

ภาคผนวก ค

โค้ดการเขียนสมการKalman Filter

โค้ดการเขียนสมการKalman Filter

```
function angle_Out = KalmanFilter(RollAngle,Rate_Gyro)
% Initialise persistent parameters
persistent X; % State vector x = [roll; pitch]
persistent P; % Error covariance matrix
persistent angle_err;
persistent E;
persistent angle;
persistent q_bias;
if isempty(X)
X = 0;
end
if isempty(P)
P = [0 0; 0 0];
end
if isempty(angle)
angle = 0.0;
end
if isempty(angle_err)
angle_err = 0;
end
if isempty(E)
E = 0;
end
if isempty(q_bias)
q_bias = 0;
end
%R_angle = 0.01;
R_angle = 0.01;
dt = 0.02; % sampling peroid
C = [1 0]; % The C matrix is a 1x2 (measurements x states)
Q_angle = 0.002;
Q_gyro = 0.003;
Q = [Q_angle 0; 0 Q_gyro]; % process covariance noise
% it indicates how much we trust the accelerometer
% relative to the gyros
%////////// Rolling in X direction //////////
angle = angle + (Rate_Gyro - q_bias)*dt; % Update our angle estimate
P(1,1) = P(1,1) - (P(2,1)+P(1,2))*dt + Q_angle*dt; % Update the covariance
matrix
P(1,2) = P(1,2) - P(2,2)*dt; % Update the covariance matrix
P(2,1) = P(2,1) - P(2,2)*dt; % Update the covariance matrix
P(2,2) = P(2,2) + Q_gyro*dt; % Update the covariance matrix
angle_err = RollAngle - angle;
E = C*P*C' + R_angle; % Compute the error estimate. From the Kalman filter
paper:
K = P*C'*inv(E); % Compute the Kalman filter gains
angle = angle + K(1)*angle_err;
q_bias = q_bias + K(2)*angle_err;
P = P - K*C*P; % Update covariance matrix
angle_Out = angle;
```