

Εισαγωγή στην πληροφορική και τις εφαρμογές της

ΠΑΝΑΓΙΩΤΗΣ ΠΑΠΑΖΟΓΛΟΥ

Website: microdev.gr

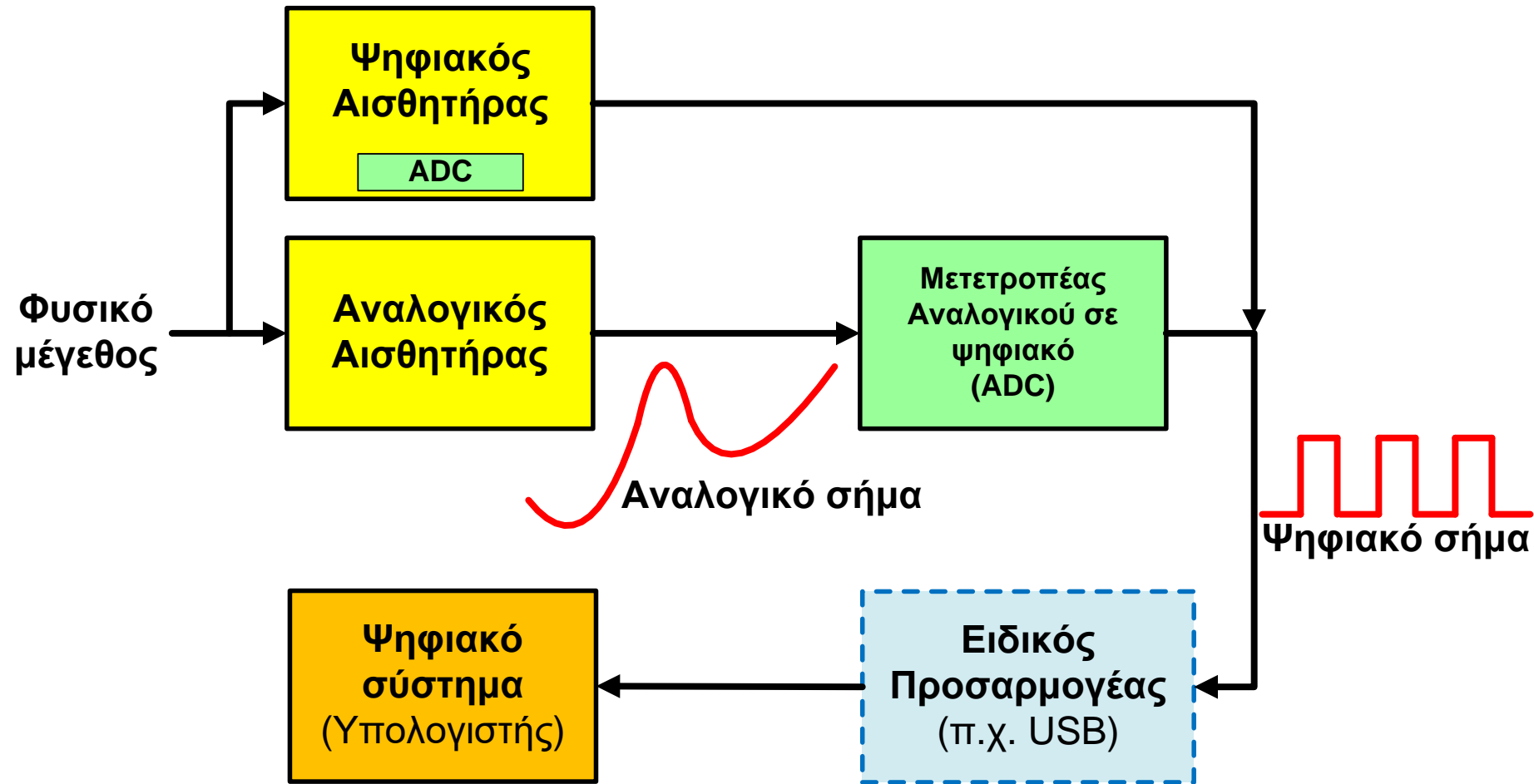


Επιστημονικές Εκδόσεις
ΤΖΙΟΛΑ

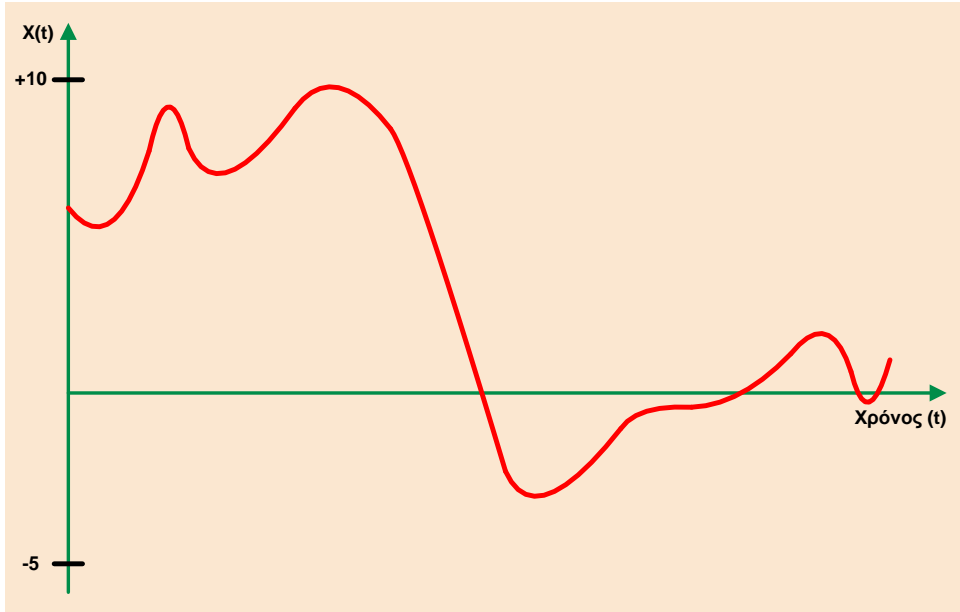
ΚΕΦΑΛΑΙΟ 19
Αλληλεπίδραση
υπολογιστή
με το φυσικό κόσμο



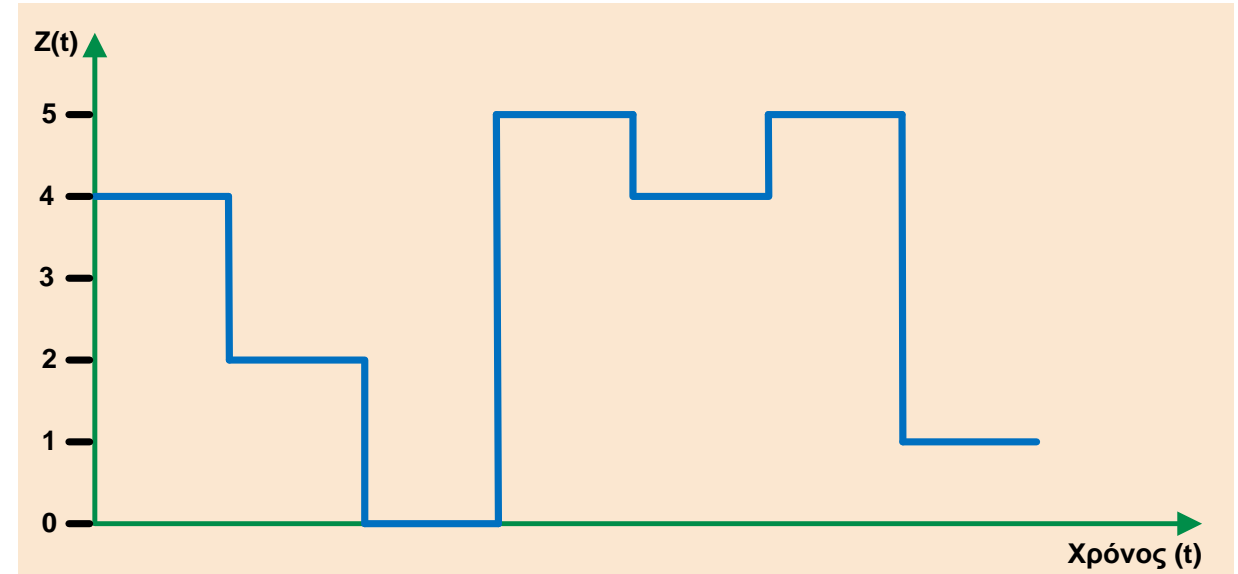
Από το φυσικό μέγεθος στον υπολογιστή



Αναλογικά και ψηφιακά σήματα

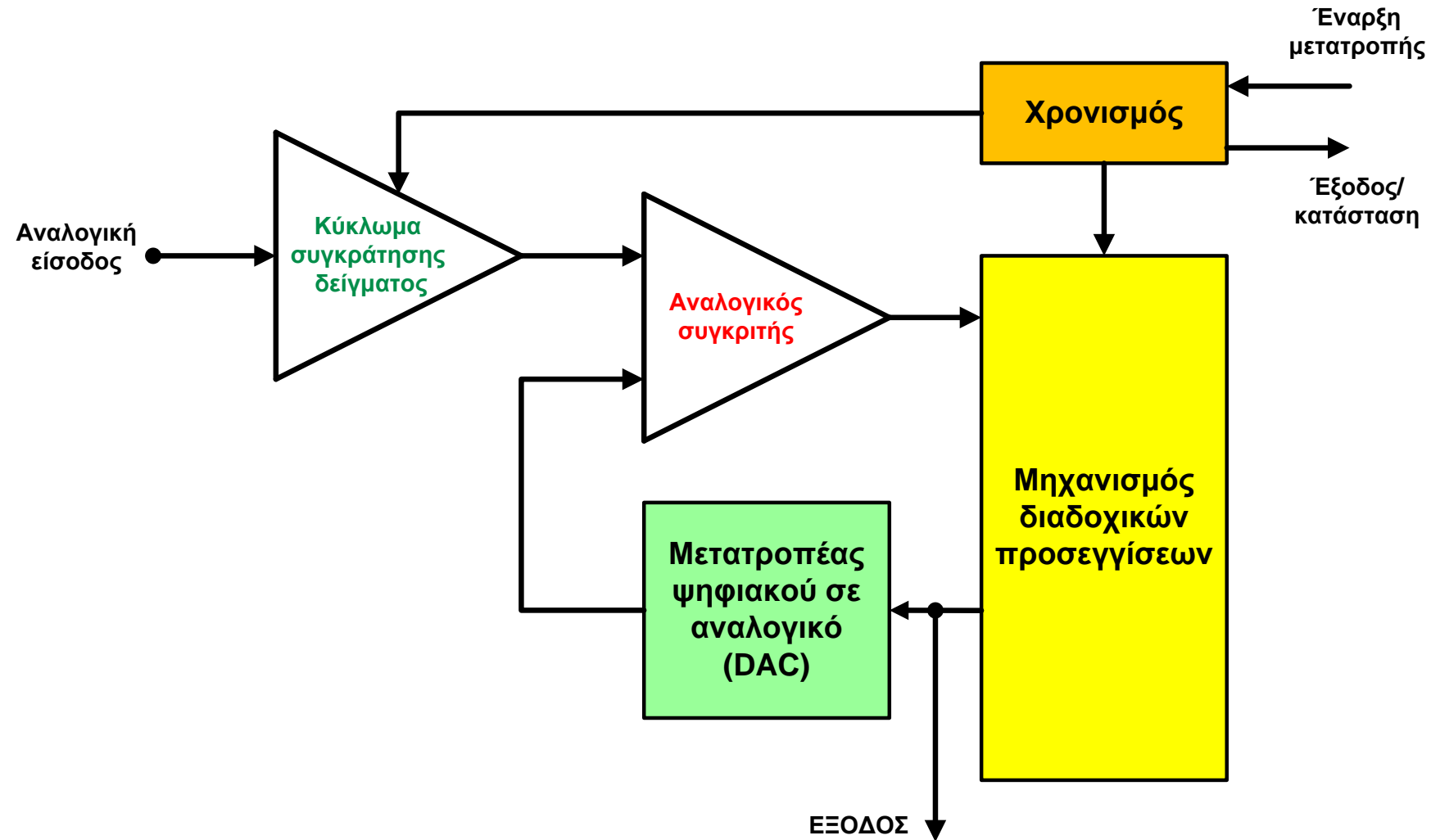


Δείγμα αναλογικού σήματος

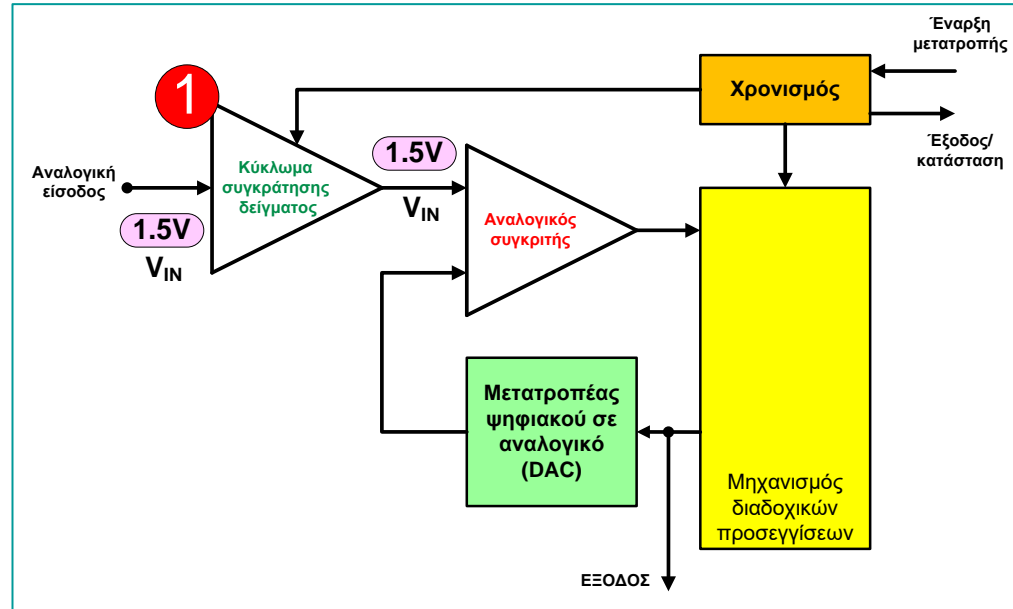


Δείγμα ψηφιακού σήματος

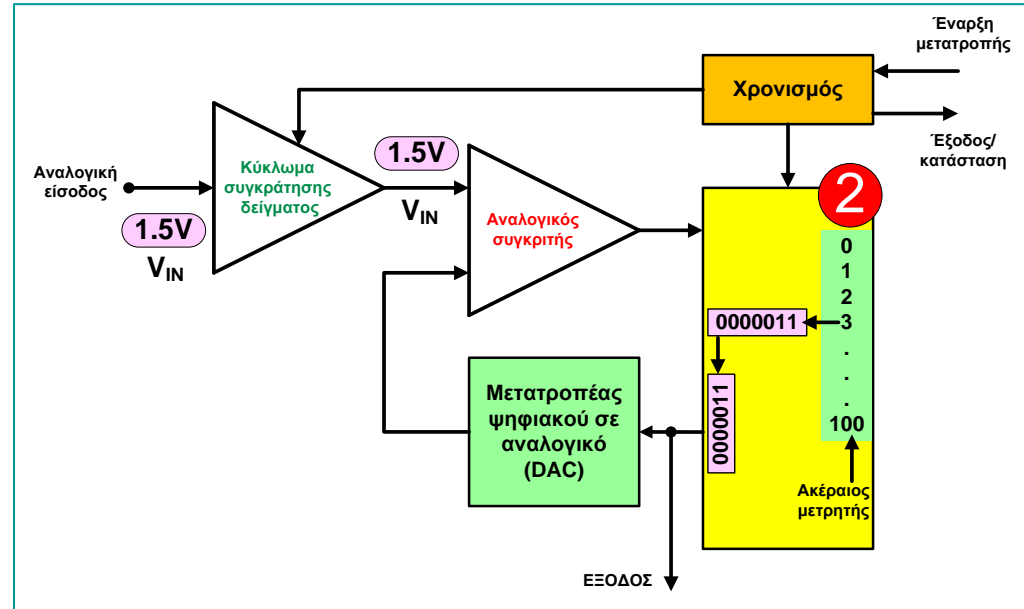
Μετατροπή με διαδοχικές προσεγγίσεις



Μετατροπή με διαδοχικές προσεγγίσεις (1)

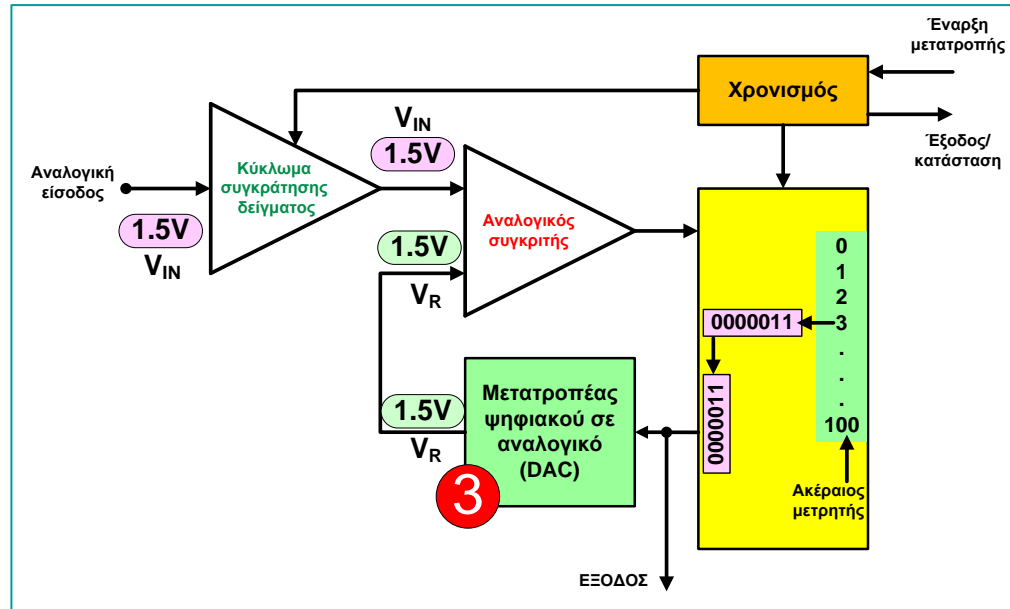


Η άγνωστη τάση εισάγεται για μέτρηση

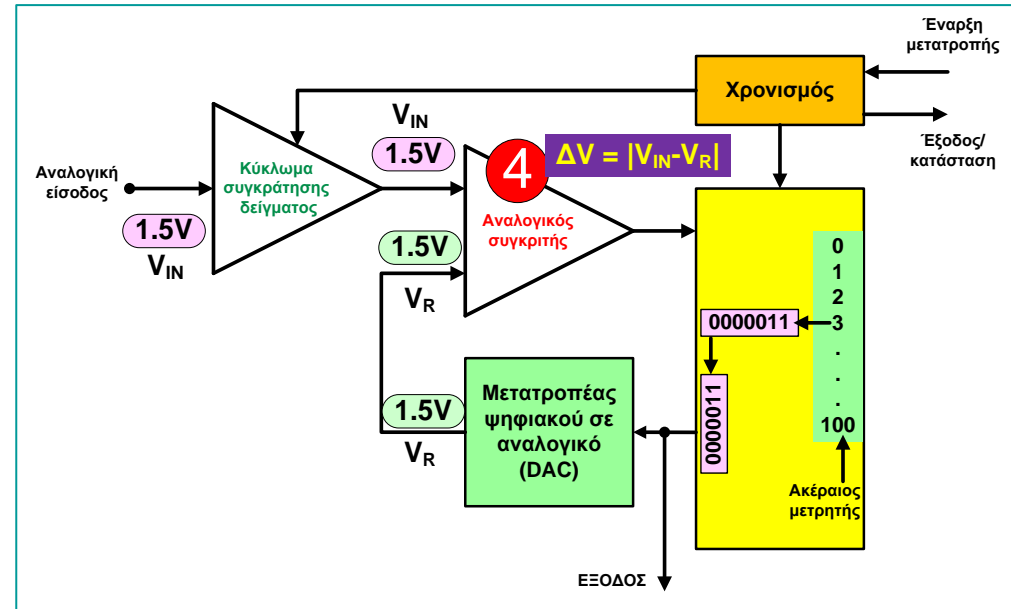


Παραγωγή δυαδικού αριθμού από τον μετρητή

Μετατροπή με διαδοχικές προσεγγίσεις (2)

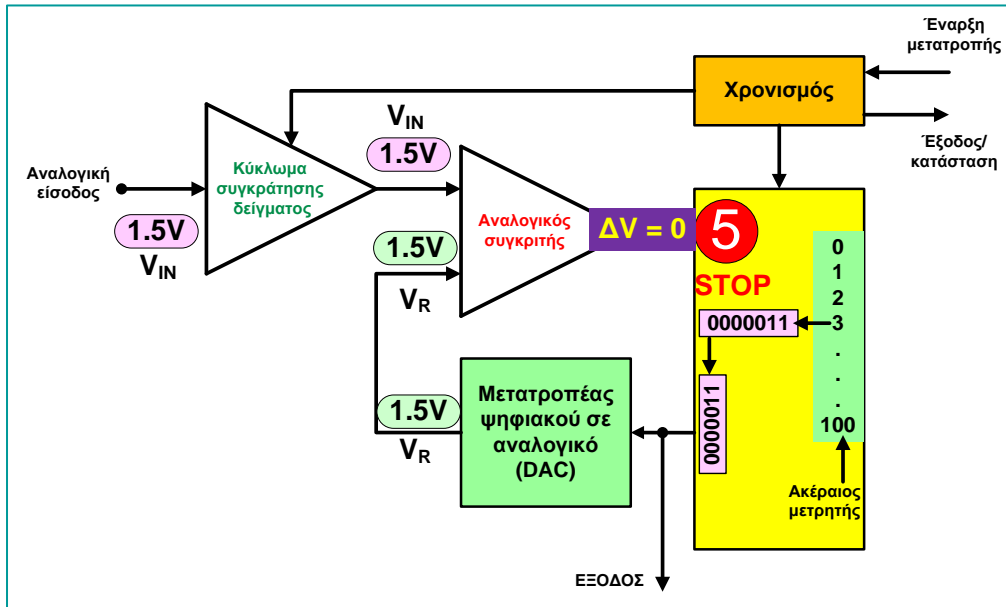


Παραγωγή αναλογικής τάσης V_R

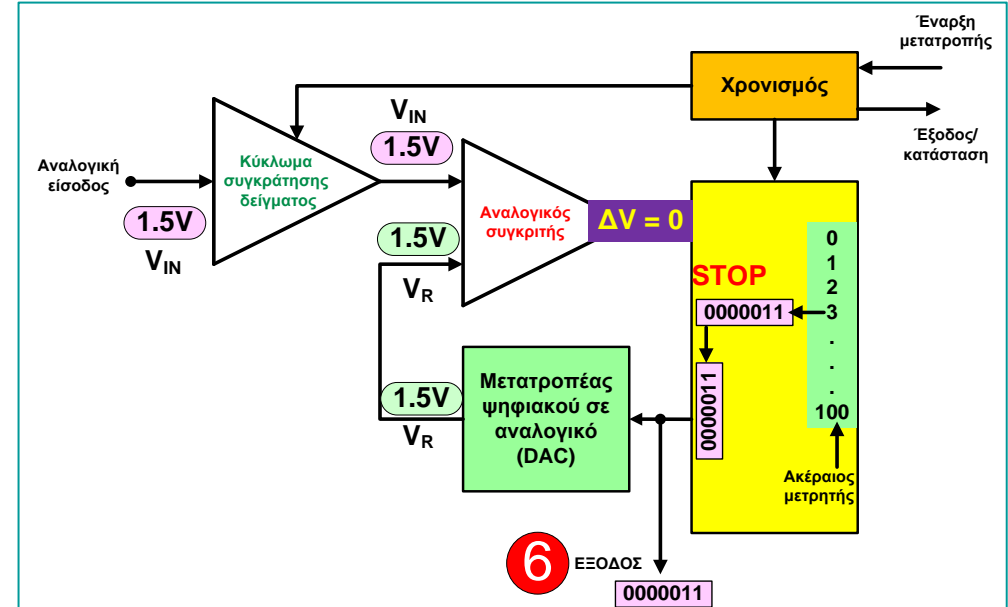


Σύγκριση τάσεων (άγνωστης, δοκιμαστικής)

Μετατροπή με διαδοχικές προσεγγίσεις (3)

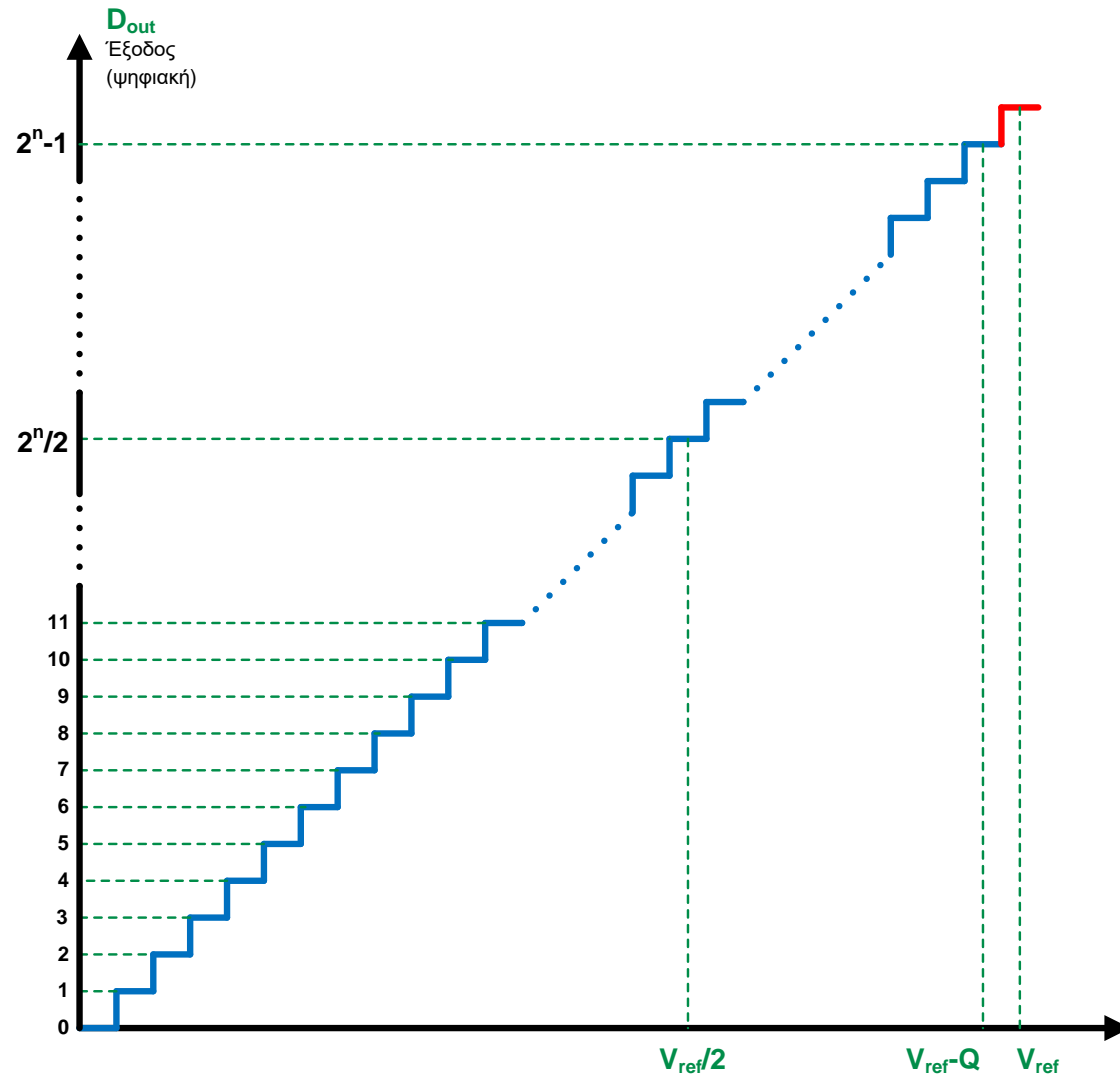


0 μετρητής σταματά



Έξοδος μετατροπέα

Αντιστοίχιση αναλογικών τάσεων σε δυαδική πληροφορία (1)



Ανάλυση μετατροπέα

$$Q = \frac{V_{ref}}{2^n}$$

Ψηφιακή έξοδος μετατροπέα

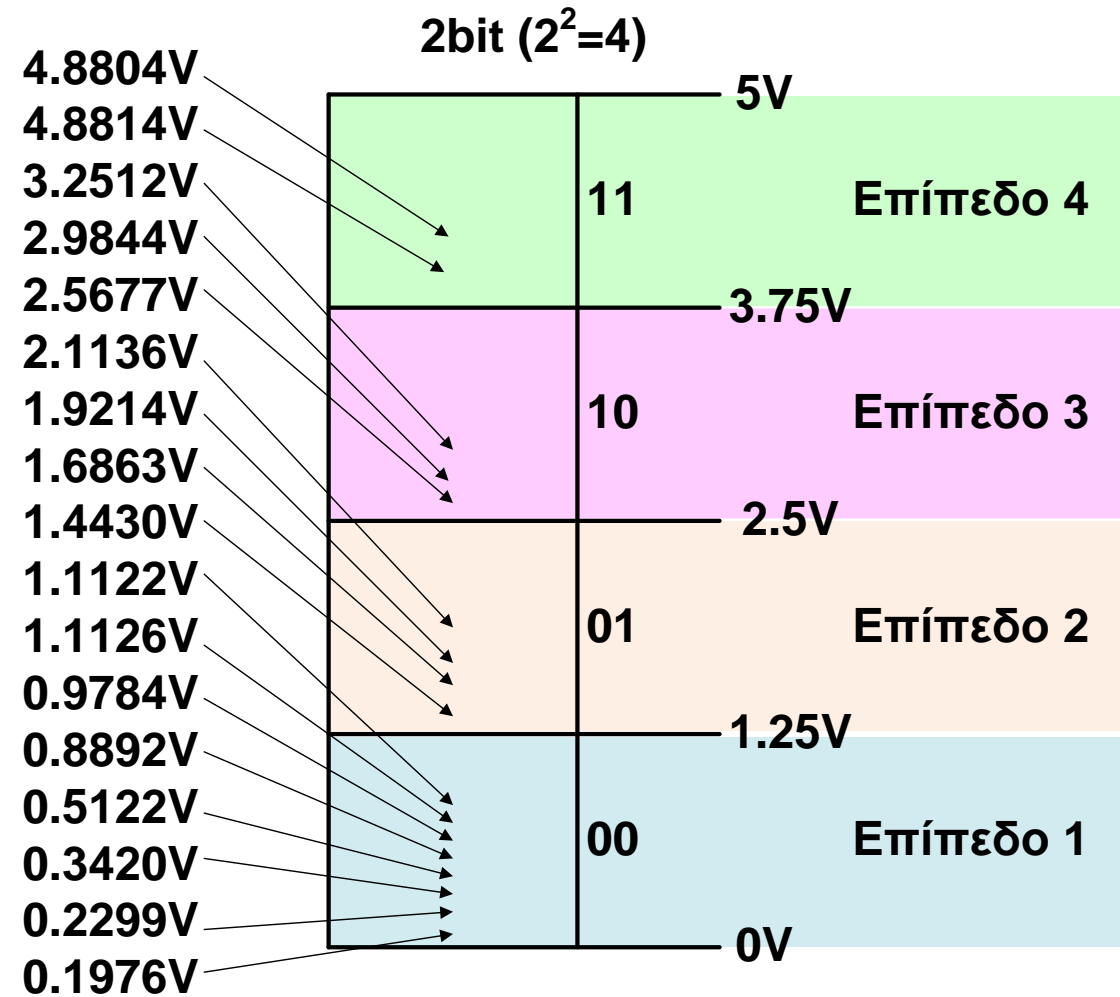
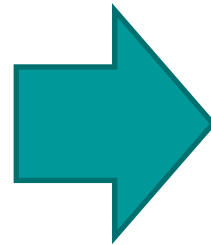
$$D_{out} = \frac{V_{in} \cdot 2^n}{V_{ref}}$$

Αντιστοίχιση αναλογικών τάσεων σε δυαδική πληροφορία (2)

Διαστήματα (στάθμες) για διάφορες τιμές του n και $V_{ref}=5.0V$		
n	Q	Πλήθος διαστημάτων (2^n)
8	0.01953V=19.53mV	$2^8=256$
10	0.00488V=4.88mV	$2^{10}=1024$
12	0.00122V=1.22mV	$2^{12}=4096$
14	0.00031V=0.31mV	$2^{14}=16384$
16	0.00008V=0.08mV	$2^{16}=65536$

Εφαρμογή: Αποτύπωση σε στάθμες

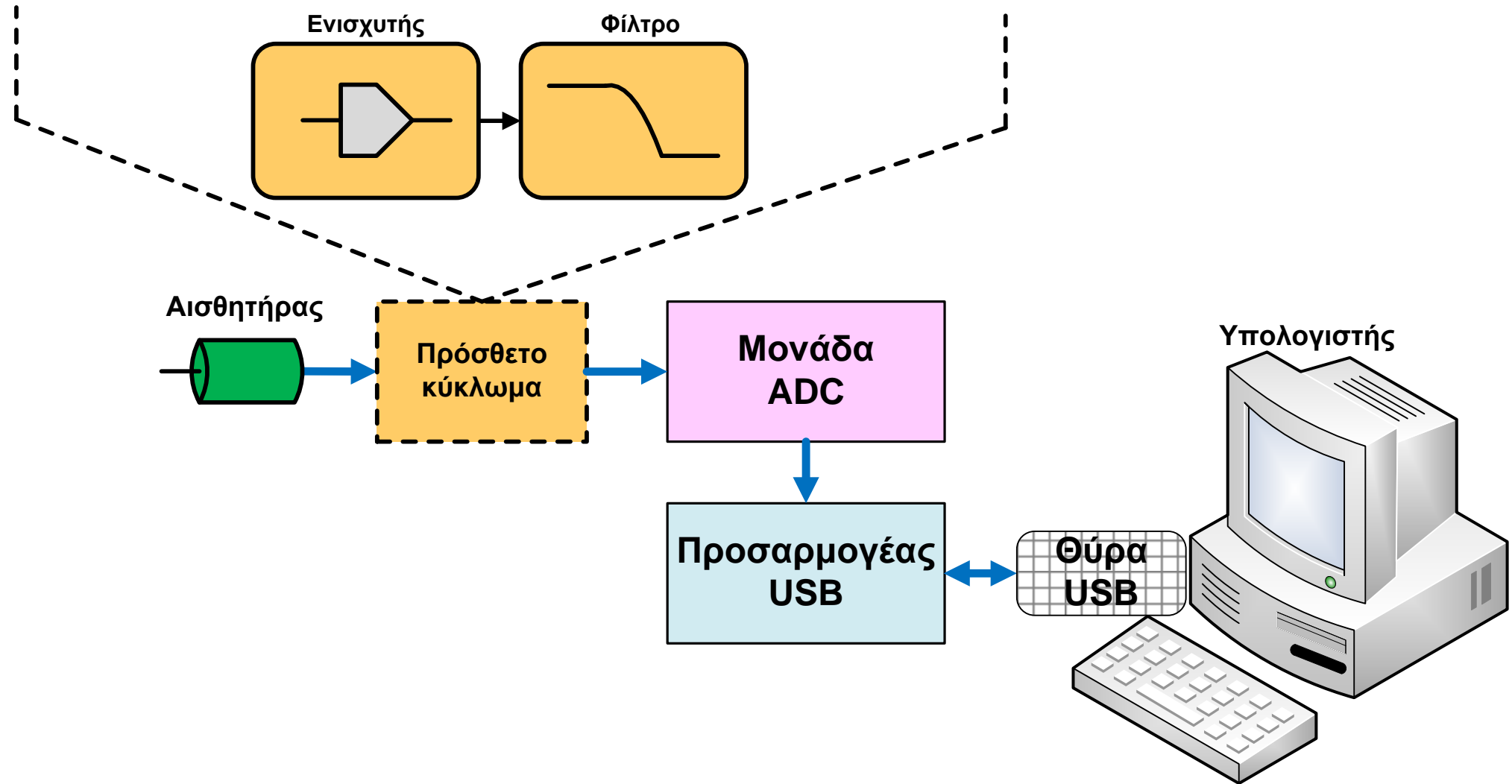
Αναλογική τάση (υποθετική τιμή)
2.8804V
2.8814V
2.2512V
2.9844V
2.5677V
2.1136V
1.9214V
1.6863V
1.4430V
1.1122V
1.1126V
0.9784V
0.8892V
0.5122V
0.3420V
0.2299V
0.1976V



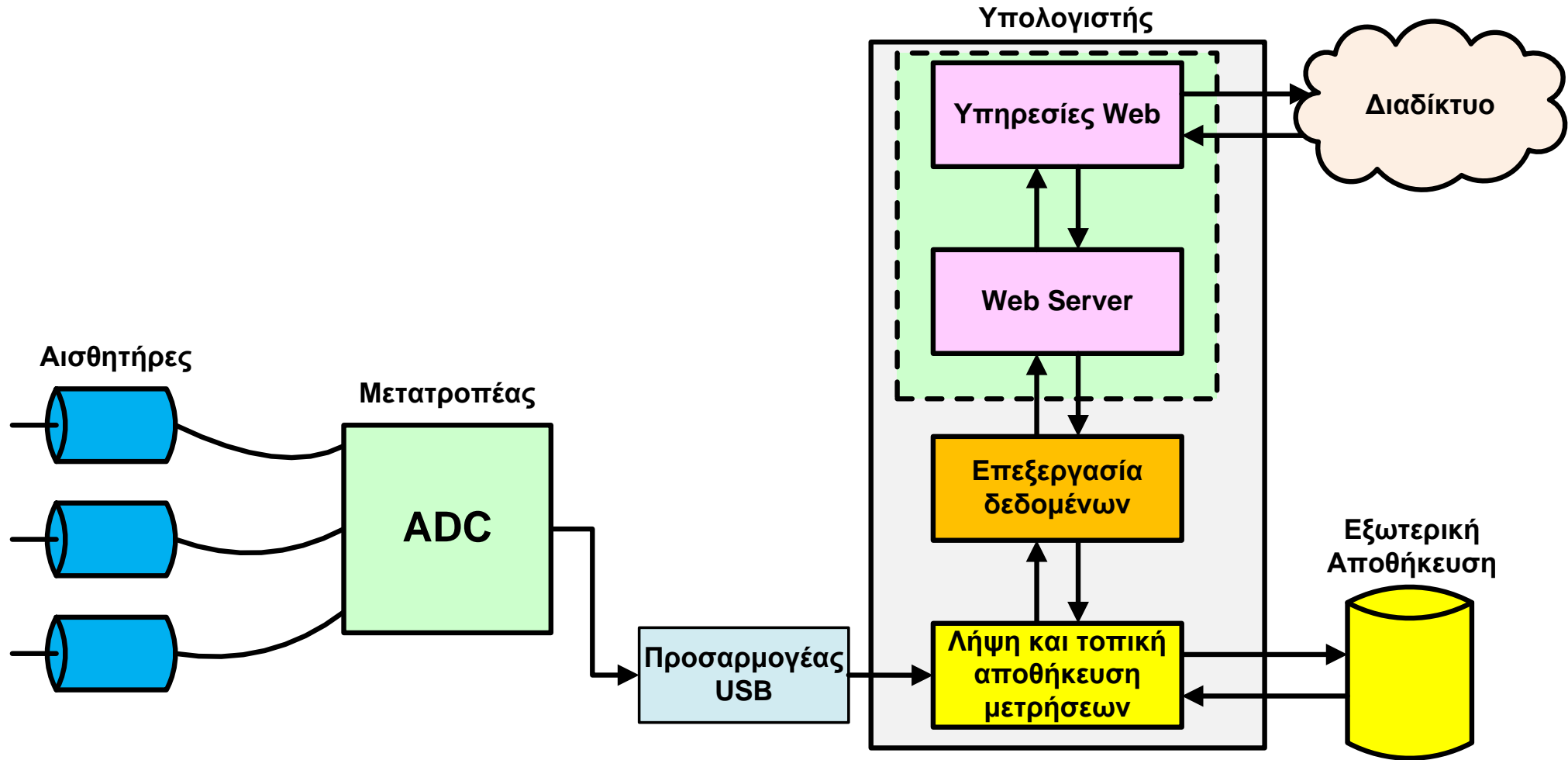
Διακριτική ικανότητα μετατροπεία
Δείτε την εφαρμογή από το βιβλίο



Ανάγνωση αναλογικών σημάτων από τον υπολογιστή

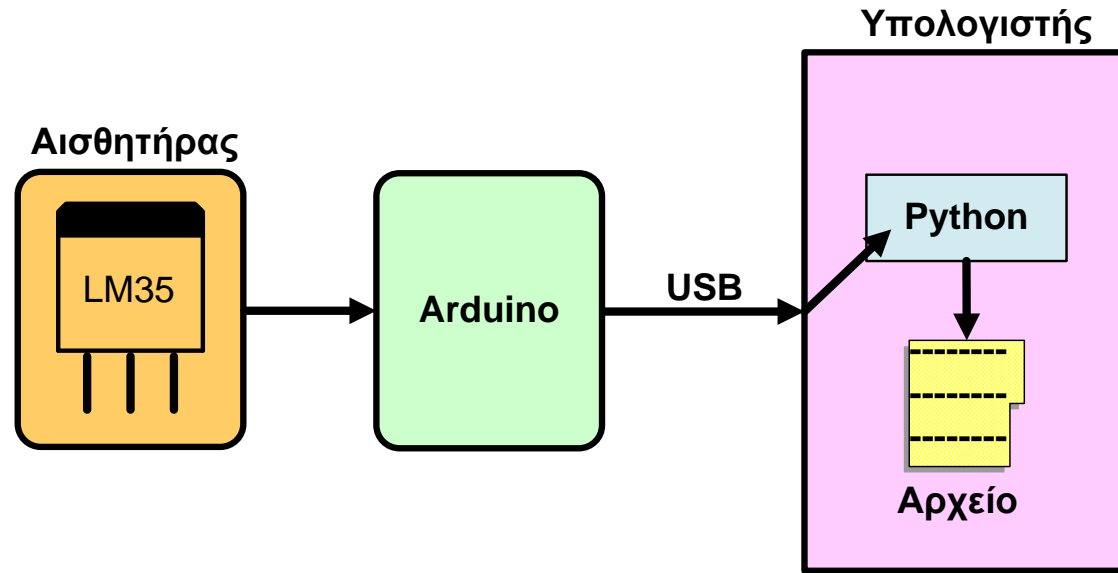


Αξιοποίηση μετρήσεων με το λογισμικό



Αποθήκευση μετρήσεων στον υπολογιστή (1)

Arduino



```
float temp,Q=1.074;

void setup()
{
  analogReference(INTERNAL);
  Serial.begin(9600);
}

void loop()
{
  for(int i=0;i<16;i++)
  {

    temp=(analogRead(0)*Q)/10.0;
    Serial.println(temp,4);
  }
  delay(10000);
}
```


Αποθήκευση μετρήσεων στον υπολογιστή (2)

```
import serial
import csv
from time import gmtime, strftime
ser = serial.Serial('/dev/ttyUSB0')
n=5
i=0
math=[0]*n
while (i<n):
    ser_bytes = str(ser.readline())
    tdate=strftime("%d-%m-%Y",gmtime())
    ttime=strftime("%H:%M:%S",gmtime())
    temp=ser_bytes[11:18]
    wline=str(i)+", "+temp+", "+tdate+", "+ttime
    print(wline)
    math[i]=wline
    i=i+1

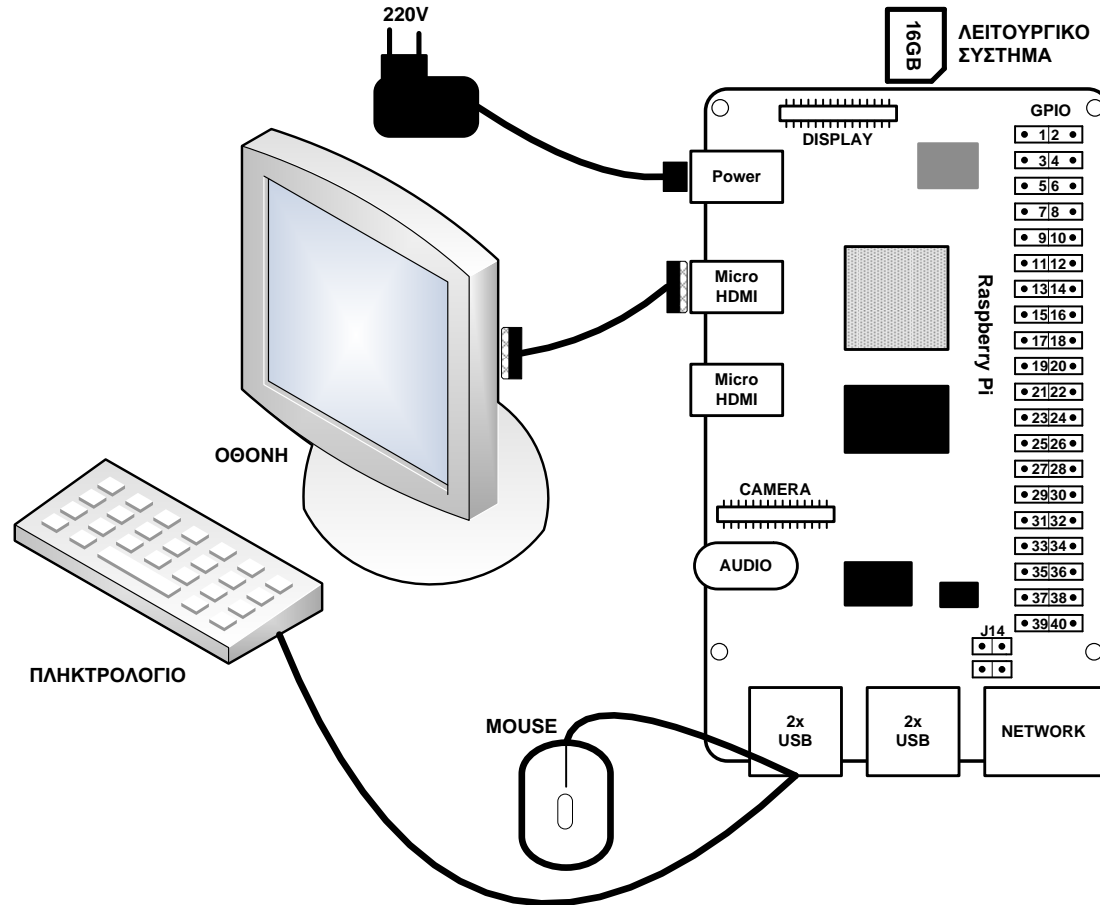
csvfile=open('csvdata.csv','w',newline='')
f=csv.writer(csvfile,delimiter="\t",quoting=csv.QUOTE_NONE)

for row in math:
    f.writerow([row])
csvfile.close()
```

Python



Ο υπολογιστής Raspberry Pi (1)

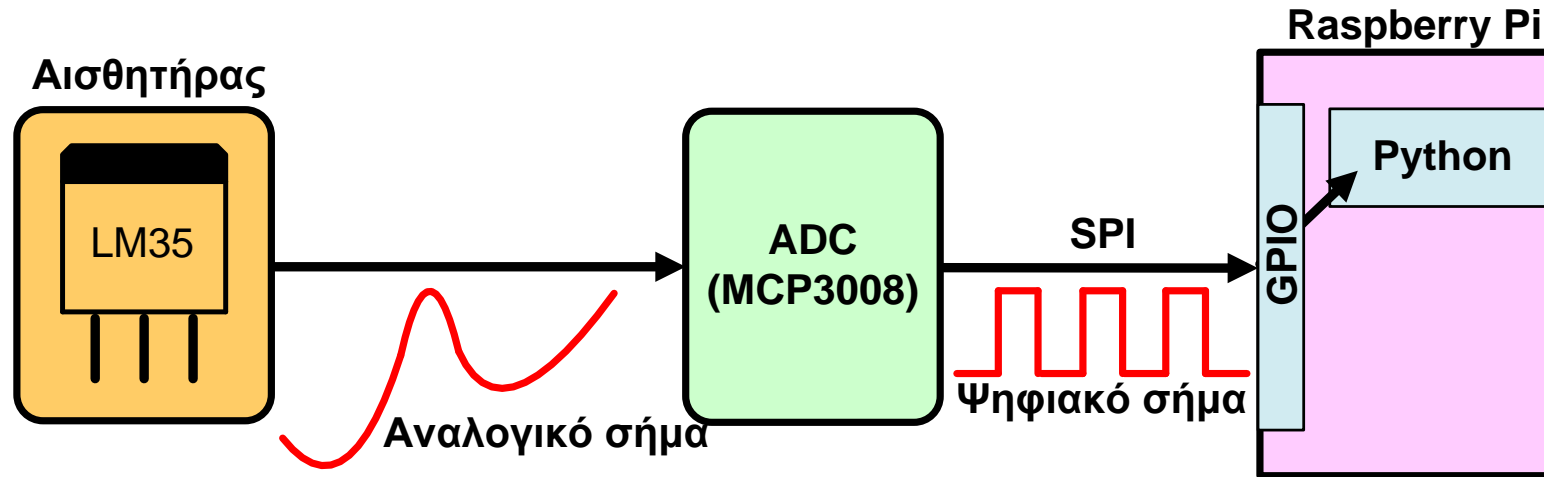


GPIO	
DC Power 3.3V	• 1 2 •
GPIO02 (SDA1, I ² C)	• 3 4 •
GPIO03 (SCL1, I ² C)	• 5 6 •
GPIO04 (GPCLK0)	• 7 8 •
Ground (γείωση)	• 9 10 •
GPIO17	• 11 12 •
GPIO27	• 13 14 •
GPIO22	• 15 16 •
DC Power 3.3V	• 17 18 •
GPIO10 (SPI0_MOSI)	• 19 20 •
GPIO09 (SPI0_MISO)	• 21 22 •
GPIO11 (SPI0_CLK)	• 23 24 •
Ground (γείωση)	• 25 26 •
GPIO00 (SDA0, I ² C)	• 27 28 •
GPIO05	• 29 30 •
GPIO06	• 31 32 •
GPIO13 (PWM1)	• 33 34 •
GPIO19	• 35 36 •
GPIO26	• 37 38 •
Ground (γείωση)	• 39 40 •
DC Power 5V	• 1 2 •
DC Power 5V	• 3 4 •
Ground (γείωση)	• 5 6 •
GPIO14 (TXD0, UART)	• 7 8 •
GPIO15 (RXD0, UART)	• 9 10 •
GPIO18 (PWM0)	• 11 12 •
Ground (γείωση)	• 13 14 •
GPIO23	• 15 16 •
GPIO24	• 17 18 •
Ground (γείωση)	• 19 20 •
GPIO25	• 21 22 •
GPIO08 (SPI0_CEO_N)	• 23 24 •
GPIO07 (SPI0_CE1_N)	• 25 26 •
GPIO01 (SCL0, I ² C)	• 27 28 •
Ground (γείωση)	• 29 30 •
GPIO12 (PWM0)	• 31 32 •
Ground (γείωση)	• 33 34 •
GPIO16	• 35 36 •
GPIO20	• 37 38 •
GPIO21	• 39 40 •



Ο υπολογιστής Raspberry Pi (2)

Μέτρηση («Ανάγνωση») θερμοκρασίας



CH0	1	16	V _{DD}
CH1	2	15	V _{REF}
CH2	3	14	AGND
CH3	4	13	CLK
CH4	5	12	D _{OUT}
CH5	6	11	D _{IN}
CH6	7	10	CS/SHDN
CH7	8	9	DGND

MCP3008

Ο υπολογιστής Raspberry Pi (4)

Μέτρηση («Ανάγνωση») θερμοκρασίας

```
import spidev
from time import sleep
spi = spidev.SpiDev()
spi.open(0,0)

def analogInput(channel):
    spi.max_speed_hz = 1250000
    adc = spi.xfer2([1, (8+channel)<<4,0])
    data = ((adc[1]&3) << 8) + adc[2]
    return data

while True:
    temp_output = analogInput(0)
    temp_volts = temp_output*(3.3/1024)
    temp = temp_volts/0.01
    print("Input:",temp_output, "V:", temp_volts, "T:",temp)
    sleep(5)
```



Δείτε τις εφαρμογές και τις ασκήσεις από το βιβλίο

