

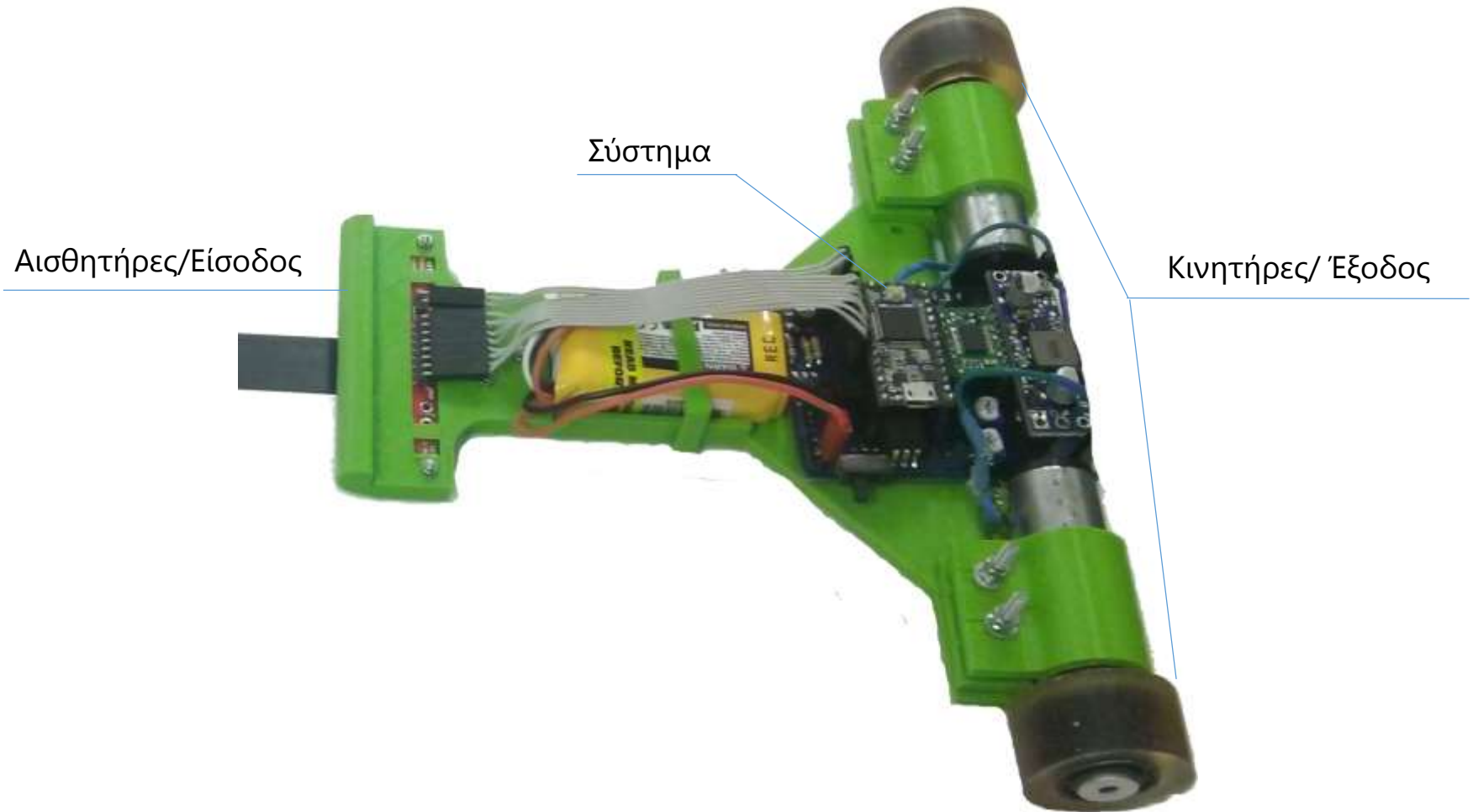
# Έλεγχος (PID-Control)

Γιάννης Παπακωνσταντινόπουλος

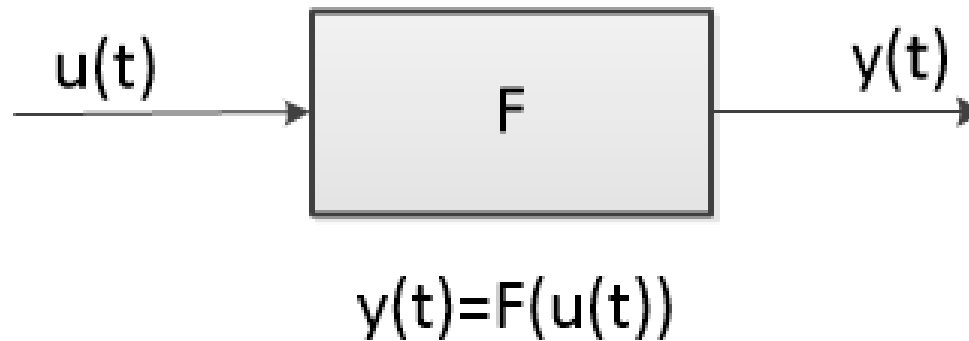
**Λέσχη Ρομποτικής**

20 Μαΐου 2016

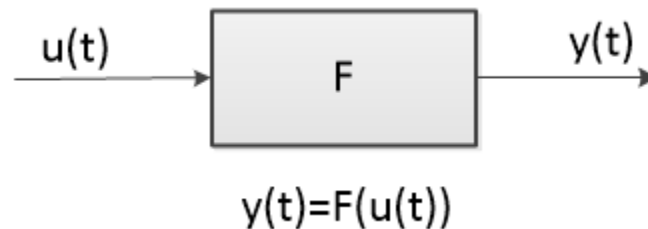
# Το ρομπότ σαν σύστημα



# Το ρομπότ σαν σύστημα



# Τι είναι σύστημα



**Σύστημα** είναι ένα σύνολο στοιχείων τα οποία είναι κατάλληλα συνδεδεμένα μεταξύ τους για να επιτελέσουν κάποιο έργο.

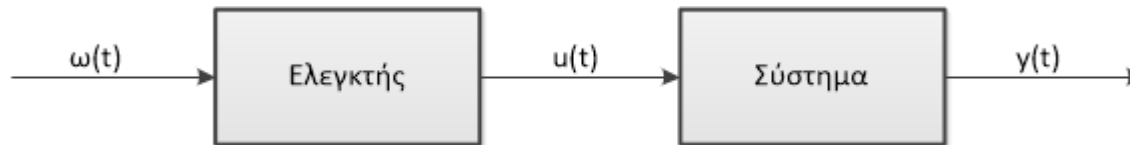
Για να φέρει όμως ένα σύστημα σε πέρας ένα έργο θα πρέπει να του δοθεί, η **κατάλληλη διέγερση**. Η απόκριση  $y(t)$  καλείται επίσης και **συμπεριφορά** του συστήματος.

# Βασικές κατηγορίες συστημάτων

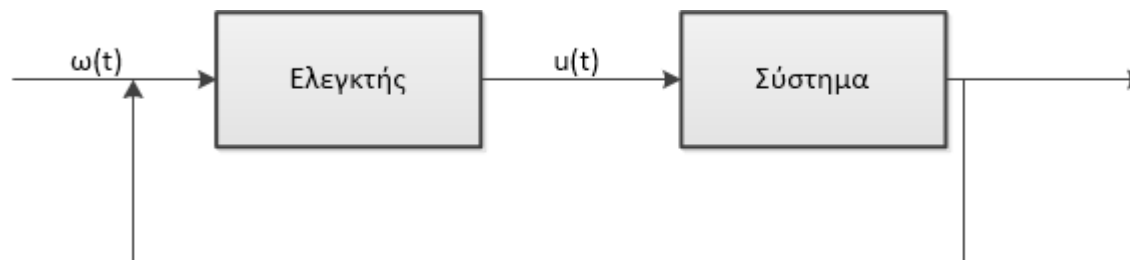
- **Ανάλογα με τη φύση των σημάτων εισόδου και εξόδου**
  - Συστήματα συνεχούς χρόνου
  - Συστήματα διακριτού χρόνου/Ψηφιακά συστήματα
  
- **Ανάλογα με τον τρόπο λειτουργίας των συστημάτων**
  - Στατικά
  - Δυναμικά
  - Χρονικά αμετάβλητα
  - Χρονικά μεταβαλλόμενα
  - Αιτιατά
  - Γραμμικά
  - Μη γραμμικά

# Ανοιχτά και κλειστά συστήματα

**Ανοιχτό σύστημα** είναι ένα σύστημα που η είσοδος  $u(t)$  δεν είναι συνάρτηση της εξόδου  $y(t)$ .

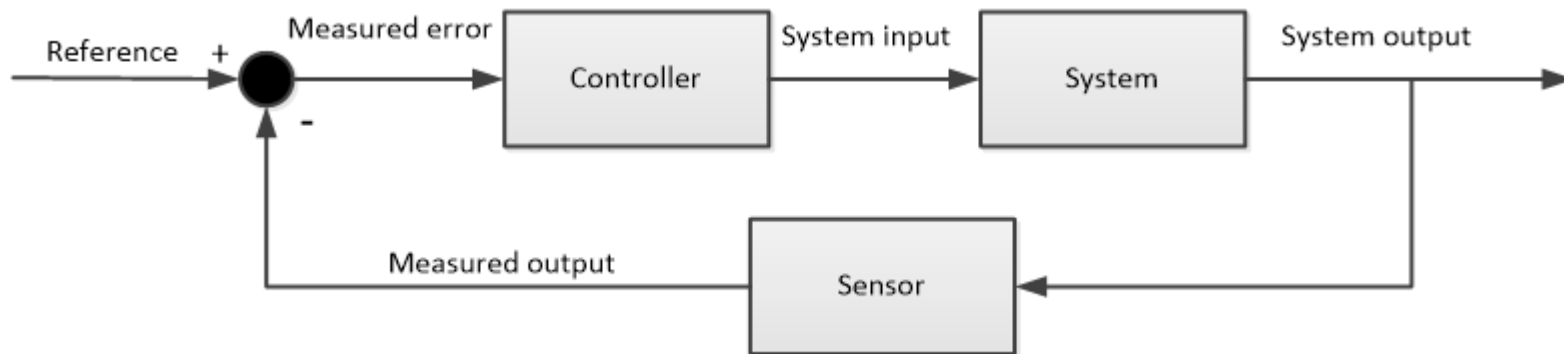


**Κλειστό σύστημα** είναι ένα σύστημα που η είσοδος  $u(t)$  είναι συνάρτηση της εξόδου  $y(t)$ .



# Κλειστό σύστημα και ελεγκτής(ClosedLoop)

Δυναμικό σύστημα με ανάδραση.



Reference: Σημείο αναφοράς(setpoint)

**Error = desired output (reference) - actual output**

# Αυτόματος έλεγχος

- Κύρια θέματα:
  - Σταθερότητα
  - Ταλάντωση
  - Ελεγχιμότητα
  - Παρατηρησιμότητα

## Συνάρτηση μεταφοράς $H(s)$

Μαθηματική αναπαράσταση της σχέσης μεταξύ της εισόδου και εξόδου, που περιγράφει το σύστημα.

$$s = \sigma + j\omega \quad , \quad X(s) = L \{ x(t) \} \quad , \quad Y(s) = L \{ y(t) \}$$

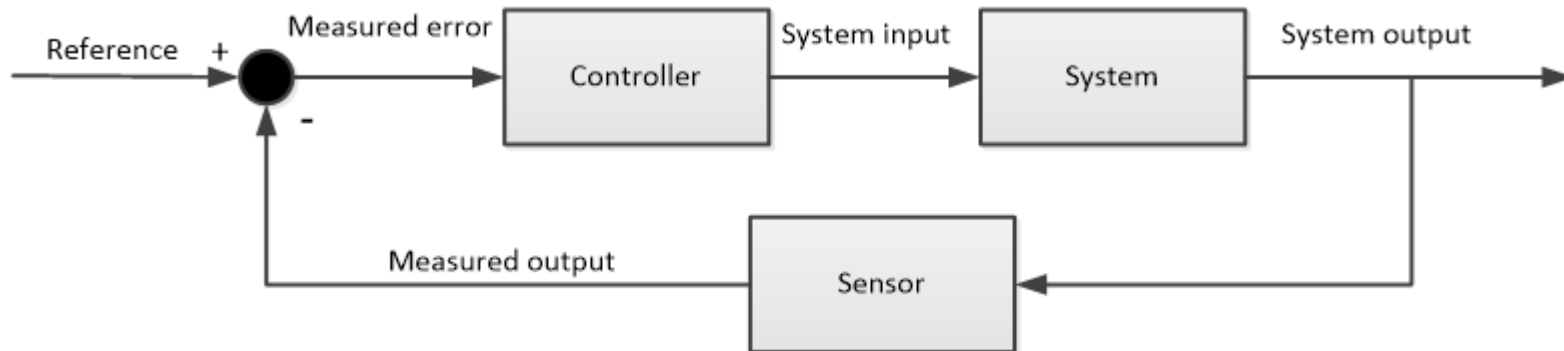
$$x(t) = h(t) * x(t) \quad \Leftrightarrow \quad Y(s) = H(s) X(s)$$



# Αυτόματος έλεγχος

Control =

1. Measure
2. Compare
3. Compute
4. Correct



# OpenLoop vs ClosedLoop(Cruise Control )

**Στόχος :** Διατήρηση της ταχύτητας του αυτοκινήτου σε μια καθορισμένη από τον οδηγό τιμή

Plant: Αυτοκίνητο

Reference: Καθορισμένη ταχύτητα

Output: Γκάζι

Sensing: Πραγματική ταχύτητα

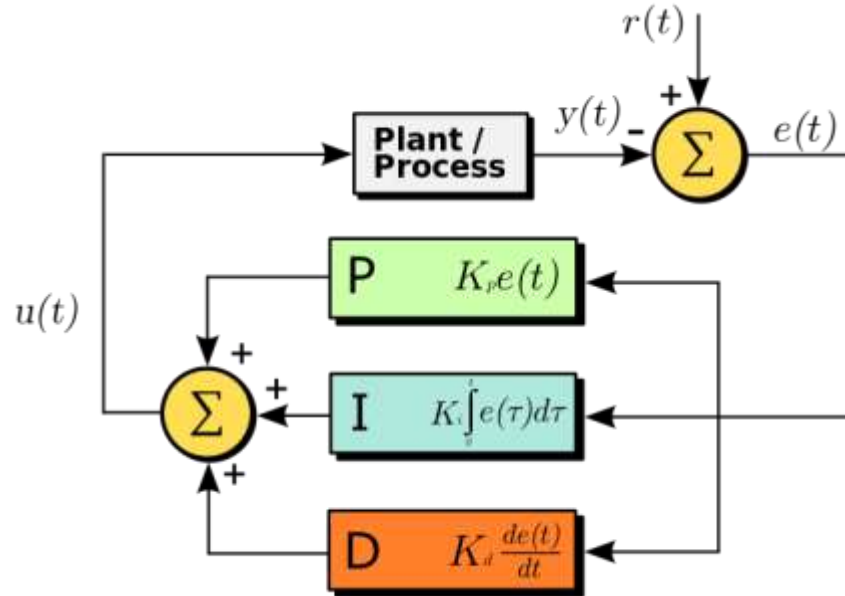
Error = reference - actual speed

OpenLoop: Κλειδώνει το γκάζι σε μια συγκεκριμένη θέση χωρίς να ξέρουμε την πραγματική ταχύτητα (no feedback)

ClosedLoop: Λήψη δεδομένων από το ταχύμετρο, μας επιστρέφει συνεχώς την πραγματική ταχύτητα την οποία αφαιρούμε από το reference , και το αποτέλεσμα καθορίζει τη θέση του γκαζιού ανάλογα με το σφάλμα

# PID Controller

## Proportional – Integral – Derivative Control loop feedback mechanism



Η μεταβλητή ( $e$ ) αντιπροσωπεύει το **σφάλμα παρακολούθησης**, δηλαδή την διαφορά ανάμεσα στην τιμή της επιθυμητής εισόδου ( $R$ ) και σε εκείνη της πραγματικής εξόδου ( $Y$ ).

**Ένας ελεγκτής PID υπολογίζει μια τιμή σφάλματος ως η διαφορά ανάμεσα σε μια μετρούμενη μεταβλητή της διαδικασίας και ένα επιθυμητό σημείο.**

# PID

---

- **P** Εξαρτάται από το σφάλμα στο παρόν  
Η χρησιμοποίηση ενός αναλογικού ελεγκτή - P -, θα έχει ως αποτέλεσμα την **ελάττωση** του **χρόνου ανύψωσης**, αλλά ποτέ την εξάλειψη του μόνιμου σφάλματος.
- **I** Εξαρτάται από τη συσσώρευση των σφαλμάτων του παρελθόντος  
Ο ολοκληρωτικός έλεγχος - I - θα **εξαλείψει το μόνιμο σφάλμα**, αλλά θα χειροτερέψει την μεταβατική Απόκριση (αριθμός των ταλαντώσεων μέχρι την τελική ισορροπία του συστήματος).
- **D** Είναι μια πρόβλεψη των μελλοντικών σφαλμάτων  
Ο διαφορικός έλεγχος - D - θα έχει ως αποτέλεσμα την **αύξηση της σταθερότητας** του συστήματος, μειώνοντας την υπερύψωση και βελτιώνοντας την μεταβατική Απόκριση.

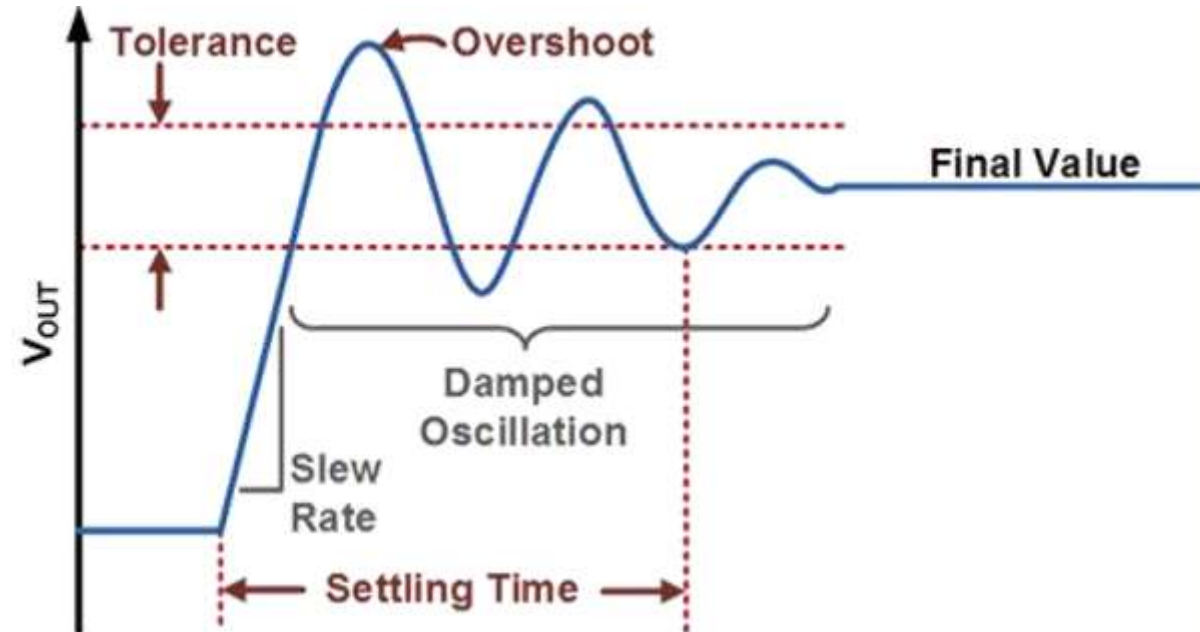
## Κέρδη $K_p$ , $K_i$ , $K_d$

Οι όροι  $K_p$ ,  $K_i$  και  $K_d$  είναι αντίστοιχα:

- $K_p$  = Αναλογικό κέρδος
- $K_i$  = Ολοκληρωτικό κέρδος
- $K_d$  = Διαφορικό κέρδος

$$u(t) = K_p e(t) + K_i \int_0^t e(\tau) d\tau + K_d \frac{de(t)}{dt}$$

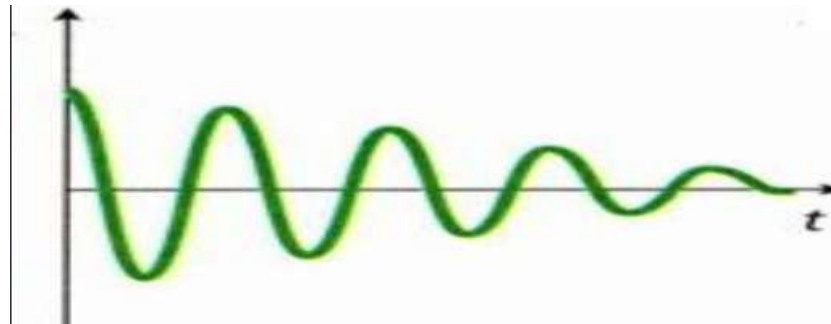
## PID (Overshoot , Oscillation)



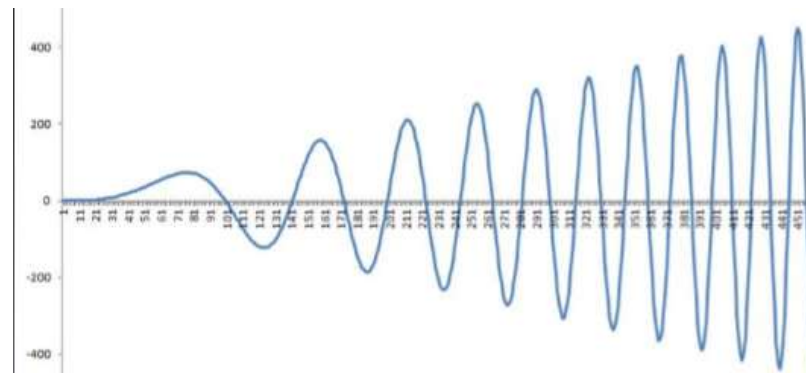
- Κάνοντας μεγάλη αλλαγή σε μικρό σφάλμα έχουμε υπερακόντιση (**Overshoot**)
- Αν ο ελεγκτής συνεχίσει την υπερακόντιση , η έξοδος θα κάνει ταλάντωση γύρω από το setpoint. (**Oscillation**)

# PID (Stable , Unstable)

Αν το oscillation μειώνεται στην πάροδο του χρόνου το σύστημα είναι **σταθερό**



Αν το oscillation αυξάνεται στην πάροδο του χρόνου το σύστημα είναι **ασταθές**



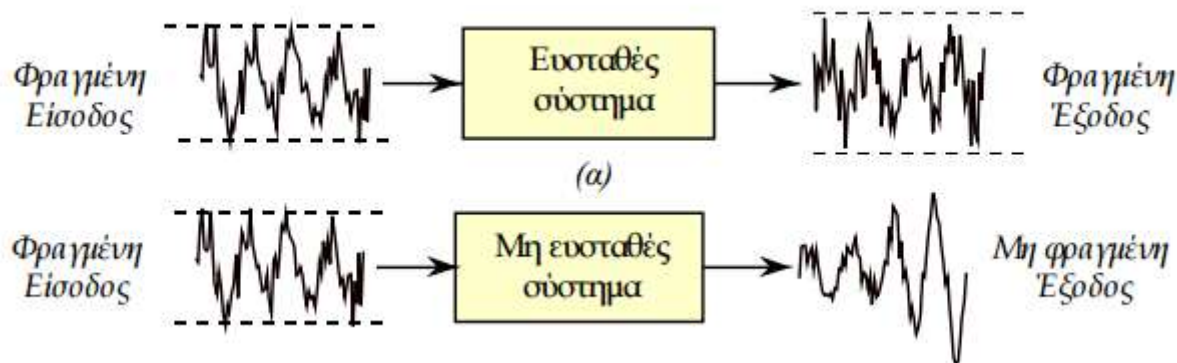
# Αστάθεια

## ΦΕΦΕ ευστάθεια

αν για κάθε θετικό αριθμό  $M1 < \infty$  για τον οποίο ισχύει  $|x(t)| \leq M1$  υπάρχει θετικός αριθμός  $M2 < \infty$  για τον οποίο ισχύει  $|y(t)| \leq M2$

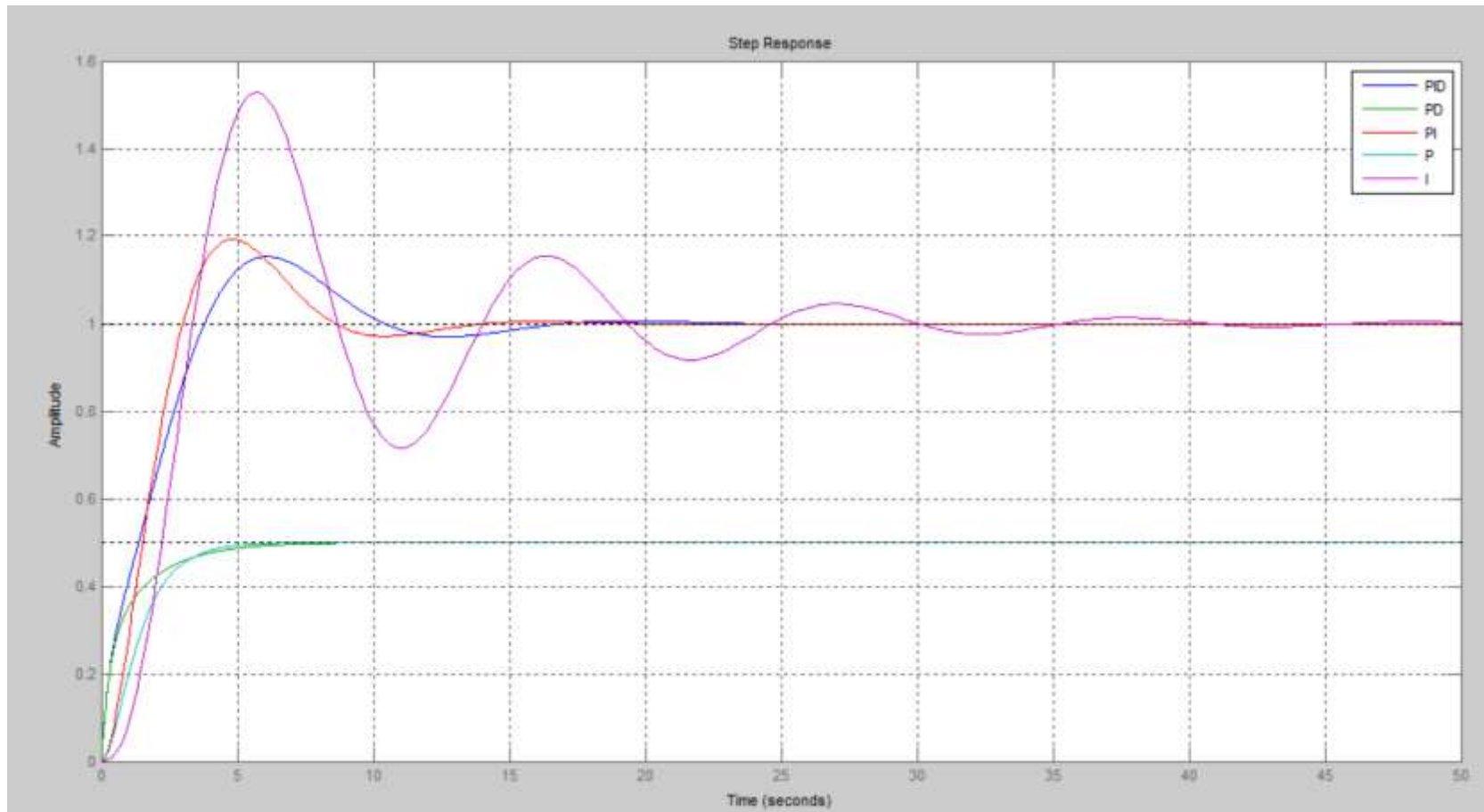
Καθορίζεται από τις ρίζες της συνάρτησης μεταφοράς  $H(s)$ .

Ένα **αιτιατό** σύστημα είναι ΦΕΦΕ ευσταθές όταν οι **πόλοι** της συνάρτησης μεταφοράς είναι **αρνητικοί**.

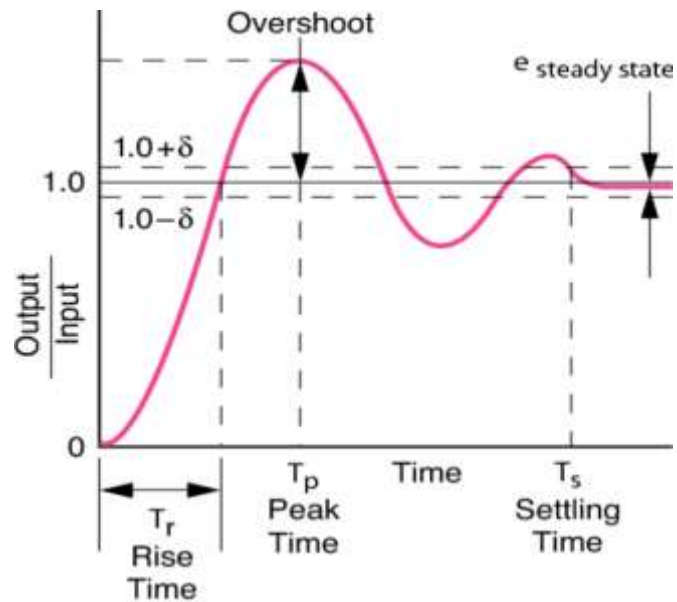




# Παράδειγμα PID



# Πως ρυθμίζουμε τις παραμέτρους



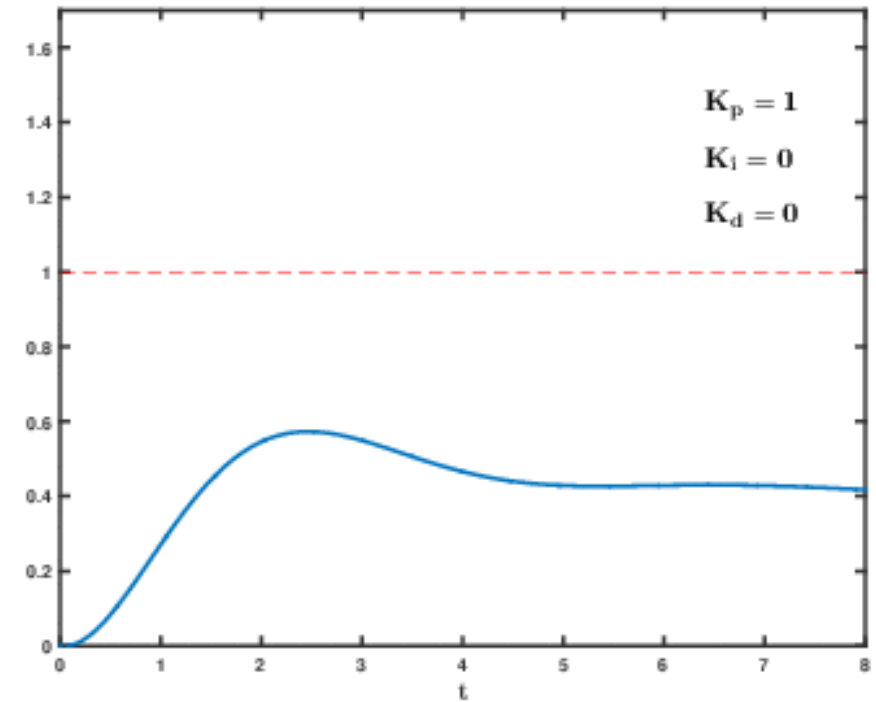
Parameter Increase	Rise time	Overshoot	Settling Time	Steady-state error
<b>Kp</b>	↓	↑	Small Change	↓
<b>Ki</b>	↓	↑	↑	Great reduce
<b>Kd</b>	Small Change	↓	↓	Small Change

Ρυθμίζουμε τα κέρδη  $K_p$  ,  $K_i$  ,  $K_d$  μέχρι να πετύχουμε το επιθυμητό αποτέλεσμα

[ P , I , PI , PD ] controllers

# Μέθοδος Ziegler & Nichols

- Set  $K_d$  ,  $K_i$  to 0.
- Increase  $K_p$  till oscillation starts



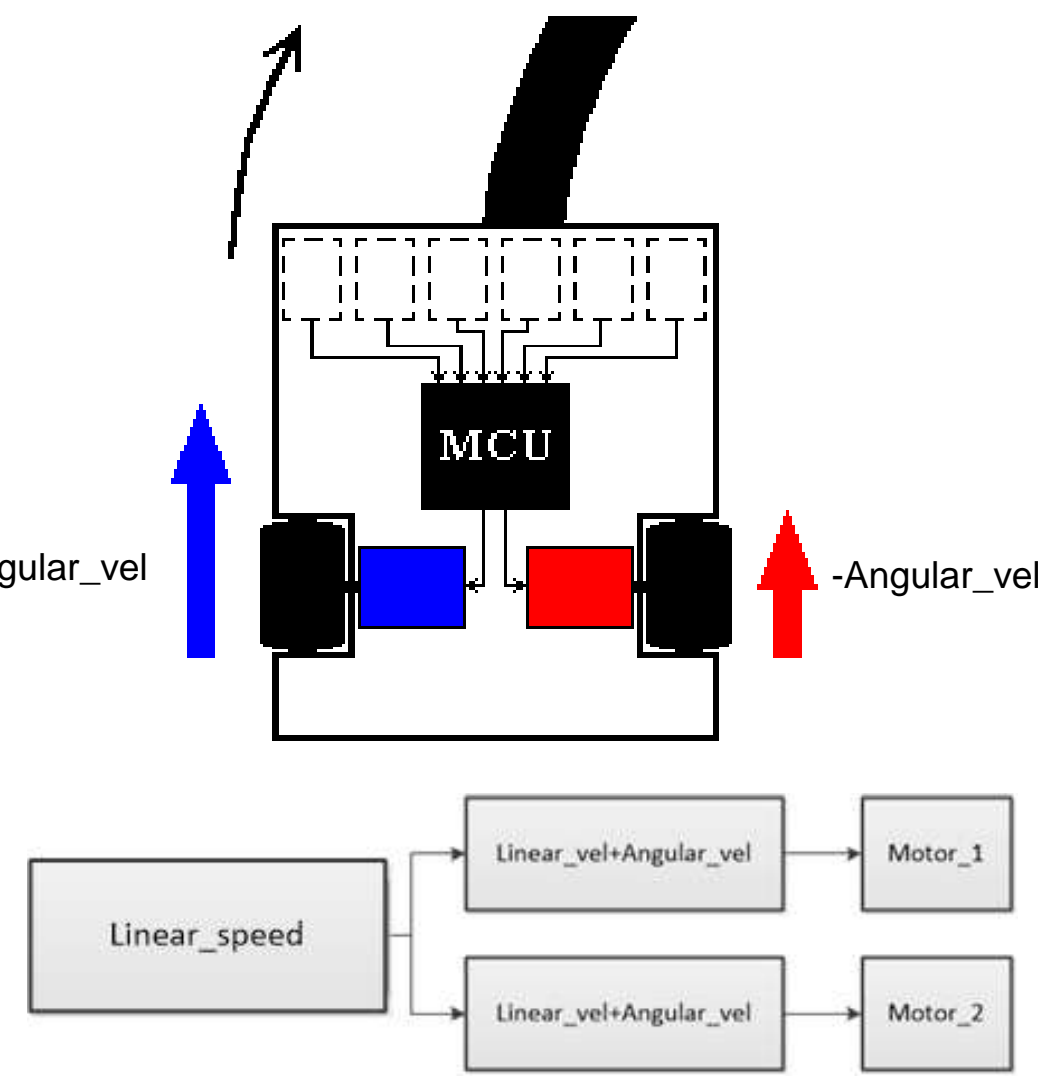
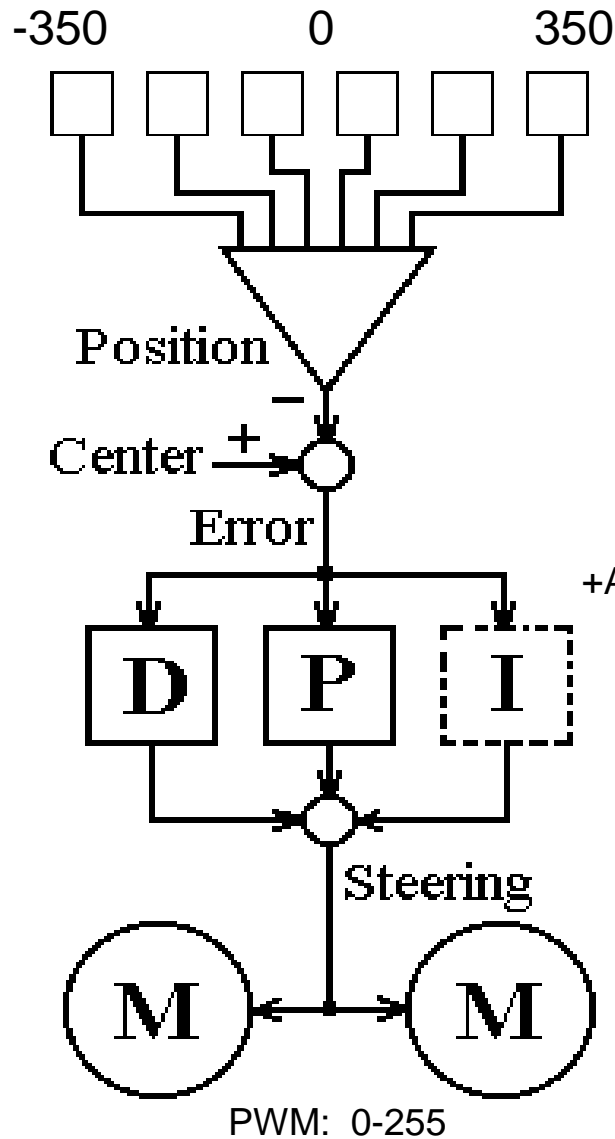
Parameter Increase	Rise time	Overshoot	Settling Time	Steady-state error
<b>Kp</b>	↓	↑	Small Change	↓
<b>Ki</b>	↓	↑	↑	Great reduce
<b>Kd</b>	Small Change	↓	↓	Small Change

# Ψευδοκώδικας

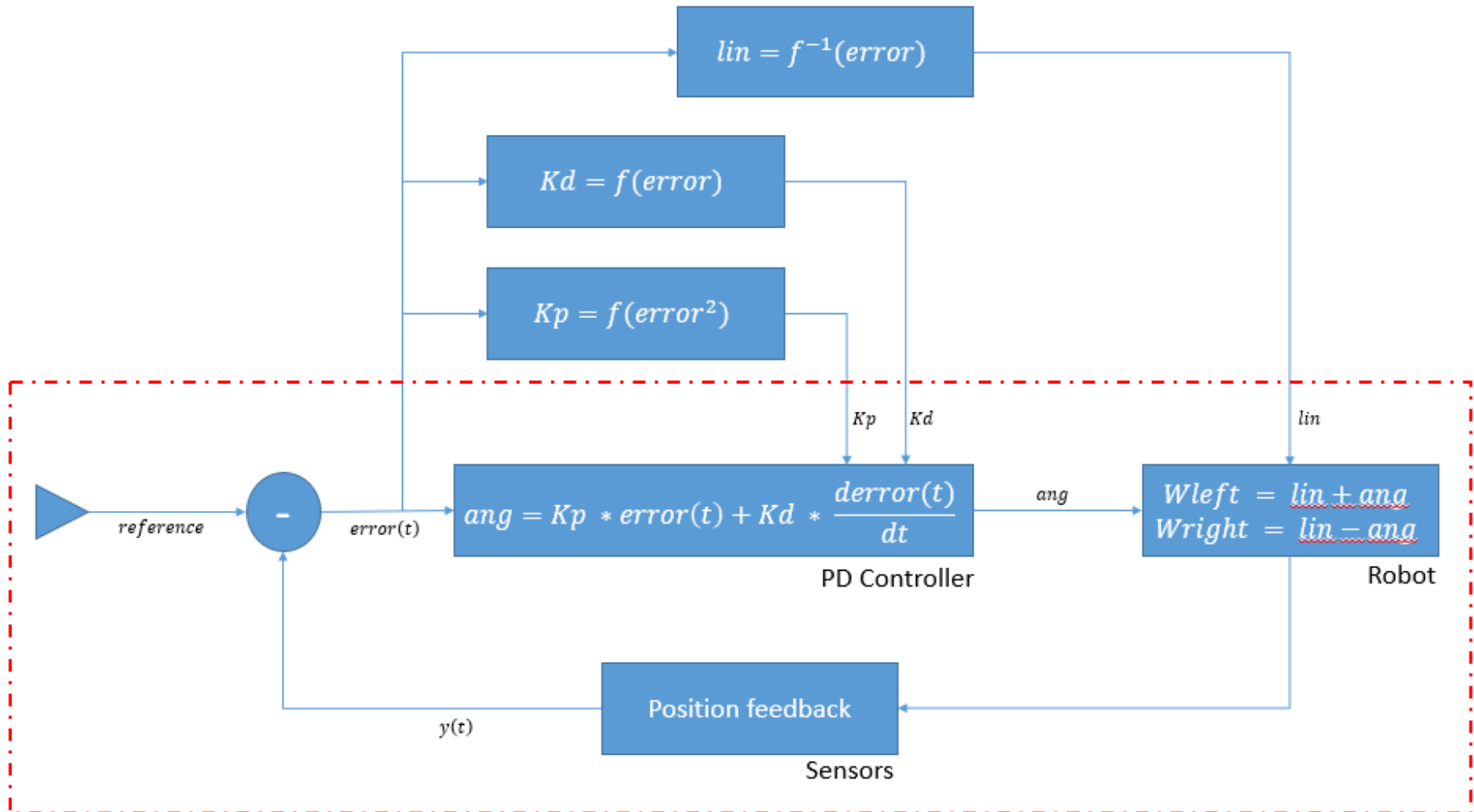
---

```
previous_error = 0
integral = 0
Repeat {
    error = setpoint - measured_value
    integral = integral + error*dt
    derivative = (error - previous_error)/dt
    output = Kp*error + Ki*integral + Kd*derivative
    previous_error = error
    wait(dt)
}
```

# PD Δρομεία



# PD Δρομέα



# Ερωτήσεις ?

My robot says:

Thank you for your attention .

