

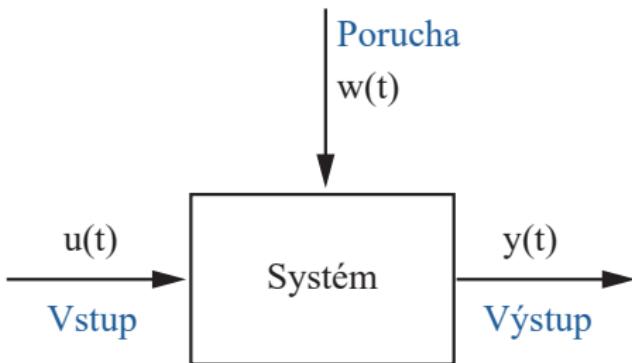
Working title drone control

Riadenie dronov

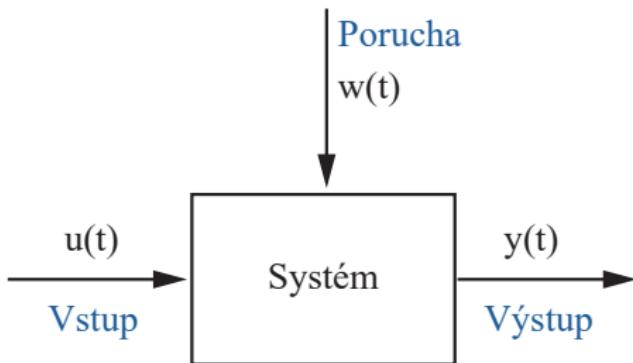
prof. Ing. Gergely Takács, PhD.



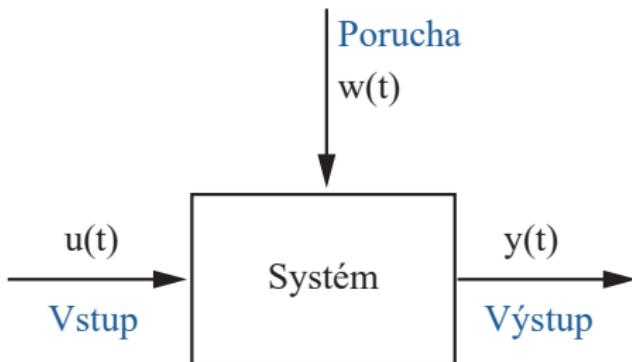
- Riadená sústava, systém alebo proces (*angl.*: plant, system, process), napr. motory drona
- Vstup $u(t)$ (*angl.*: input) sú akčné zásahy, napr. PWM do motorov
- Výstup $y(t)$ (*angl.*: output) je meraná, tzv. manipulovaná veličina (*angl.*: manipulated variable); napr. klopenie $\theta(t)$ drona
- Porucha (*angl.*: disturbance) $w(t)$ je vplyv vonkajšieho prostredia, napr. vietor



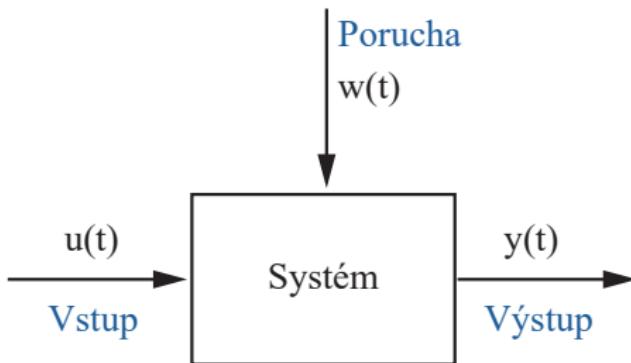
- Riadená sústava, systém alebo proces (*angl.*: plant, system, process), napr. motory drona
- Vstup $u(t)$ (*angl.*: input) sú akčné zásahy, napr. PWM do motorov
- Výstup $y(t)$ (*angl.*: output) je meraná, tzv. manipulovaná veličina (*angl.*: manipulated variable); napr. klopenie $\theta(t)$ drona
- Porucha (*angl.*: disturbance) $w(t)$ je vplyv vonkajšieho prostredia, napr. vietor



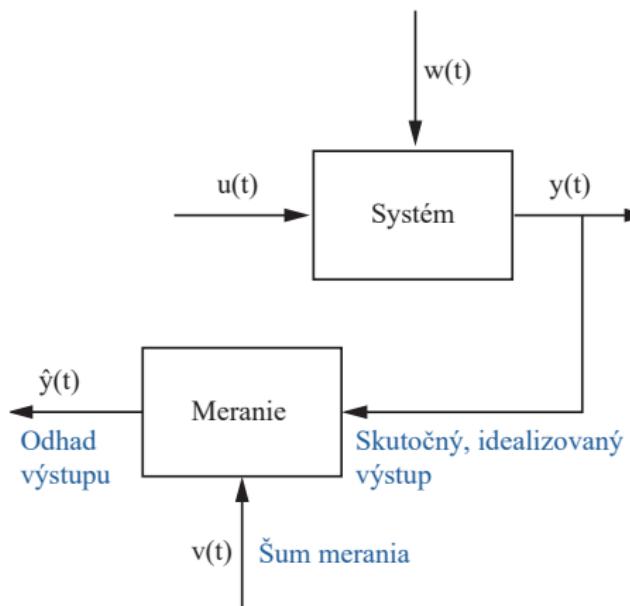
- Riadená sústava, systém alebo proces (*angl.*: plant, system, process), napr. motory drona
- Vstup $u(t)$ (*angl.*: input) sú akčné zásahy, napr. PWM do motorov
- Výstup $y(t)$ (*angl.*: output) je meraná, tzv. manipulovaná veličina (*angl.*: manipulated variable); napr. klopenie $\theta(t)$ drona
- Porucha (*angl.*: disturbance) $w(t)$ je vplyv vonkajšieho prostredia, napr. vietor



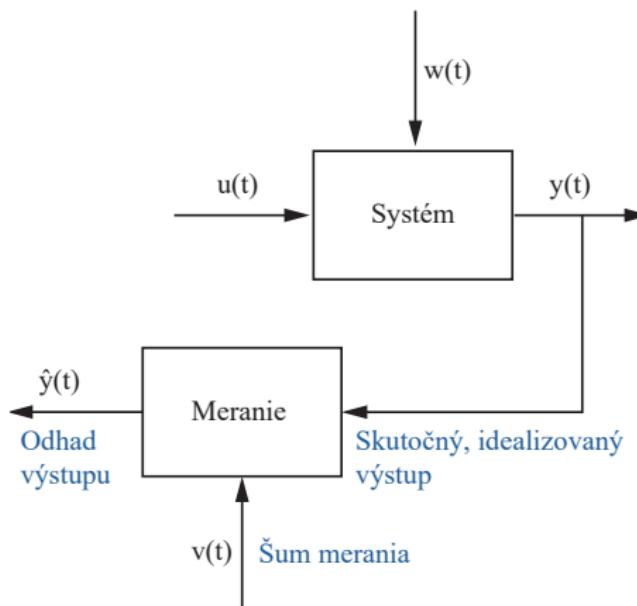
- Riadená sústava, systém alebo proces (*angl.*: plant, system, process), napr. motory drona
- Vstup $u(t)$ (*angl.*: input) sú akčné zásahy, napr. PWM do motorov
- Výstup $y(t)$ (*angl.*: output) je meraná, tzv. manipulovaná veličina (*angl.*: manipulated variable); napr. klopenie $\theta(t)$ drona
- Porucha (*angl.*: disturbance) $w(t)$ je vplyv vonkajšieho prostredia, napr. vietor



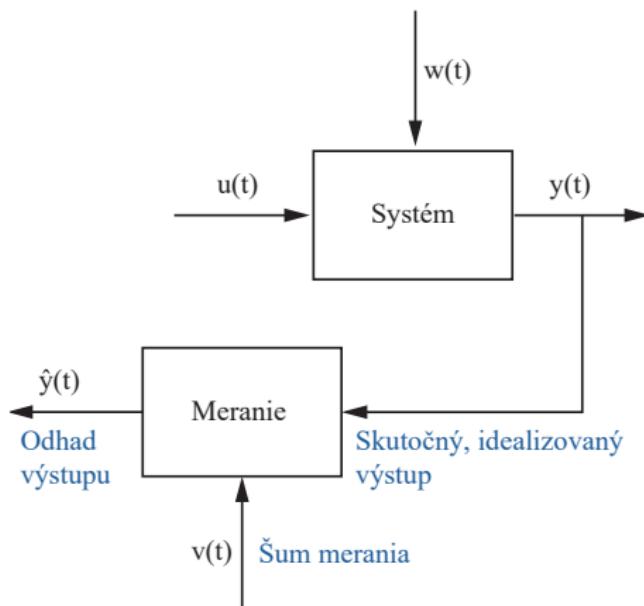
- Výstupy $y(t)$ sú idealizované
- Poruchy, teda šum merania $v(t)$ a dynamika snímača vplýva na výsledok
- Merania môžeme korigovať, resp. nemerané veličiny odhadnúť $\hat{y}(t)$
- Pre jednoduchosť často predpokladáme ideálne meranie $y(t) = \hat{y}(t)$



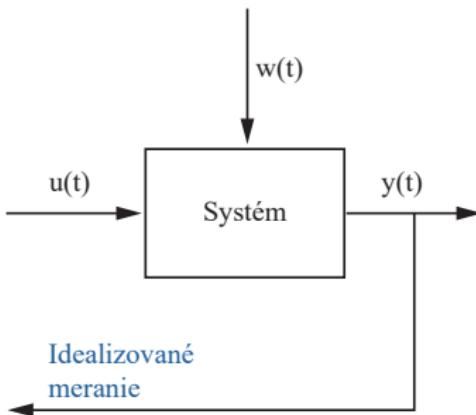
- Výstupy $y(t)$ sú idealizované
- Poruchy, teda šum merania $v(t)$ a dynamika snímača vplýva na výsledok
- Merania môžeme korigovať, resp. nemerané veličiny odhadnúť $\hat{y}(t)$
- Pre jednoduchosť často predpokladáme ideálne meranie $y(t) = \hat{y}(t)$



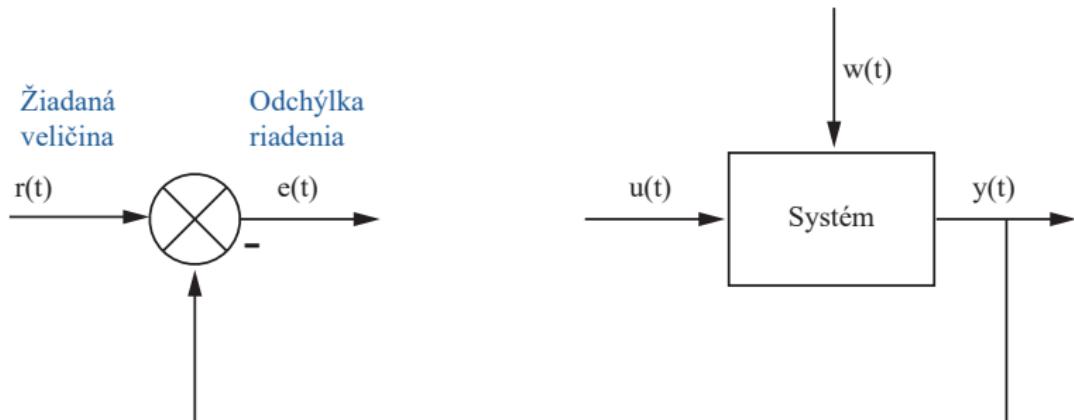
- Výstupy $y(t)$ sú idealizované
- Poruchy, teda šum merania $v(t)$ a dynamika snímača vplýva na výsledok
- Merania môžeme korigovať, resp. nemerané veličiny odhadnúť $\hat{y}(t)$
- Pre jednoduchosť často predpokladáme ideálne meranie $y(t) = \hat{y}(t)$



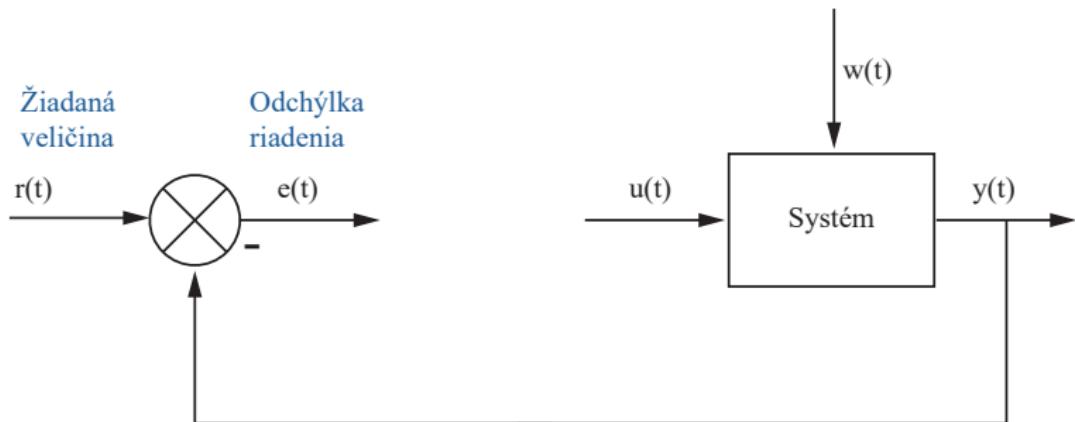
- Výstupy $y(t)$ sú idealizované
- Poruchy, teda šum merania $v(t)$ a dynamika snímača vplýva na výsledok
- Merania môžeme korigovať, resp. nemerané veličiny odhadnúť $\hat{y}(t)$
- Pre jednoduchosť často predpokladáme ideálne meranie $y(t) = \hat{y}(t)$



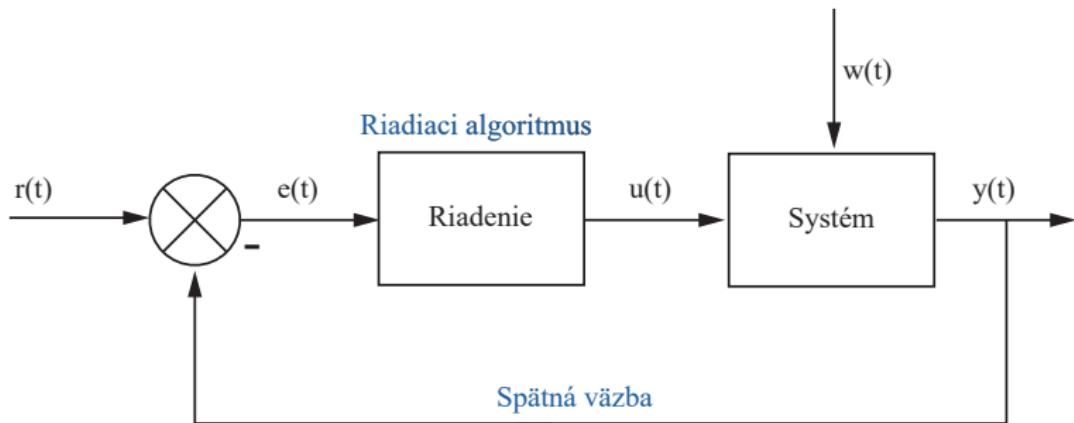
- Žiadaná hodnota alebo referencia (*angl.*: setpoint, reference) $r(t)$ vyjadruje na akú hodnotu chceme dostať výstup, potom
- Rozdiel medzi žiadanou hodnotou a výstupom $e(t) = r(t) - y(t)$ je odchýlka riadenia (*angl.*: error)



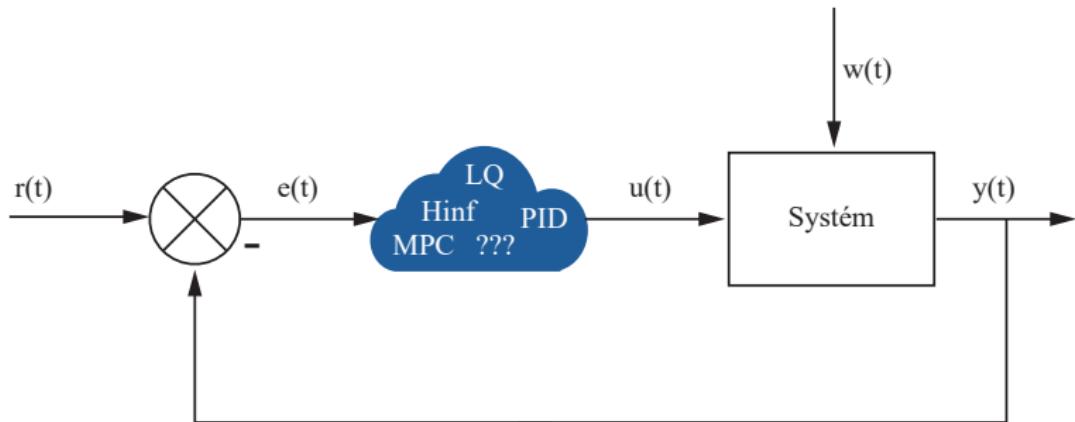
- Žiadaná hodnota alebo referencia (*angl.*: setpoint, reference) $r(t)$ vyjadruje na akú hodnotu chceme dostať výstup, potom
- Rozdiel medzi žiadanou hodnotou a výstupom $e(t) = r(t) - y(t)$ je odchýlka riadenia (*angl.*: error)



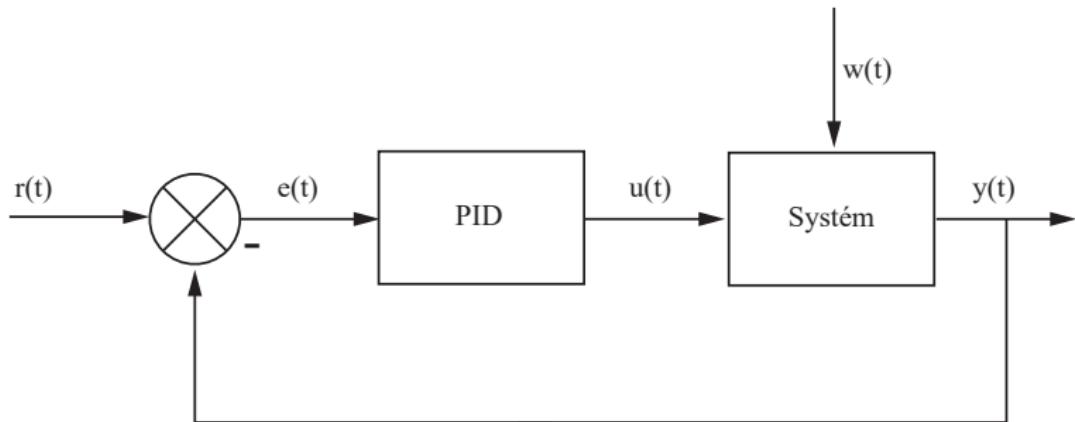
- Riadenie je logika ktorá prepočíta odchýlku riadenia $e(t)$ na vstupy $u(t)$
- Môže to byť ľubovoľné, do konca aj analógové a mechanické (e.g. parné stroje)
- Dobrým kompromisom medzi komplexnosťou (pochopenie, návrh) a implementovateľnosťou je PID riadenie
- Asi 95% riadiacich algoritmov v praxi sú PID...
- ...a väčšina z nich sú slabo naladené.



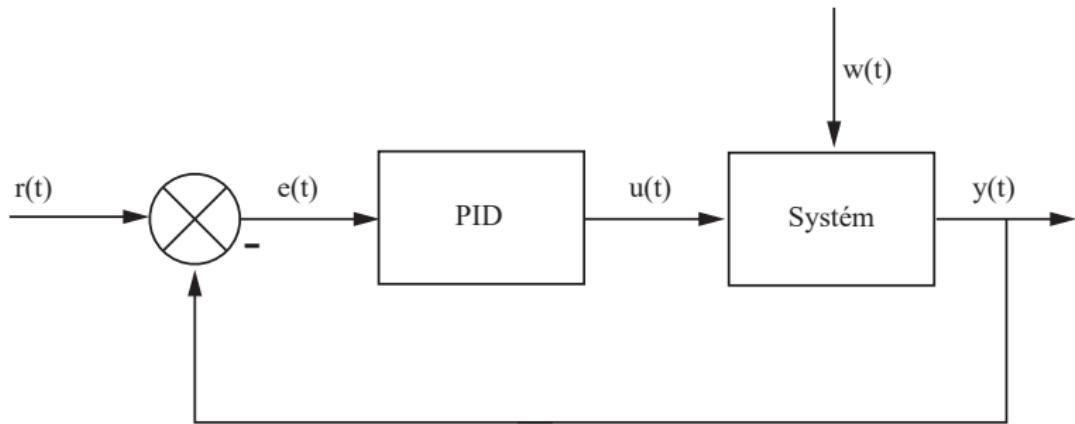
- Riadenie je logika ktorá prepočíta odchýlku riadenia $e(t)$ na vstupy $u(t)$
- Môže to byť ľubovoľné, do konca aj analógové a mechanické (e.g. parné stroje)
- Dobrým kompromisom medzi komplexnosťou (pochopenie, návrh) a implementovateľnosťou je PID riadenie
- Asi 95% riadiacich algoritmov v praxi sú PID...
- ...a väčšina z nich sú slabo naladené.



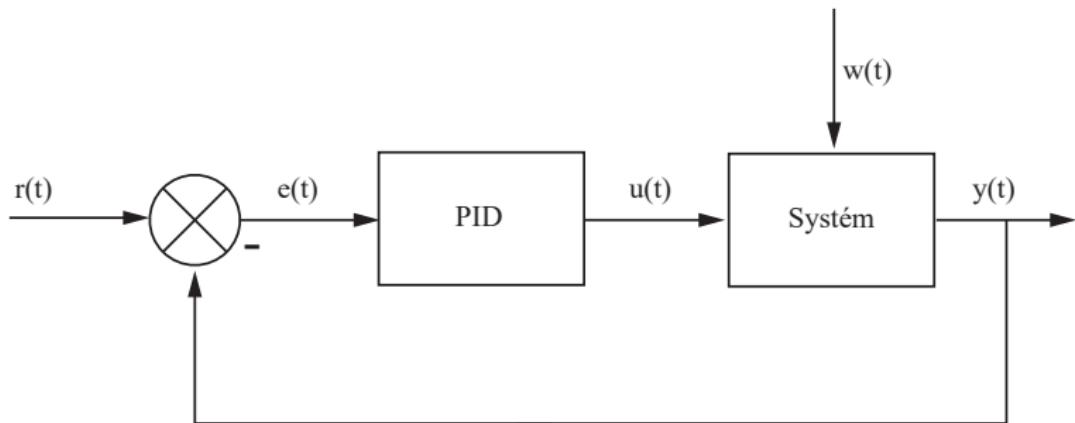
- Riadenie je logika ktorá prepočíta odchýlku riadenia $e(t)$ na vstupy $u(t)$
- Môže to byť ľubovoľné, do konca aj analógové a mechanické (e.g. parné stroje)
- Dobrým kompromisom medzi komplexnosťou (pochopenie, návrh) a implementovateľnosťou je PID riadenie
- Asi 95% riadiacich algoritmov v praxi sú PID...
- ...a väčšina z nich sú slabo naladené.



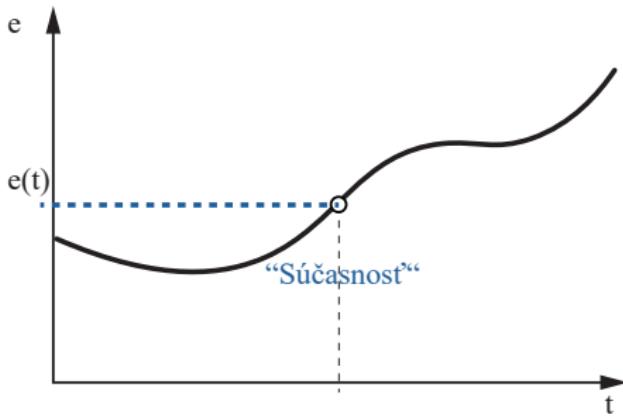
- Riadenie je logika ktorá prepočíta odchýlku riadenia $e(t)$ na vstupy $u(t)$
- Môže to byť ľubovoľné, do konca aj analógové a mechanické (e.g. parné stroje)
- Dobrým kompromisom medzi komplexnosťou (pochopenie, návrh) a implementovateľnosťou je PID riadenie
- Asi 95% riadiacich algoritmov v praxi sú PID...
- ...a väčšina z nich sú slabo naladené.



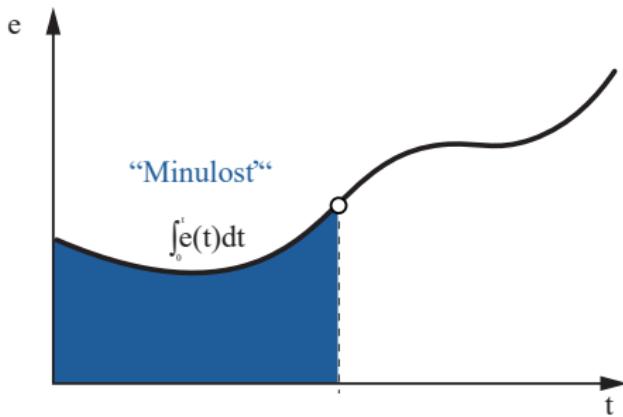
- Riadenie je logika ktorá prepočíta odchýlku riadenia $e(t)$ na vstupy $u(t)$
- Môže to byť ľubovoľné, do konca aj analógové a mechanické (e.g. parné stroje)
- Dobrým kompromisom medzi komplexnosťou (pochopenie, návrh) a implementovateľnosťou je PID riadenie
- Asi 95% riadiacich algoritmov v praxi sú PID...
- ...a väčšina z nich sú slabo naladené.



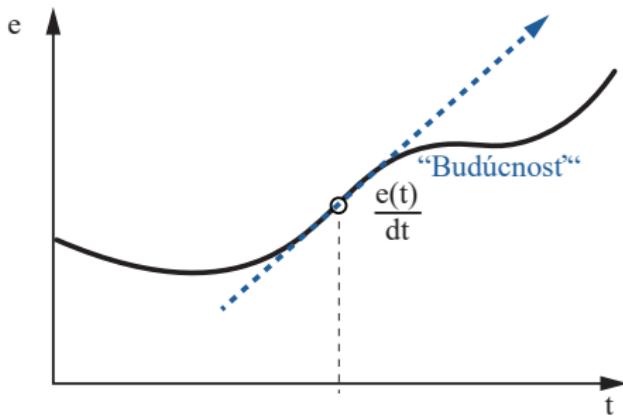
- **Súčasnosť** = proporcionálna zložka: okamžitá odchýlka $e(t)$, ale majme možnosť na ladenie, tak $K_P e(t)$.
- **Minulosť** = integračná zložka: celková odchýlka $e(t)$ v minulosti, ale majme možnosť na ladenie, tak $K_I \int_0^t e(t)dt$.
- **Budúcnosť** = derivačná zložka: projektovaná odchýlka $e(t)$ do budúcnosti, ale majme možnosť na ladenie, tak $K_D \frac{de(t)}{dt}$.



- **Súčasnosť** = proporcionálna zložka: okamžitá odchýlka $e(t)$, ale majme možnosť na ladenie, tak $K_P e(t)$.
- **Minulosť** = integračná zložka: celková odchýlka $e(t)$ v minulosti, ale majme možnosť na ladenie, tak $K_I \int_0^t e(t) dt$.
- **Budúcnosť** = derivačná zložka: projektovaná odchýlka $e(t)$ do budúcnosti, ale majme možnosť na ladenie, tak $K_D \frac{de(t)}{dt}$.



- **Súčasnosť** = proporcionálna zložka: okamžitá odchýlka $e(t)$, ale majme možnosť na ladenie, tak $K_P e(t)$.
- **Minulosť** = integračná zložka: celková odchýlka $e(t)$ v minulosti, ale majme možnosť na ladenie, tak $K_I \int_0^t e(t) dt$.
- **Budúcnosť** = derivačná zložka: projektovaná odchýlka $e(t)$ do budúcnosti, ale majme možnosť na ladenie, tak $K_D \frac{de(t)}{dt}$.



- **Súčasnosť** = proporcionálna zložka: okamžitá odchýlka $e(t)$, ale majme možnosť na ladenie, tak $K_P e(t)$.
- **Minulosť** = integračná zložka: celková odchýlka $e(t)$ v minulosti, ale majme možnosť na ladenie, tak $K_I \int_0^t e(t) dt$.
- **Budúcnosť** = derivačná zložka: projektovaná odchýlka $e(t)$ do budúcnosti, ale majme možnosť na ladenie, tak $K_D \frac{de(t)}{dt}$.
tzv. paralelná (ideálna) forma

$$u(t) = K_P e(t) + K_I \int_0^t e(t) dt + K_D \frac{de(t)}{dt} \quad (1)$$

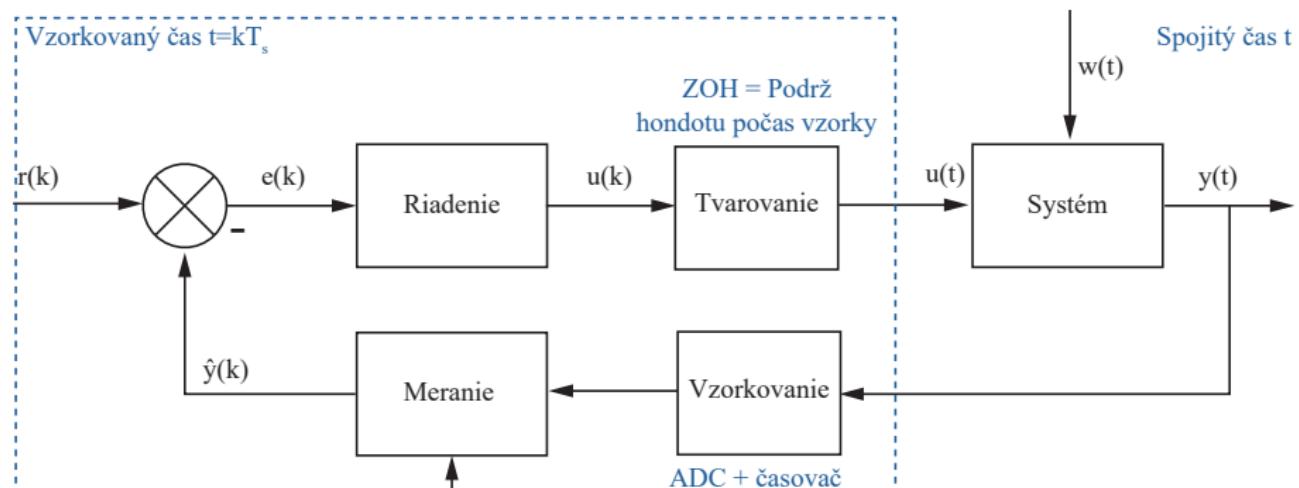
- **Súčasnosť** = proporcionálna zložka: okamžitá odchýlka $e(t)$, ale majme možnosť na ladenie, tak $K_P e(t)$.
- **Minulosť** = integračná zložka: celková odchýlka $e(t)$ v minulosti, ale majme možnosť na ladenie, tak $K_I \int_0^t e(t) dt$.
- **Budúcnosť** = derivačná zložka: projektovaná odchýlka $e(t)$ do budúcnosti, ale majme možnosť na ladenie, tak $K_D \frac{de(t)}{dt}$.
tzv. paralelná (ideálna) forma

$$u(t) = K_P e(t) + K_I \int_0^t e(t) dt + K_D \frac{de(t)}{dt} \quad (1)$$

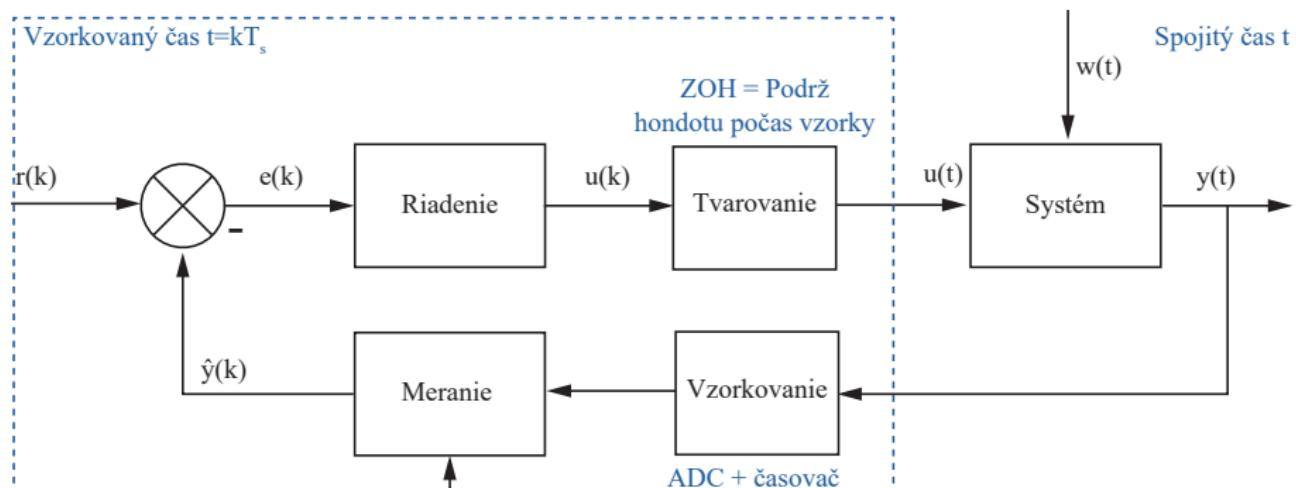
alebo tzv. štandardná forma (máme intuitívnejšie časové konštanty, ale zosilnenie je viazané)

$$u(t) = K_P \left(e(t) + \frac{1}{T_I} \int_0^t e(t) dt + T_D \frac{de(t)}{dt} \right) \quad (2)$$

- Výpočtová realizácia neumožní počítať spojito, preto
 - ▶ Vzorkujeme na strane výstupov — ADC + slučka v hard real-time pomocou časovačov
 - ▶ Tvarujeme na strane vstupov — zero order hold (ZOH) len znamená podržíme hodnotu počas vzorky
- Vzorkovanie mení spojity čas na $t = kT_s$. Periódna je voliteľná, ale
 - ▶ musí zachytiť dominantnú dynamiku riadeného dejha,
 - ▶ a výpočtová realizácia musí stíhať.



- Výpočtová realizácia neumožní počítať spojito, preto
 - ▶ Vzorkujeme na strane výstupov — ADC + slučka v hard real-time pomocou časovačov
 - ▶ Tvarujeme na strane vstupov — zero order hold (ZOH) len znamená podržíme hodnotu počas vzorky
- Vzorkovanie mení spojitý čas na $t = kT_s$. Periódna je voliteľná, ale
 - ▶ musí zachytiť dominantnú dynamiku riadeného dejha,
 - ▶ a výpočtová realizácia musí stíhať.



- Systém, resp. proces zostáva vždy spojity
- Koncepty ako integrácia, derivácia musíme numericky approximovať, t.j. ak T_s je vzorkovací čas, naše PID bude

$$u(k) = K_P e(k) + K_I \underbrace{\sum_0^t e(k) T_s}_{\text{"obdlzniky"}} + K_D \underbrace{\frac{e(k) - e(k-1)}{T_s}}_{\text{"vstupanie"}} \quad (3)$$

- alebo v prakticky vhodnejšej tzv. inkrementálnej forme pre MCU (bez dôkazu, vid'. Wikipedia 2021)

$$u(k) = u(k-1) + \left(K_P + K_I T_s + \frac{K_D}{T_s} \right) e(k) + \left(-K_P - 2 \frac{K_D}{T_s} \right) e(k-1) + \frac{K_D}{T_s} e(k-2) \quad (4)$$

- Systém, resp. proces zostáva vždy spojity
- Koncepty ako integrácia, derivácia musíme numericky approximovať, t.j. ak T_s je vzorkovací čas, naše PID bude

$$u(k) = K_P e(k) + K_I \underbrace{\sum_0^t e(k) T_s}_{\text{"obdlzniky"}} + K_D \underbrace{\frac{e(k) - e(k-1)}{T_s}}_{\text{"vstupanie"}} \quad (3)$$

- alebo v prakticky vhodnejšej tzv. inkrementálnej forme pre MCU (bez dôkazu, vid'. Wikipedia 2021)

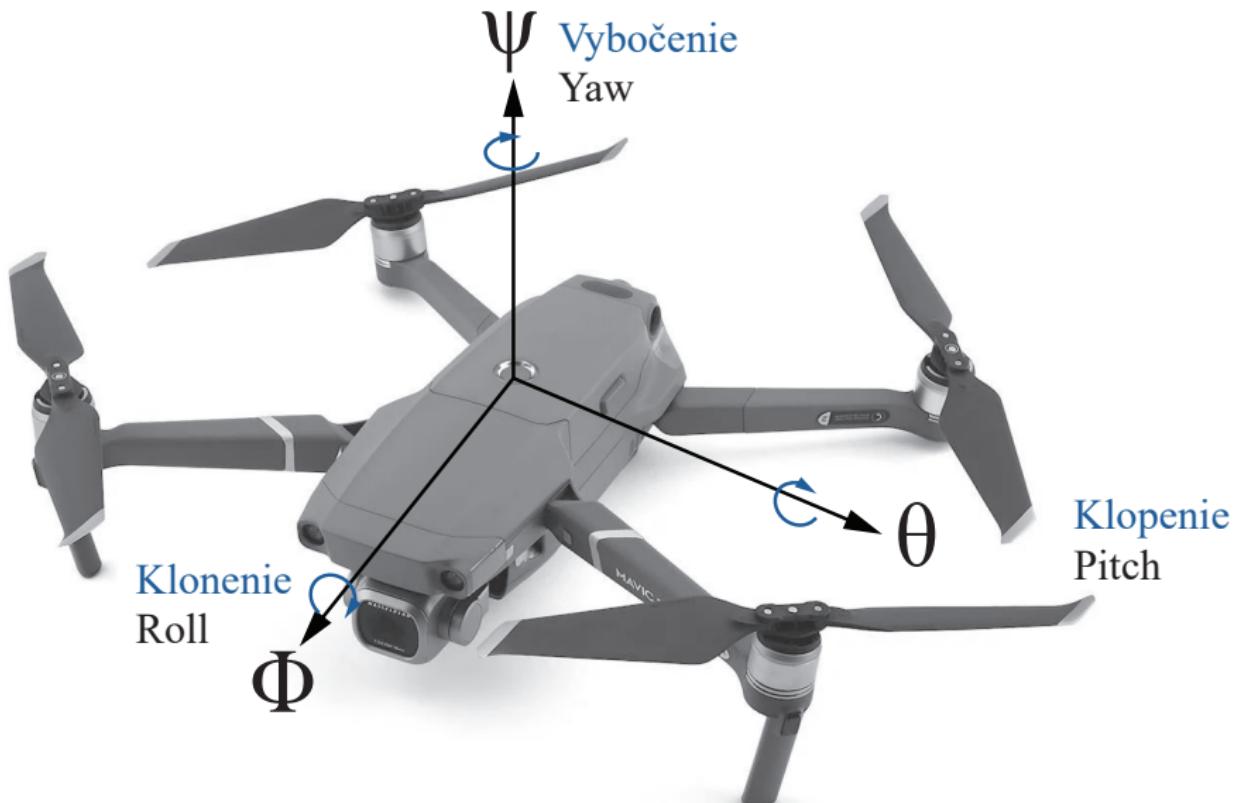
$$u(k) = u(k-1) + \left(K_P + K_I T_s + \frac{K_D}{T_s} \right) e(k) + \left(-K_P - 2 \frac{K_D}{T_s} \right) e(k-1) + \frac{K_D}{T_s} e(k-2) \quad (4)$$

- Systém, resp. proces zostáva vždy spojity
- Koncepty ako integrácia, derivácia musíme numericky approximovať, t.j. ak T_s je vzorkovací čas, naše PID bude

$$u(k) = K_P e(k) + K_I \underbrace{\sum_0^t e(k) T_s}_{\text{"obdlzniky"}} + K_D \underbrace{\frac{e(k) - e(k-1)}{T_s}}_{\text{"vstupanie"}} \quad (3)$$

- alebo v prakticky vhodnejšej tzv. inkrementálnej forme pre MCU (bez dôkazu, vid'. [Wikipedia 2021](#))

$$u(k) = u(k-1) + \left(K_P + K_I T_s + \frac{K_D}{T_s} \right) e(k) + \left(-K_P - 2 \frac{K_D}{T_s} \right) e(k-1) + \frac{K_D}{T_s} e(k-2) \quad (4)$$



Ďakujem za Vašu pozornosť.

- [1] Wikipedia. *PID controller*. [online]. [cited 25.11.2021]; Available from https://en.wikipedia.org/wiki/PID_controller. 2021.