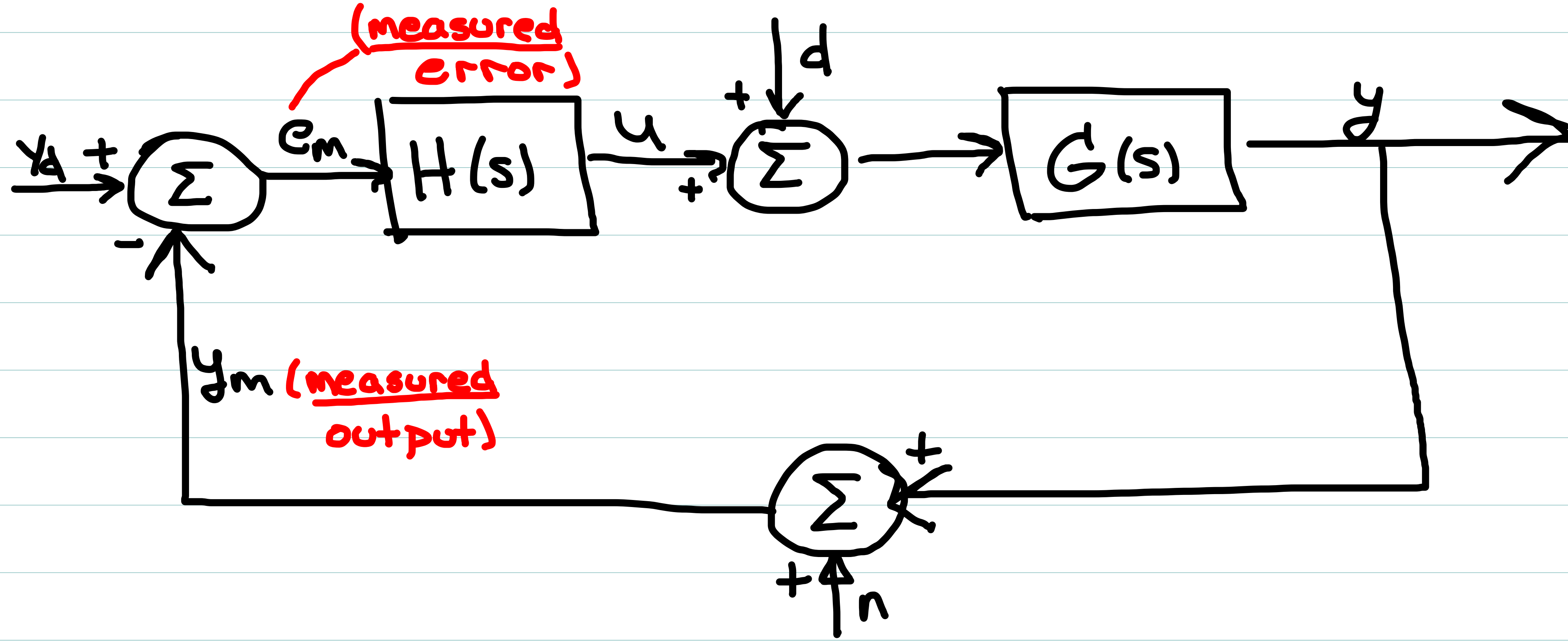
# Effect of sensor noise




Now: $Y = G[U+D]$, $U = HE_m = H[Y_d - (Y+N)]$

So: $Y = GHY_d - GHY + GD - GHN$

Or: $\boxed{Y = TY_d - S_i D - TN}$ ← Bad

$$Y_d \longrightarrow \boxed{T} \longrightarrow Y$$

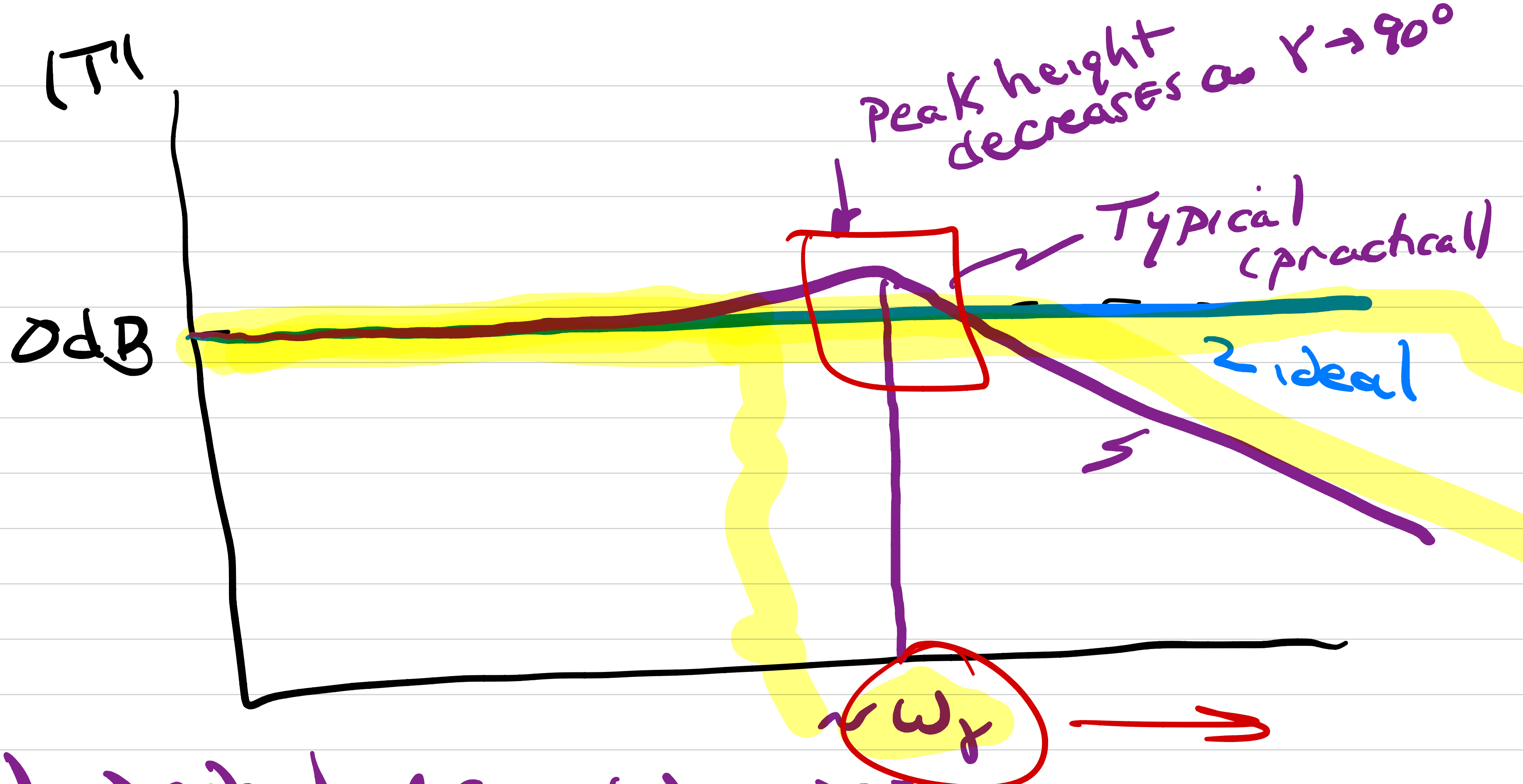
ideally (for tracking) $\boxed{Y = Y_d}$

$\Rightarrow$ ideally, $T(s) = 1$

$\Rightarrow$ $|T(j\omega)| = 0 \text{ dB}$

$\angle T(j\omega) = 0 \text{ deg}$

for all $\omega \geq 0$




Typical → ideal AS: $\omega_r \to \infty$
$\gamma \geq 60°$

But $\omega_r \to \infty$ means
- infinite phase loss from delay
- No robustness to model uncertainty
- impractically large $u(t)$
- high noise sensitivity

and hence: $E = Y_d - Y$ satisfies:

$$\boxed{E = (1-T)Y_d - S_i D + TN}$$

or: $E = \underbrace{\boxed{SY_d}}_{\text{Tracking error}} - \underbrace{\boxed{S_i D}}_{\text{error due to disturbance}} + \underbrace{TN}_{\text{Add'l error due to noise}}$

New term!

Note: TF from noise to $Y$ is same as TF from $Y_d$ to $Y$ (both are $T(s)$)

Implication: ⟹ feedback loop tries to "track the noise"

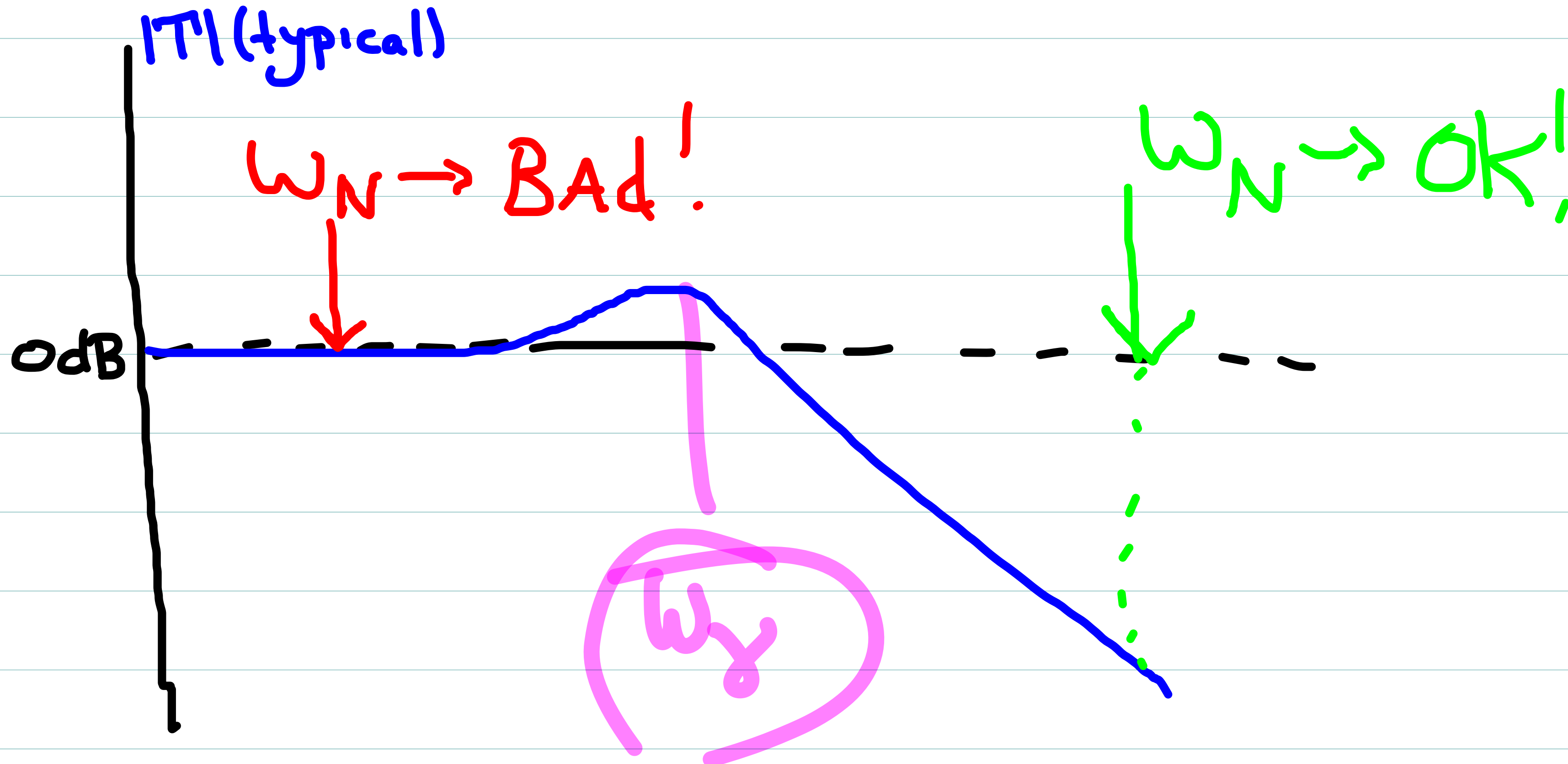
Equivalently: ⟹ noise is indistinguishable from "signal" $y(t)$ loop is trying to control!

# Impact of Noise

Assume for simplicity noise is "tonal": $n(t) = N\sin(\omega_N t)$
(it isn't really, but useful starting point!)

Then Added error is upper bounded by $N|T(j\omega_N)|$

$\Rightarrow$ Need $|T(j\omega)|$ small at noise frequency $\omega_N$!

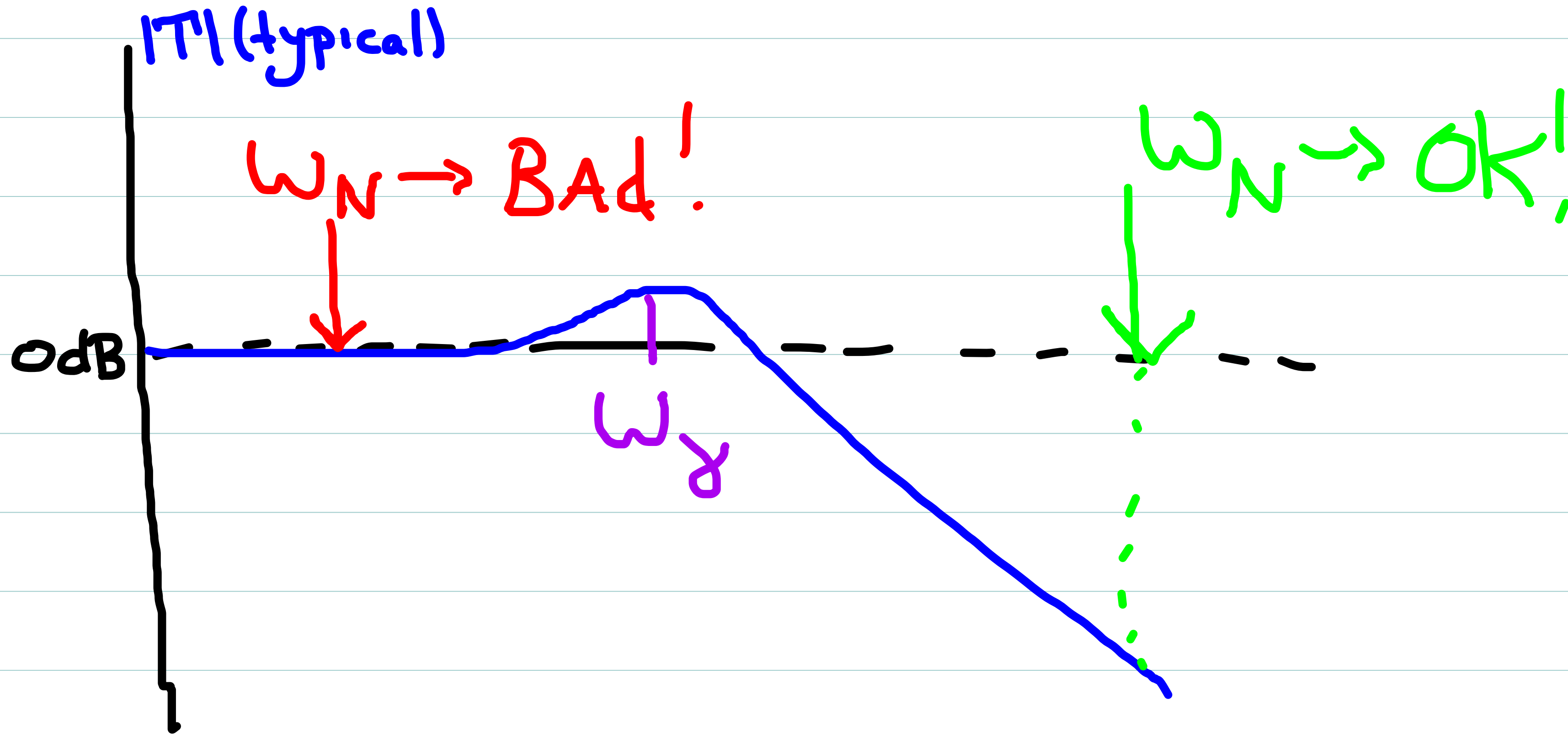


# Impact of Noise

Assume for simplicity noise is "tonal": $n(t) = N\sin(\omega_N t)$
(it isn't really, but useful starting point!)

Then Added error is upper bounded by $\boxed{N|T(j\omega_N)|}$

$\Rightarrow$ Need $|T(j\omega)|$ small at noise frequencies!



# Design Implications, I

$\Rightarrow$ Need $\omega_\gamma << \omega_N$

$\Rightarrow$ Constrains $\omega_\gamma$ / bandwidth

$\Rightarrow$ Conversely, designs with larger $\omega_\gamma$ will show worse performance due to increased noise impact!

Essentially, we need to make sure there is adequate separation between the frequencies we are trying to track (bandwidth), and the frequency of the noise.

$\Rightarrow$ Works against our desire for large $\omega_\gamma$ (fast settling)

# Another perspective:

With Noise, controller implementation equation is:

$$u(t) = C_0 \, e_m(t) + \sum c_K x_K(t)$$

$$\dot{x}_K(t) = a_K x_K(t) + e_m(t) \qquad [a_K \text{ poles of } H(s)]$$

Noise impacts $u(t)$:

⟹ directly if $C_0 \neq \emptyset$

⟹ indirectly through $x_K(t)$

$x_K(t)$ diff'l eq'ns have a "filtering" property (reduce magnitude of noise effects)

⟹ Designs with $C_0 = \emptyset$ have superior noise resistance

# Design Implications, II

$C_0 = \emptyset \iff H(s)$ has more poles than zeros

$\Rightarrow$ Designs with this property have better noise resistance!

$\Rightarrow$ Works against our need to increase phase margin

Most "Advanced" controller designs have 1 more pole than zeros to ensure good noise filtering.

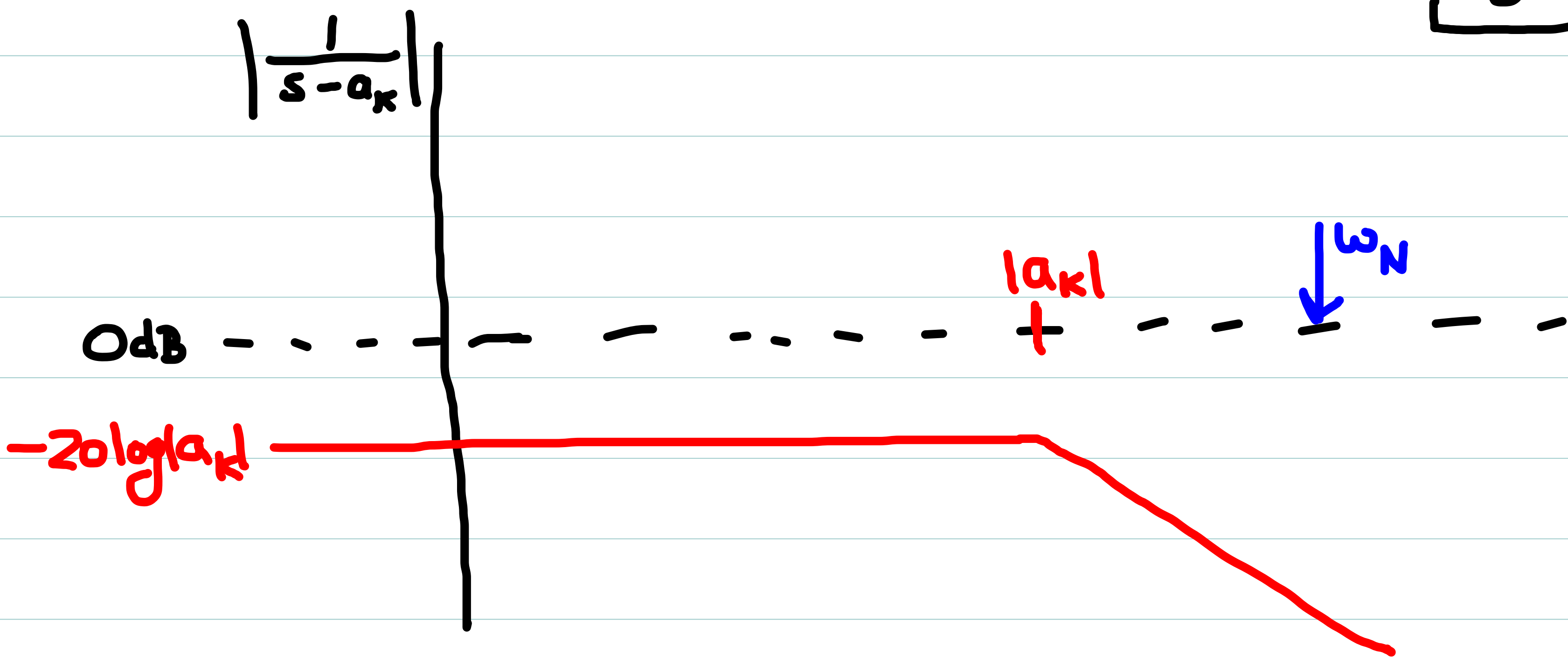
However, superior transient performance is achievable with $C_0 \neq \emptyset$ provided noise is not a significant issue.

# "Filtering" by $x_K(t)$ States

$$\dot{x}_K(t) = a_K X_K(t) + e_m = a_K X_K(t) + \underbrace{e(t)}_{\text{true error}} - \underbrace{n(t)}_{\text{sensor noise}}$$

$$\Rightarrow X_K(s) = \left[\frac{1}{s+a_K}\right]\left[E(s) - N(s)\right]$$

$$E-N \longrightarrow \boxed{\frac{1}{s-a_K}} \longrightarrow X_K$$




*Noise is attenuated in $x_K(t)$ if $\underbrace{|a_K|}_{\text{compensator pole}} \ll \underbrace{\omega_N}_{\text{noise frequency}}$.*

# Design implication, III

For good Noise rejection, compensator poles should be significantly lower frequency than the Noise

$\Rightarrow$ Avoid excessively high frequency poles in $H(s)$ (ie. poles very far from imag Axis).

$\Rightarrow$ Another advantage of "minimum $\beta$" lead comp design:

By minimizing $\beta$ (ratio of pole location to zero location in $H(s)$), we are bringing the pole as close to imag Axis as possible while still providing necessary $\varphi_{req}$ at desired $\omega_x$.

# Why it's bad to differentiate $y(t)$.

One is tempted to implement a $H(s)$ with only a zero (or more generally with 1 more zero than pole) by numerically differentiating $y(t)$ $\approx \dot{y}(t)$

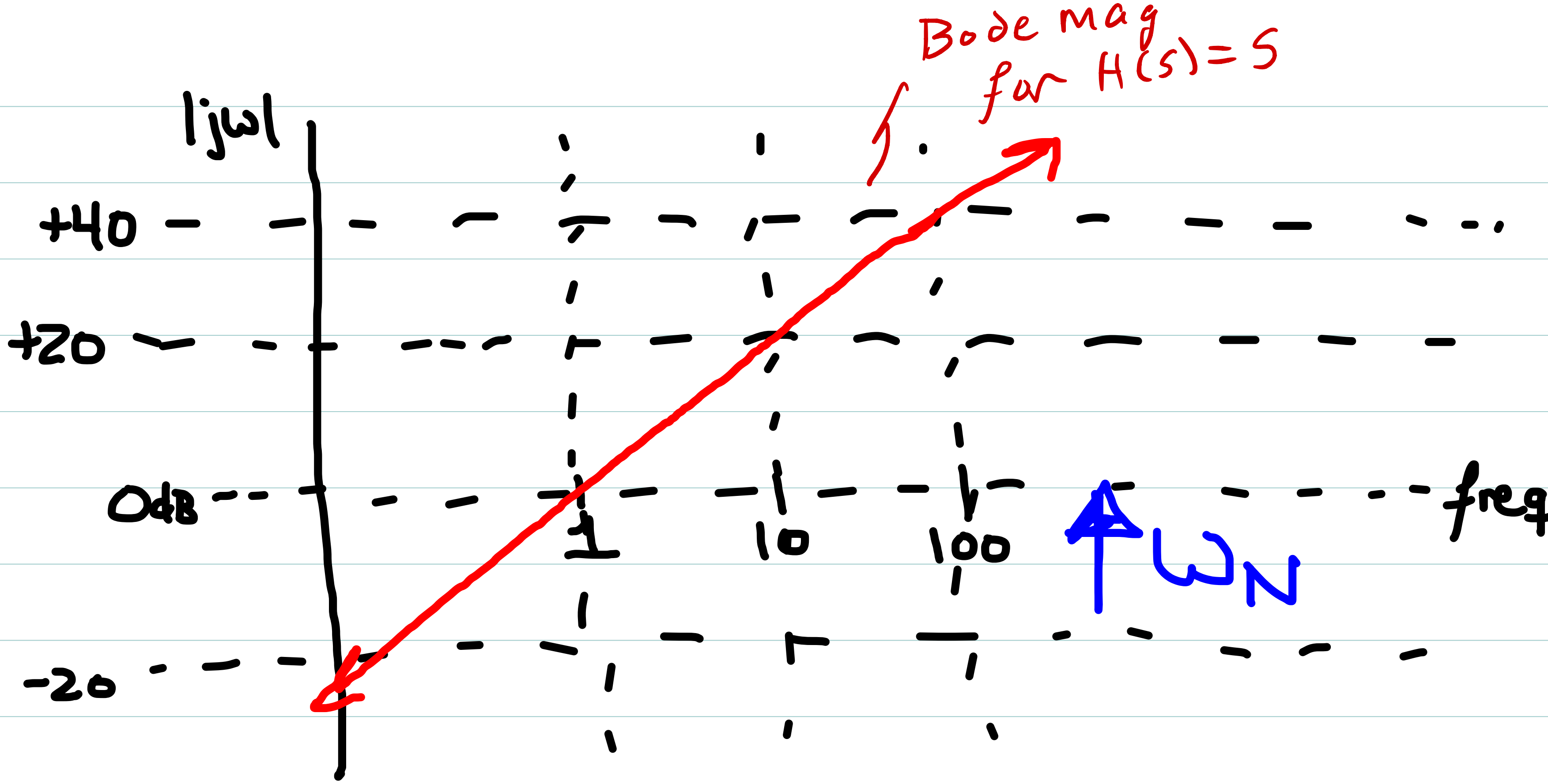
This would be needed since, as we've seen, such compensators will result in $u(t)$ having a term proportional to $\dot{e}(t)$ [hence $\dot{y}(t)$]

But with noise, we're really diff'ing $y_m(t) = y(t) + n(t)$.

Let $z(t) = \frac{d}{dt} y_m(t)$ be an estimate of $\dot{y}(t)$

$$\Rightarrow Z(s) = s\left[Y(s) + N(s)\right] \qquad \xrightarrow{Y+N} \boxed{s} \xrightarrow{Z}$$

Impact of noise depends on freq. response of $s$.




Differentiation amplifies the effect of noise

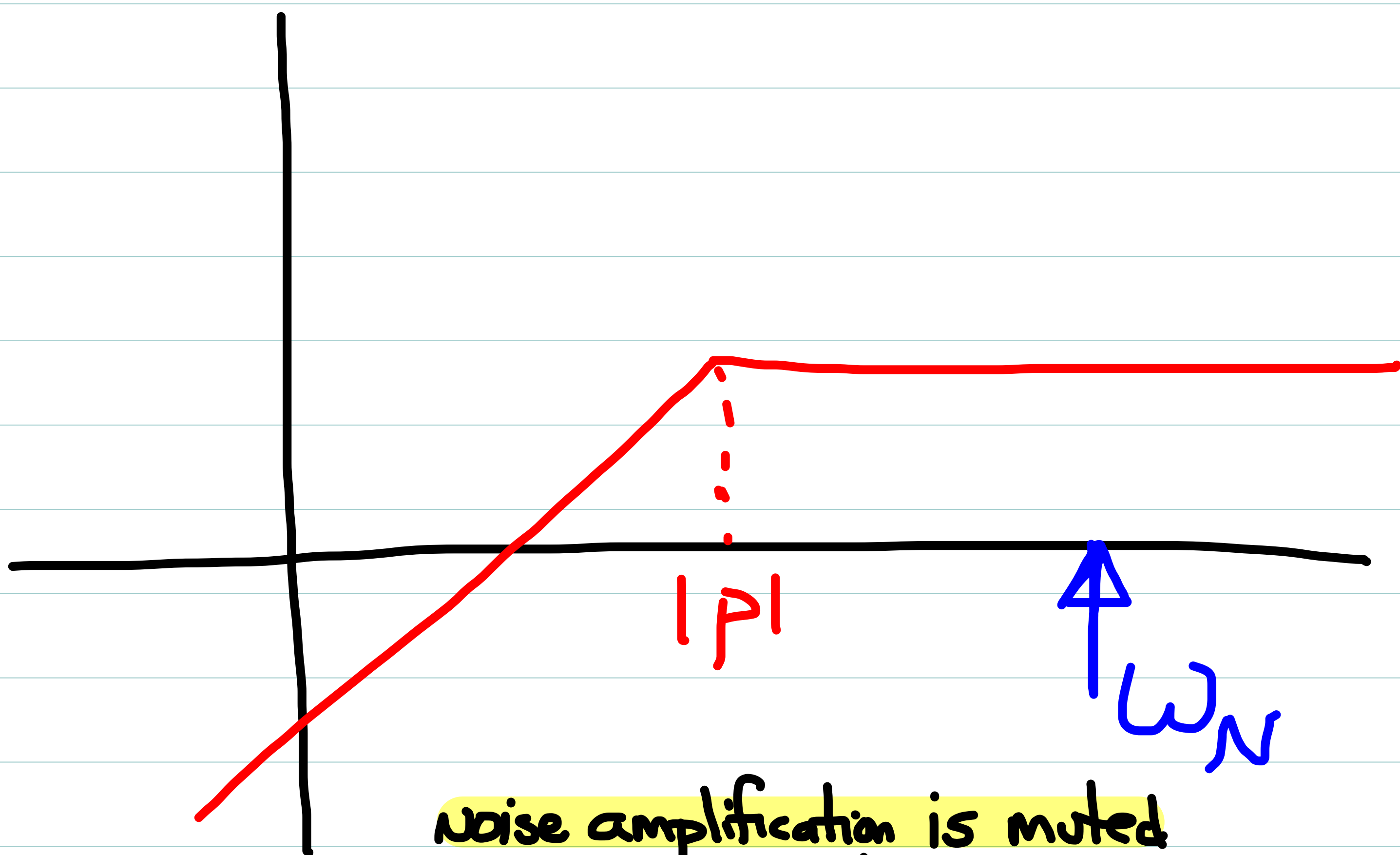
explicitly: if again $n(t) = \varepsilon \sin(\omega_N t)$, $\omega_N \gg 1$
then

$$z(t) = \frac{d}{dt}\left[y(t) + n(t)\right] = \dot{y}(t) + \underbrace{\varepsilon \omega_N}_{\text{Not small!}} \cos(\omega_N t)$$

(potentially larger than $\dot{y}$)

Note that if we added a pole to our derivative estimation scheme

$$Z(s) = \left[\frac{s}{s-p}\right] Y_m(s)$$




Noise amplification is muted and may be tolerable.

If we used this strategy to replace the derivative information needed for implementation an ideal zero:

$$H(s) = K(s-z) \implies H(s) = K\left[\frac{s}{s-p} - z\right]$$

Then:

$$H(s) = K\left[\frac{(1-z)s + pz}{s-p}\right]$$

which is a <u>lead compensator</u> (for typical case $p<z$).

So really, a lead compensator is effectively a "practical" implementation of an ideal zero, which acknowledges the imperfect nature of the measurement process.

Alt: a lead comp is a PD with velocity measurements replaced by a low pass filtered <u>estimate</u> of velocity.

The most basic (and essential) task of the control engineer — achieving a stable closed-loop system with nominal performance characteristics — is straightforward to approach.

However, it is tricky to also incorporate and balance the competing constraints of

- Implementation constraints (relative degree of $H(s)$)
- Tracking accuracy
- Disturbance rejection
- Noise rejection
- Model uncertainty
- Sensor/Actuator/Computation delays
- Actuator Limits/Control saturation
- Power/weight/cost demands

The "best" design is one which achieves an acceptable trade-off among these competing factors.

There is no "one true design" which makes the "ideal" tradeoff — so don't waste time looking for it!

Find something that works acceptably well, and move on

# Major, common families of compensators

① $H(s) = K \Rightarrow u(t) = K e(t)$ "Proportional" control

② $H(s) = K_P + K_D s = K(s - z)$ $\left(K = K_D,\ z = -K_P/K_D\right)$

$\Rightarrow u(t) = K_P e(t) + K_D \dot{e}(t)$ "Prop. + Derivative (PD) control"

<u>Note</u>: implementable if <u>both</u> $y(t)$ and $\dot{y}(t)$ measured directly)

③ $H(s) = K_P + \frac{K_I}{s} = K\left[\frac{s - z}{s}\right]$ $\left(K = K_P,\ z = -K_I/K_P\right)$

$\Rightarrow u(t) = K_P e(t) + K_I x_1(t)$
$\dot{x}_1(t) = e(t)$

Equivalently: $u(t) = K_P e(t) + K_I \int_0^t e(\tau)\, d\tau$
"prop. + integral (PI) control")

4. $H(s) = K_P + K_D s + K_I/s = K\left[\frac{(s-z_1)(s-z_2)}{s}\right]$

$(K = K_D\,;\ z_1, z_2$ roots of $K_D s^2 + K_P s + K_I)$

$\Rightarrow u(t) = K_P e(t) + K_D \dot{e}(t) + K_I \int_0^t e(\tau)\,d\tau$

"Prop/Int/Deriv (PID) control"

Notes: a.) Very popular. Special purpose chips which do this computation are commonly available

b.) 1) - 3) above are special cases of this more general form.

c.) Provides 2 zeros to help meet margin/xover requirements, and pole at origin to help with tracking/dist. rejection requirements.

d.) Like PD, requires direct measurement of $\dot{y}(t)$

⑤ $H(s) = K\left[\frac{(s-z)}{s-p}\right]$, $|z|<|p|$

"Lead compensator"

Notes: a.) "Implementable" form of PD control when only $y(t)$ measured

b.) Using minimal values of $\beta = |p|/|z|$ helps with noise rejection and control saturation

⑥ $H(s) = K\left[\frac{(s-z_1)(s-z_2)}{s(s-p)}\right]$ $|p|>|z_1|,|z_2|$

"PI/Lead": "implementable" form of PID when only $y(t)$ measured

---

Of course, a designer is free to choose $H(s)$ as desired. These are common "go to" starting points which can be modified or added to as needed.