation. Here down below we can see the equations for the Kalman-filter:

$$\hat{\zeta}_{k|k-1} = \hat{\zeta}_{k-1|k-1} + \frac{1}{2e}u_k$$
$$P_{k|k-1} = P_{k-1|k-1} + \sigma_w^2 \,.$$

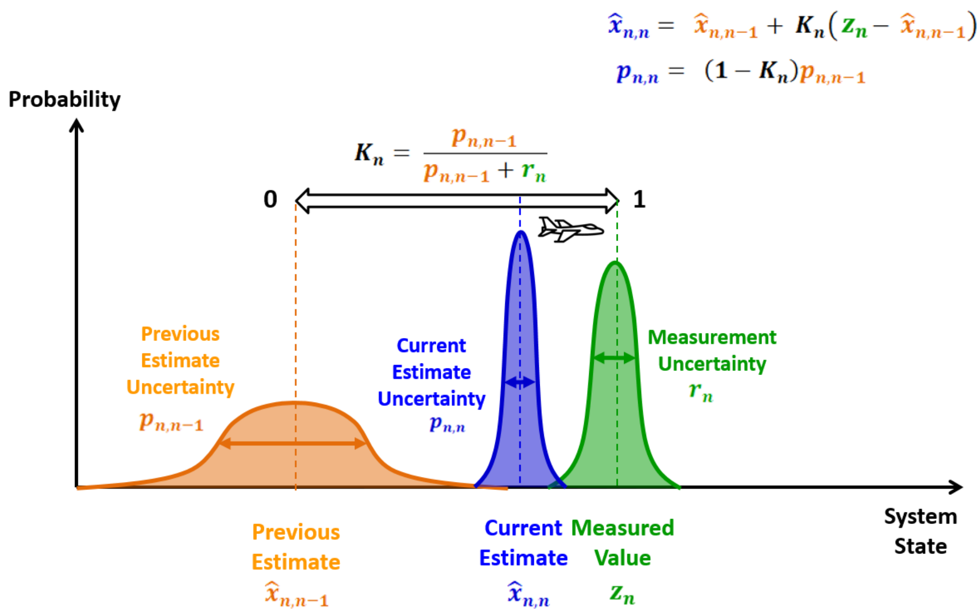
$$S_k = P_{k|k-1} + \sigma_v^2$$
$$K_k = P_{k|k-1}S_k^{-1} \,.$$

$$\tilde{z}_k = z_k - \hat{\zeta}_{k|k-1}$$

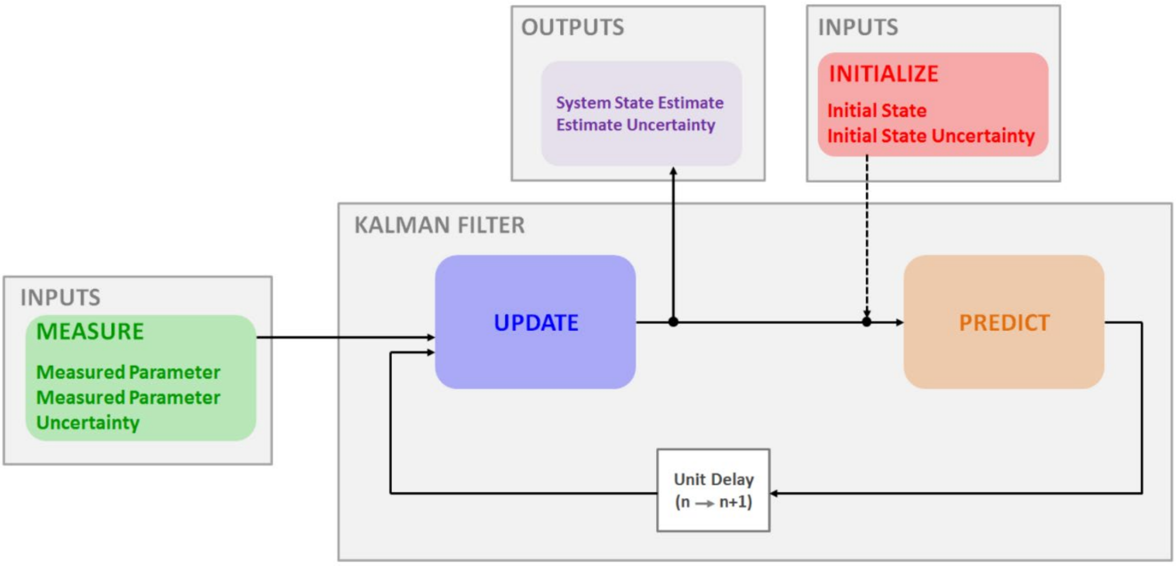
$$\hat{\zeta}_{k|k} = \hat{\zeta}_{k|k-1} + K_k\tilde{z}_k$$

$$P_{k|k} = (1 - K_k)P_{k|k-1}$$

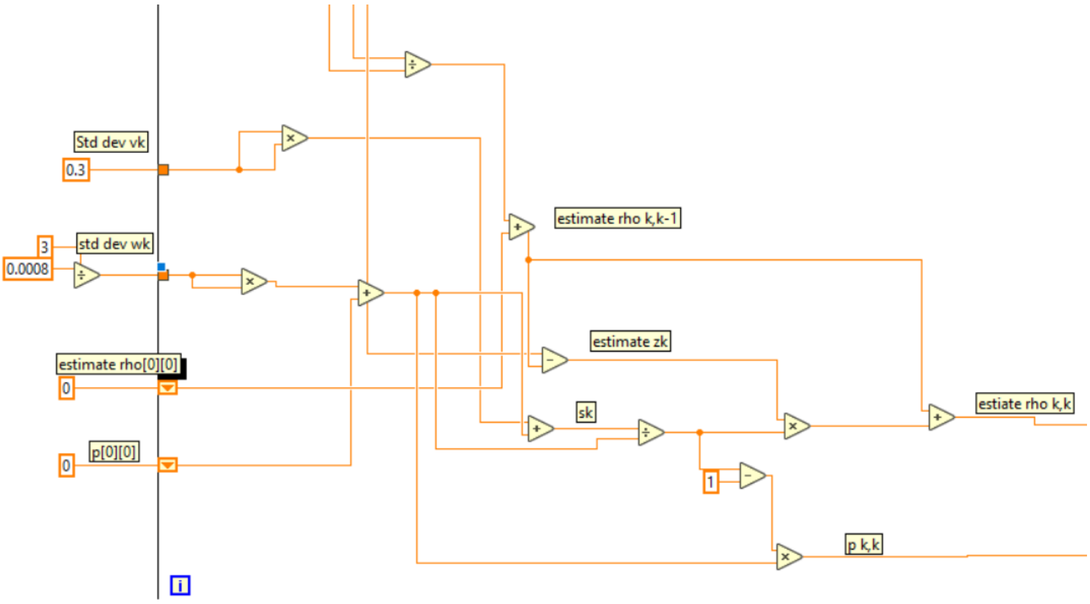
We generally use Kalman filter to decrease the error estimation of the positional and orientational estimate uncertainty. The concept can be more clear from the screenshot down below:



The operational block of Kalman filter can be observed down below:



Here in the NILabview, we can implement the equations for the kalman filter. The implementation is given here:



We can notice that after using this filter, the errors will be much compensated due to the radius change.