Linear Quadratic Gaussian Integral(LQGI)

This controller works for the most cases. It's a standard controller for simple problems such as temperature control, water level control, electrical DC motor control etc. You should choose this controller if you have:

- No disturbances is effecting the system
- No reference model following is required
- The control object is temperature system, electrical or fluid system.

Here is a step by step to create the infinite horizon LQGI controller:

1. Start to estimate your discrete state space model. Select the OKID, MOESP or N4SID methods to get the estimated model.

$$\dot{x}(k+1)=Ax(k)+Bu(k)$$
$$y(k)=Cx(k)+Du(k)+w(k)$$

2. Start to solving Discrete Algebraic Riccati Equation(DARE) by iterating it like 100 times:

$$X(k+1)=AX(k)A^T-(A^TX(k)C^T)(R+CX(k)C^T)^{-1}(CX(k)A^T)+Q$$

Where $w(k)$ is the measured steady state noise vector from a sensor or sensors and:

$$Q=I$$
$$R=E\{w(k)w^T(k)\}$$
$$X(0)=0$$

3. After solving DARE, then find the steady state Kalman gain matrix K

$$K=(CX(k)C^T+R)^{-1}(CX(k)A^T)$$

4. Now create the augmented state space model with integral action.

$$\begin{bmatrix} x(k+1) \\ x_i(k+1) \end{bmatrix} = \underbrace{\begin{bmatrix} A & 0 \\ -C & 1 \end{bmatrix}}_{\mathring{A}} \begin{bmatrix} x(k) \\ x_i(k) \end{bmatrix} + \underbrace{\begin{bmatrix} B \\ -D \end{bmatrix}}_{\mathring{B}} u(k) + \begin{bmatrix} 0 \\ 1 \end{bmatrix} r(k)$$

$$y(k)=\underbrace{\begin{bmatrix} C & 0 \end{bmatrix}}_{\mathring{C}} \begin{bmatrix} x(k) \\ x_i(k) \end{bmatrix} + Du(k)$$

5. Start solving another DARE by iterating it like 100 times.

$$X(k+1)=\mathring{A}^TX(k)\mathring{A}-(\mathring{A}^TX(k)\mathring{B})(R+\mathring{B}^TX(k)\mathring{B})^{-1}(\mathring{B}^TX(k)\mathring{A})+Q$$

With:

$$Q=I$$
$$R=\rho I,\rho>0$$
$$X(0)=0$$

6. After have solve that DARE, then find the control law and integral law:

$$\begin{bmatrix} L & L_i \end{bmatrix}=(\mathring{B}^TX(k)\mathring{B}+R)^{-1}(\mathring{B}^TX(k)\mathring{A})$$

Now we have found our Kalman gain matrix K and the control law L and the integral law L_i
Now it's time to create the LQGI controller:

1. Set the initial state vector $x(k)=0, x_i(k)=0, k=0$, and your reference $r(k)$ to your desire value.
2. Measure with your sensor or sensors the output signal $y_m(k)$
3. Compute the control law

   $u(k)=K_f r(k)-(Lx(k)-L_i x_i(k))$

   And update the integral law.

   $x_i(k+1)=x_i(k)+r(k)-y_m(k)$

   K_r is a reference gain where q is an integral pole. It will make the integral action slower $0 \leftarrow q$ or faster $q \rightarrow 1$.

   $K_f=\frac{L_i}{1-q}, 0 \le q < 1$

4. Then compute the estimated state from the Kalman filter.

   $x(k+1)=(A-KC)x(k)+Bu(k)+Ky_m(k)$

Now repeat the following to control the object with the LQGI controller:

1. $u(k)=K_f r-(Lx(k)-L_i x_i(k))$ - Control
2. $x_i(k+1)=x_i(k)+r(k)-y_m(k)$ - Integrate
3. $x(k+1)=(A-KC)x(k)+Bu(k)+Ky_m(k)$ - Estimate

Practical methods:

- Saturation on $u(k)$
- Anti-windup on $x_i(k)$