Universitatea Tehnică din Cluj-Napoca
Departamentul Calculatoare

Proiectarea cu Micro-Procesoare

Lector: Mihai Negru

An 3 – Calculatoare și Tehnologia Informației

Seria B

Curs 8: Procesarea Semnalelor Analogice

http://users.utcluj.ro/~negrum/

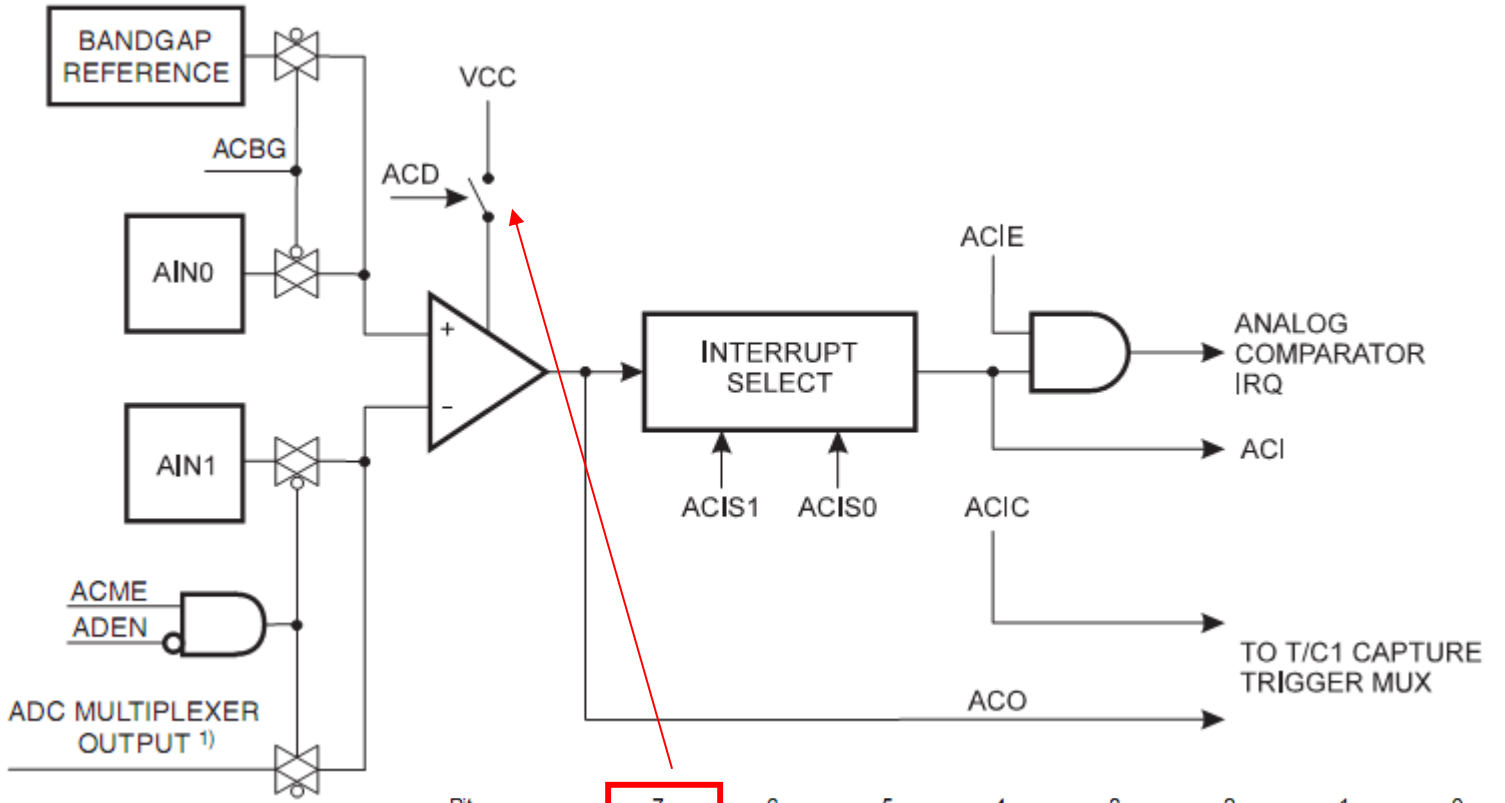
Resurse ATMega pentru semnal analogic

- **Comparator analogic:** compară un semnal de intrare cu alt semnal de intrare, sau cu tensiuni de referință, indicând printr-un bit relația dintre aceste tensiuni (care este mai mare), sau producând cereri de întrerupere

- **Convertor analog/digital pe 10 biți**, cu 16 intrări analogice multiplexate: convertește semnalul analogic într-o valoare numerică intre 0 și 1023.

Comparator Analogic

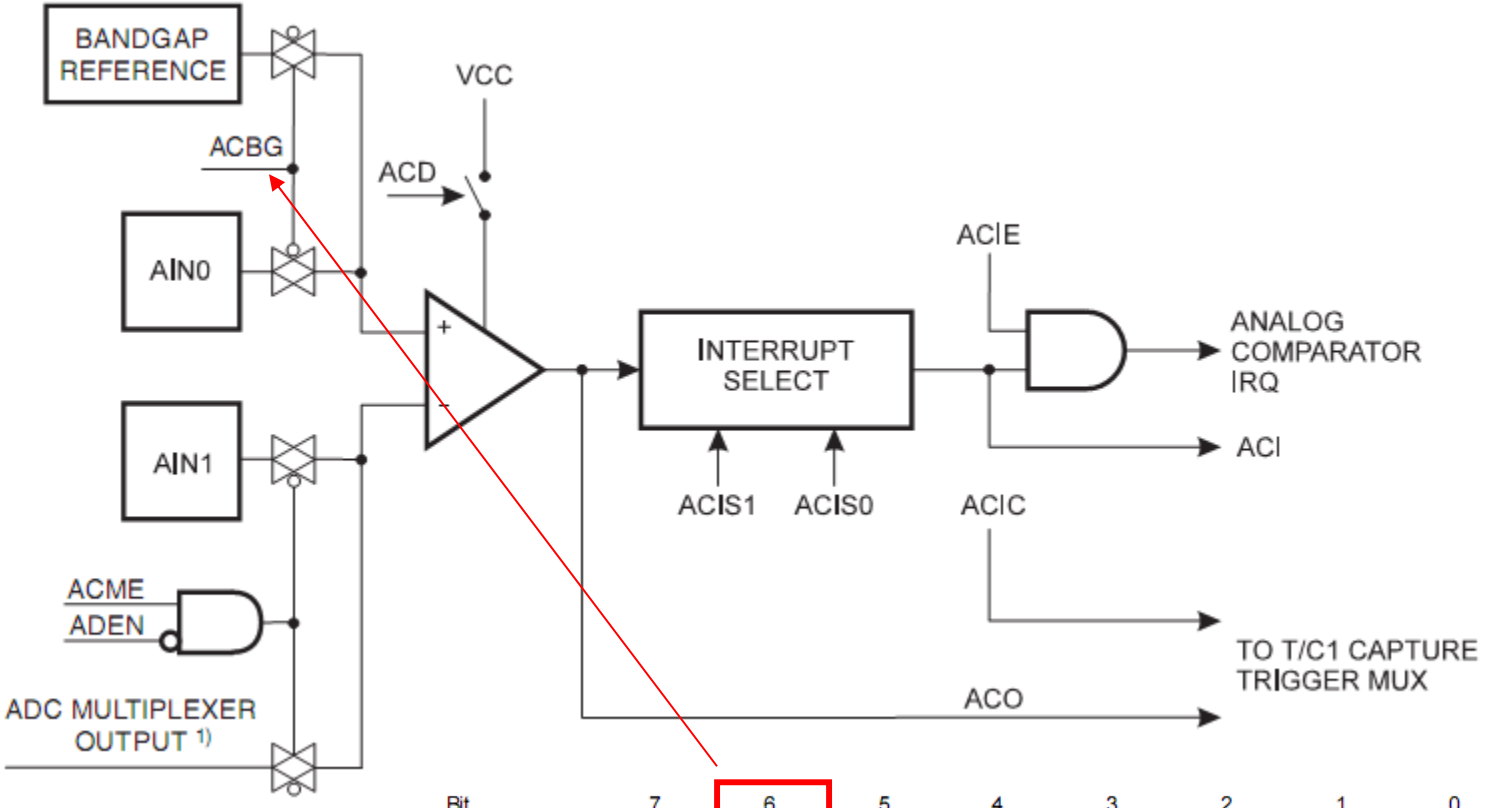
- Comparație între două semnale analogice, AIN0 și AIN1. Daca AIN0 > AIN1, ieșirea ACO este activată (‘1’). AC se activează prin bitul **ACD (Analog Comparator Disable)** din **ACSR**



Bit	7	6	5	4	3	2	1	0	
0x08 (0x28)	ACD	ACBG	ACO	ACI	ACIE	ACIC	ACIS1	ACIS0	ACSR
Read/Write	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	N/A	0	0	0	0	0	

Comparator Analogic

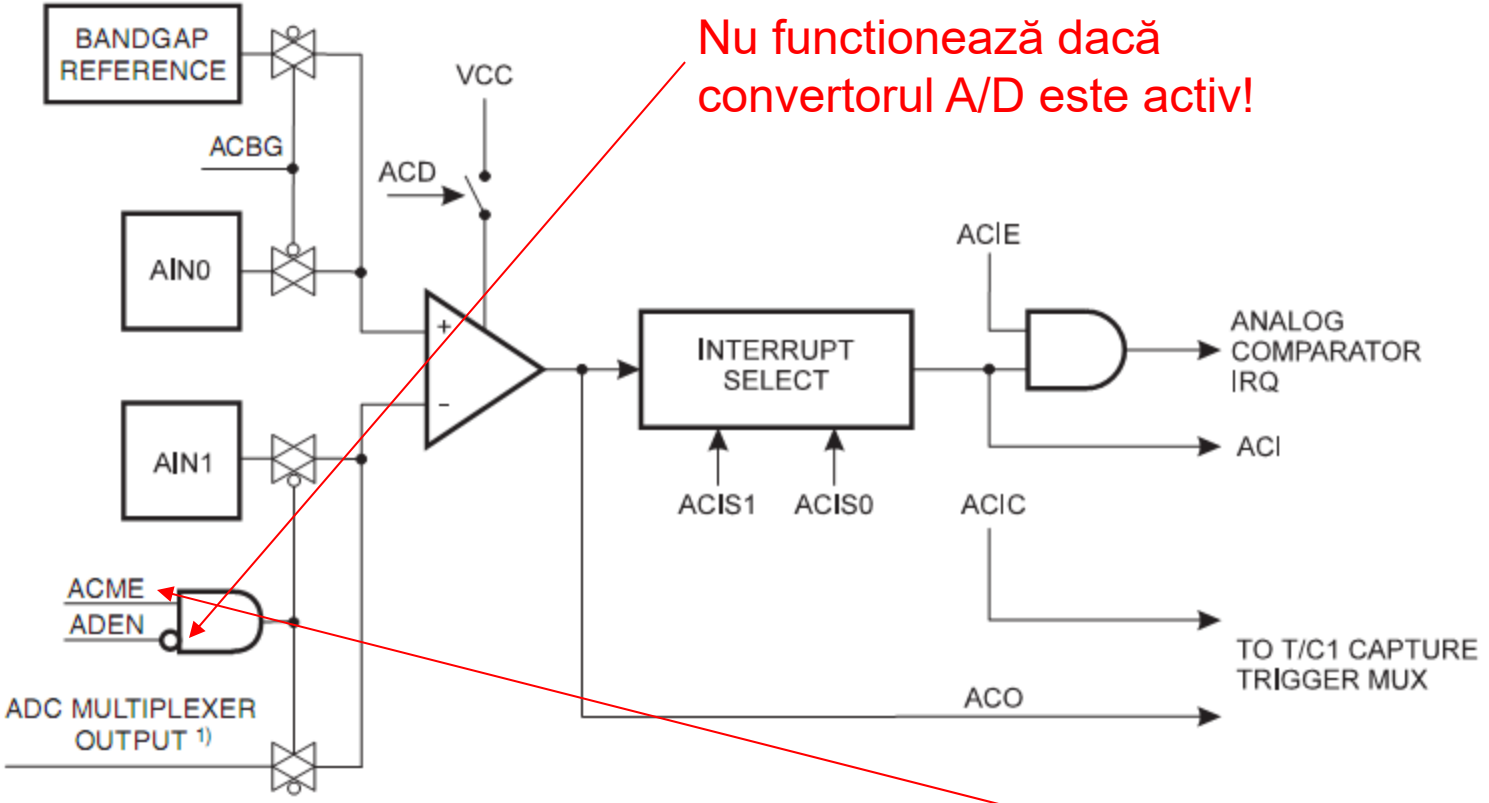
- **Semnalul de referință** AIN0 – poate fi un semnal extern, sau poate fi o valoare internă (ACBG = Analog Comparator BandGap Select, BANDGAP, **1.26V**)



Bit	7	6	5	4	3	2	1	0	
0x08 (0x28)	ACD	ACBG	ACO	ACI	ACIE	ACIC	ACIS1	ACIS0	ACSR
Read/Write	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	N/A	0	0	0	0	0	

Comparator Analogic

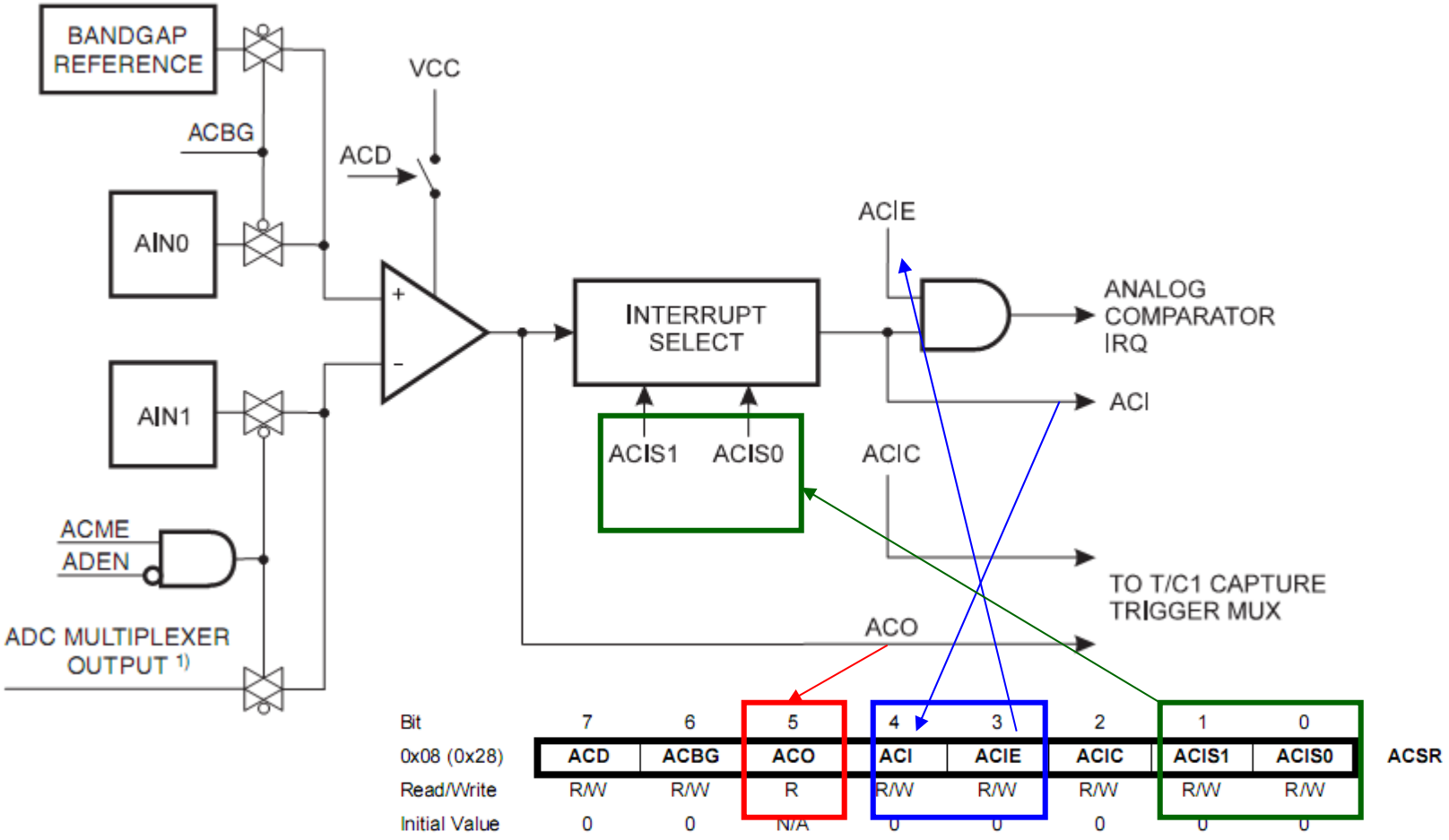
- **Alegerea semnalului de intrare –** Semnalul de comparat poate proveni de la pinul AIN1 sau de la una din cele 8 intrări ale multiplexorului analogic de la convertorul A/D. Selectia din registrul **SFIOR**



Bit	7	6	5	4	3	2	1	0	
0x20 (0x40)	TSM	–	–	–	ACME	PUD	PSR2	PSR10	SFIOR
Read/Write	R/W	R	R	R	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

Comparator Analogic

- **Utilizarea rezultatului comparației (ACO = Analog Comparator Output)**
- Citire directă, declanșare eveniment capture, sau generare cerere intrerupere



Comparator Analogic

- **Configurarea si folosirea intreruperilor**
- ACIE – Analog Comparator Interrupt Enable
- ACIS1, ACIS0 – modul de declansare a intreruperii

ACIS1	ACIS0	Interrupt Mode
0	0	Comparator Interrupt on Output Toggle.
0	1	Reserved
1	0	Comparator Interrupt on Falling Output Edge.
1	1	Comparator Interrupt on Rising Output Edge.

Bit	7	6	5	4	3	2	1	0	
0x08 (0x28)	ACD	ACBG	ACO	ACI	ACIE	ACIC	ACIS1	ACIS0	ACSR
Read/Write	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	N/A	0	0	0	0	0	

Adresa întreruperii AC

0x002E	ANALOG COMP	Analog Comparator

Convertor Analog/Digital

- **Conversia semnalului analogic în semnal digital**
  - Transformarea unui semnal analogic (tensiune) de intrare într-o valoare digitală pe **10 biți**
  - Intrarea poate fi "single end" – tensiunea de intrare se calculeaza între pinul de intrare și GND, sau diferențială – diferența de tensiune între doi pini de intrare
  - Pentru intrarea diferențială, se poate utiliza amplificare (Gain)

$$ADC = \frac{V_{IN} * 1024}{V_{REF}}$$

ADC: 0…1023

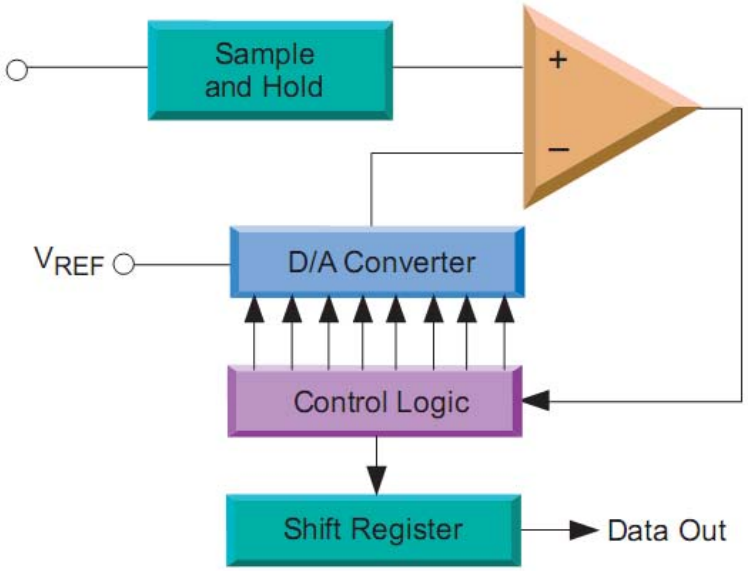
$$ADC = \frac{(V_{POS} - V_{NEG}) \cdot Gain \cdot 512}{V_{REF}}$$

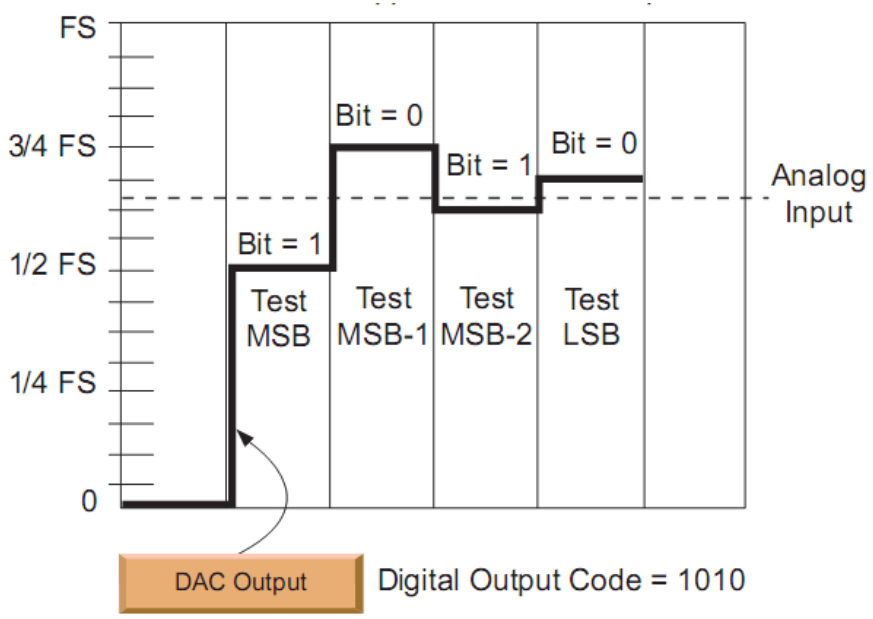
ADC: -512…511, complement față de 2

- V_{REF} poate fi:
  - AVCC – conectat în placă la VCC
  - A_{REF} – tensiune externă de referința
  - Tensiune de referință internă **2.56 V**

Convertor Analog/Digital

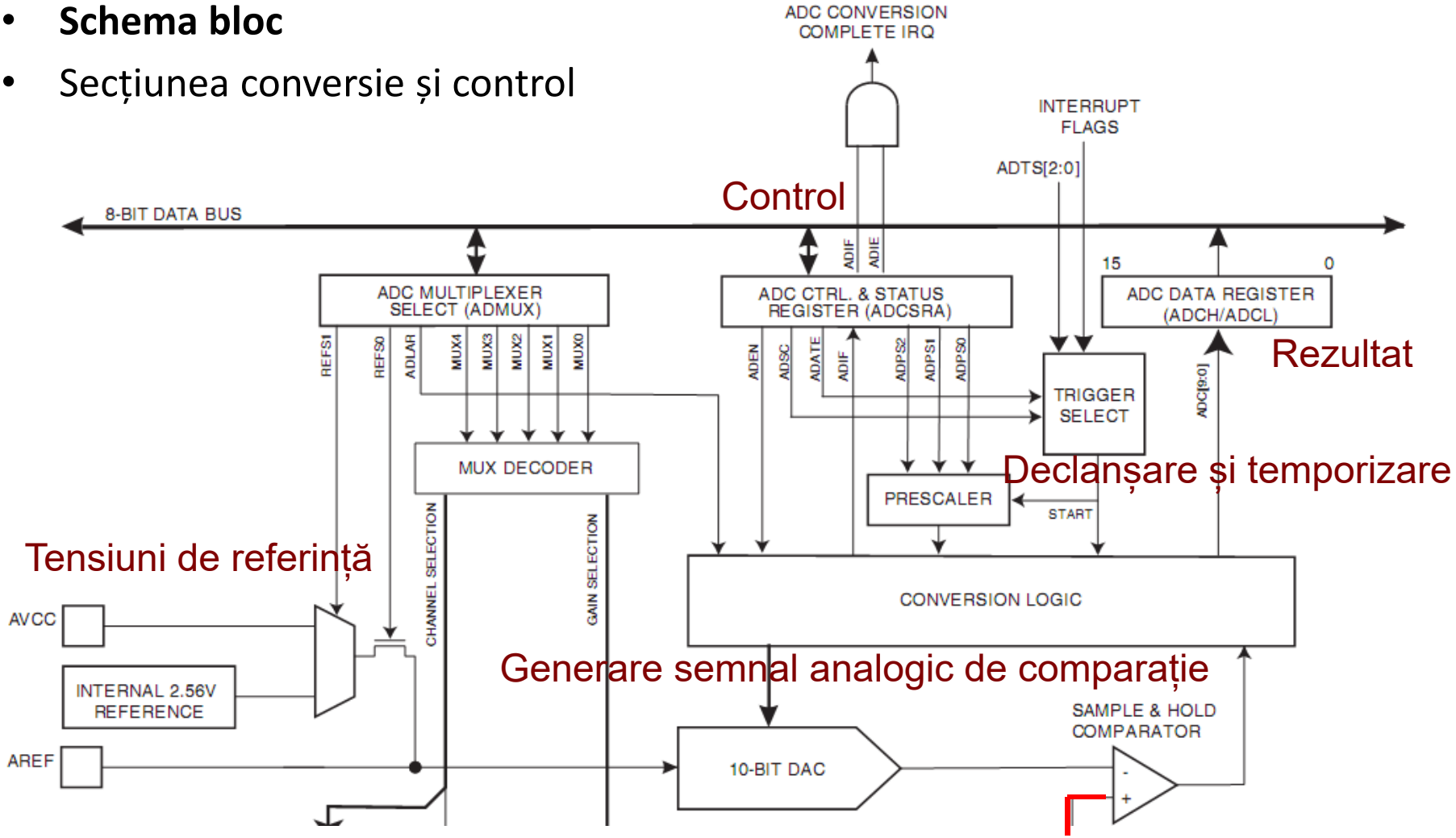
- **Principiul de funcționare**
  - Comparații succesive cu diviziuni ale unei tensiuni de referință
  - Principiul este similar cu căutarea binară





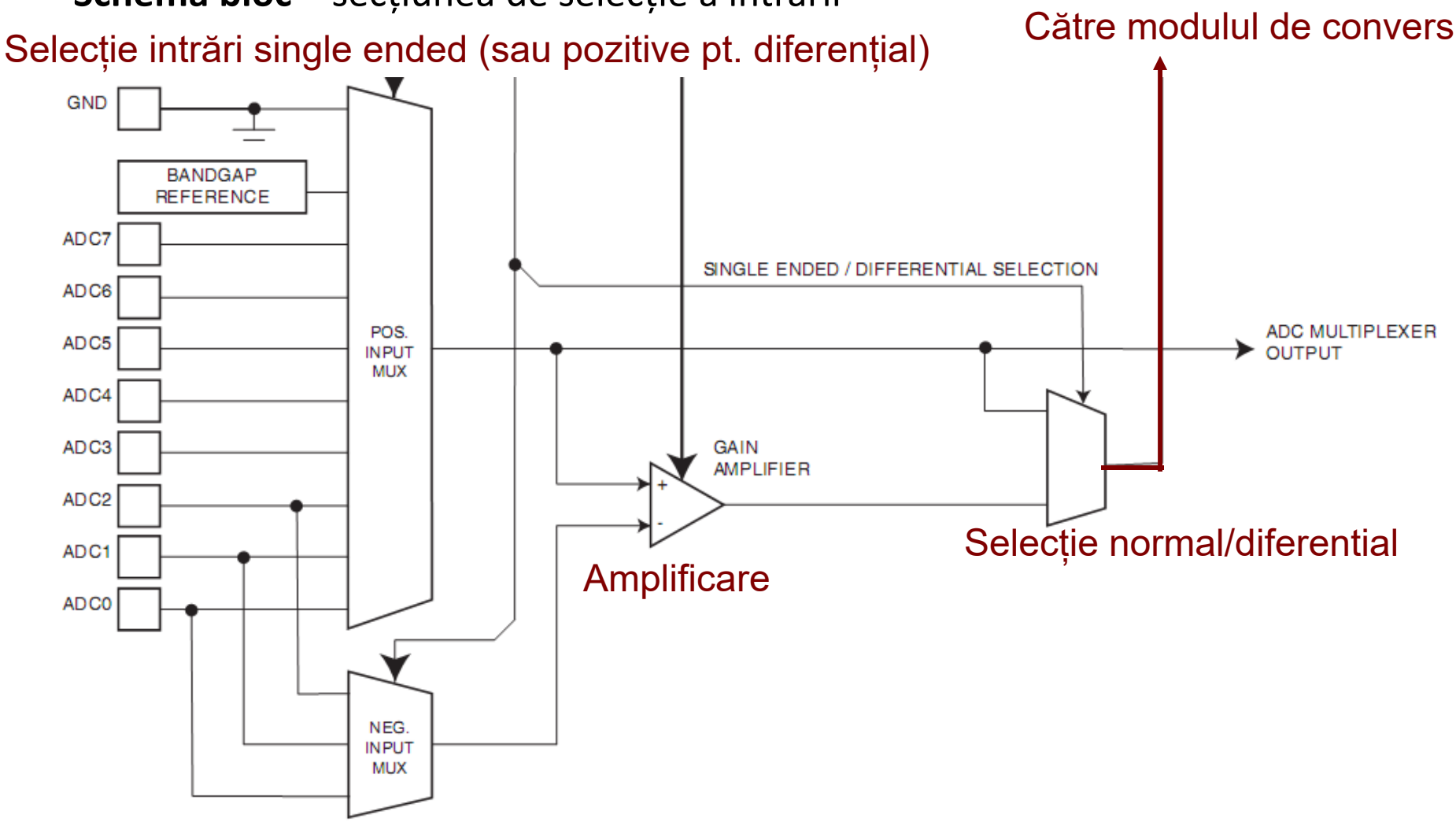
Convertor Analog/Digital

- **Schema bloc**
- Secțiunea conversie și control



Convertor Analog/Digital

- **Schema bloc** – secțiunea de selecție a intrării



Convertor Analog/Digital

- **Configurare convertor**
  - Activare – setarea bitului **ADEN** din registrul **ADCSRA**
  - Folosire întreruperi la terminarea conversiei – setarea bitului **ADIE** din **ADCSRA**
  - Selecție frecvență pentru ceasul convertorului: bitii **ADPS2:0**

7	6	5	4	3	2	1	0	
ADEN	ADSC	ADATE	ADIF	ADIE	ADPS2	ADPS1	ADPS0	ADCSRA
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

ADPS2	ADPS1	ADPS0	Division Factor
0	0	0	2
0	0	1	2
0	1	0	4
0	1	1	8
1	0	0	16
1	0	1	32
1	1	0	64
1	1	1	128

Selecția tensiunilor de referință:

- biții **REFS1:0** din **ADMUX**

REFS1	REFS0	Voltage Reference Selection
0	0	AREF, Internal Vref turned off.
0	1	AVCC with external capacitor at AREF pin.
1	0	Reserved
1	1	Internal 2.56V Voltage Reference with external capacitor at AREF pin.

7	6	5	4	3	2	1	0	
REFS1	REFS0	ADLAR	MUX4	MUX3	MUX2	MUX1	MUX0	ADMUX
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

Convertor Analog/Digital

- **Configurare date de ieșire și accesare rezultate**
- Dispunerea celor 10 biți de date este controlată de bitul **ADLAR** din **ADMUX**

7	6	5	4	3	2	1	0	
REFS1	REFS0	ADLAR	MUX4	MUX3	MUX2	MUX1	MUX0	ADMUX
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

ADLAR = 0 – ajustare la dreapta

15	14	13	12	11	10	9	8	
–	–	–	–	–	–	ADC9	ADC8	ADCH
ADC7	ADC6	ADC5	ADC4	ADC3	ADC2	ADC1	ADC0	ADCL
7	6	5	4	3	2	1	0	

ADLAR = 1 – ajustare la stânga

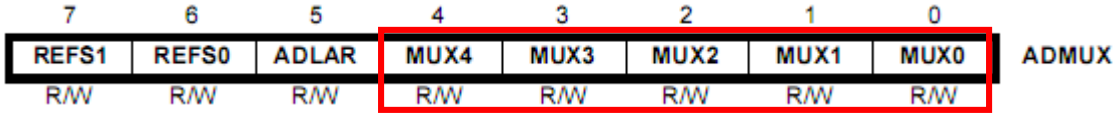
15	14	13	12	11	10	9	8	
ADC9	ADC8	ADC7	ADC6	ADC5	ADC4	ADC3	ADC2	ADCH
ADC1	ADC0	–	–	–	–	–	–	ADCL
7	6	5	4	3	2	1	0	

- Citirea datelor: prima data se citeste **ADCL**, apoi **ADCH**. La citirea ADCL, registrul ADCH ramane cu aceeași valoare până este citit

- Dacă ADLAR = 1, se poate citi doar **ADCH**, ca rezultat pe 8 biți (rezoluție mai scăzută) ADCH = Vin * 256 / Vref

Convertor Analog/Digital

- **Selecția intrărilor**
- Configurarea biților **MUX4:0** din **ADMUX**



Aceste tabele nu conțin toate combinațiile!
Pentru detalii, consultați datasheet-ul corespunzător chipului pe care il programați

MUX4..0	Single Ended Input
00000	ADC0
00001	ADC1
00010	ADC2
00011	ADC3
00100	ADC4
00101	ADC5
00110	ADC6
00111	ADC7

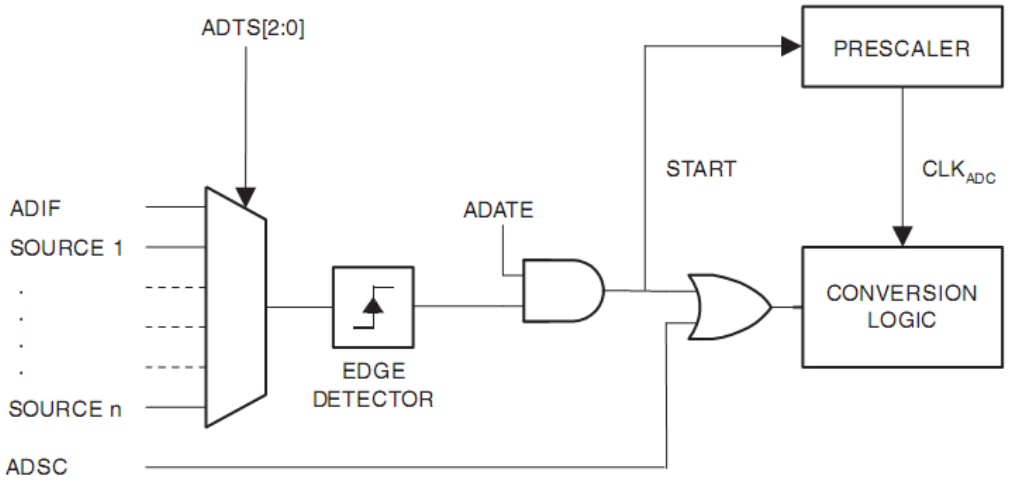
MUX4..0	Positive Differential Input	Negative Differential Input	Gain
01000	ADC0	ADC0	10x
01001	ADC1	ADC0	10x
01010	ADC0	ADC0	200x
01011	ADC1	ADC0	200x
01100	ADC2	ADC2	10x
01101	ADC3	ADC2	10x
01110	ADC2	ADC2	200x
01111	ADC3	ADC2	200x
10000	ADC0	ADC1	1x

Convertor Analog/Digital

- **Declanșare conversie**

7	6	5	4	3	2	1	0	
ADEN	ADSC	ADATE	ADIF	ADIE	ADPS2	ADPS1	ADPS0	ADCSRA
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

  - La cerere – scrierea bitului **ADSC** din **ADCSRA**. Acest bit rămâne ‘1’ în timpul conversiei, și se șterge la terminare
  - Automat, declanșat de variații ale semnalelor de intrare – dacă bitul **ADATE** din ADCSRA e setat. Una din surse este ADIF – flag-ul care semnalează terminarea unei conversii, și eventual cererea unei întreruperi. În acest caz, o nouă conversie incepe când se termină cea anterioară.
  - Sursele pentru declanșare automată → biții **ADTS** din **ADCSRB**

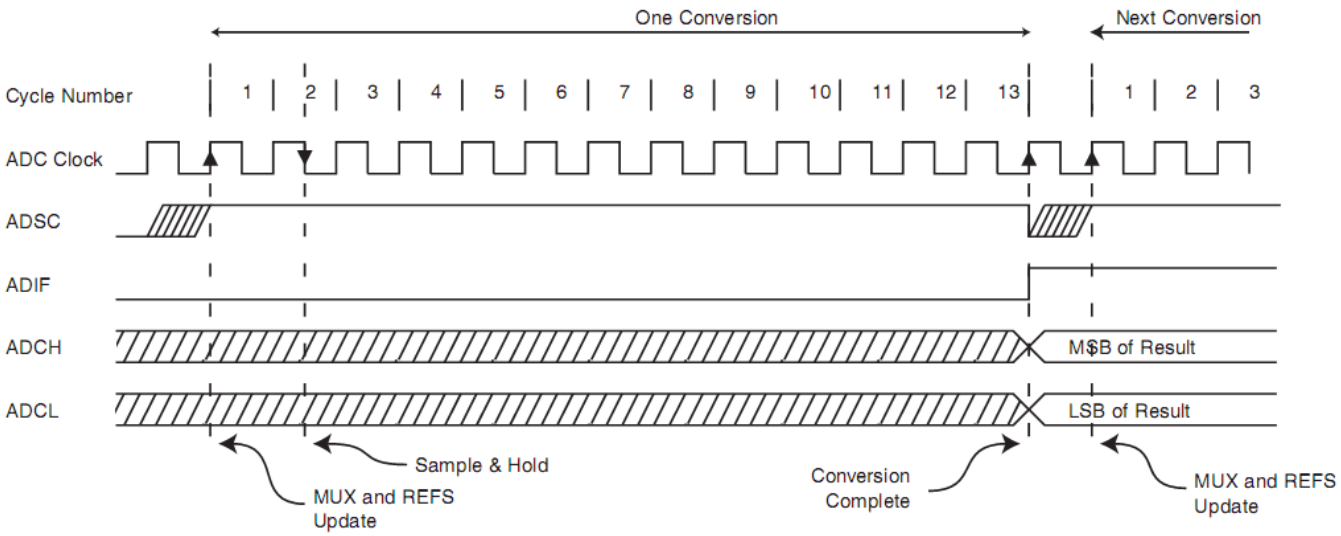


ADTS2	ADTS1	ADTS0	Trigger Source
0	0	0	Free Running mode
0	0	1	Analog Comparator
0	1	0	External Interrupt Request 0
0	1	1	Timer/Counter0 Compare Match
1	0	0	Timer/Counter0 Overflow
1	0	1	Timer/Counter1 Compare Match B
1	1	0	Timer/Counter1 Overflow
1	1	1	Timer/Counter1 Capture Event

7	6	5	4	3	2	1	0	
–	–	–	–	–	ADTS2	ADTS1	ADTS0	ADCSRB
R	R	R	R	R	R/W	R/W	R/W	

Convertor Analog/Digital

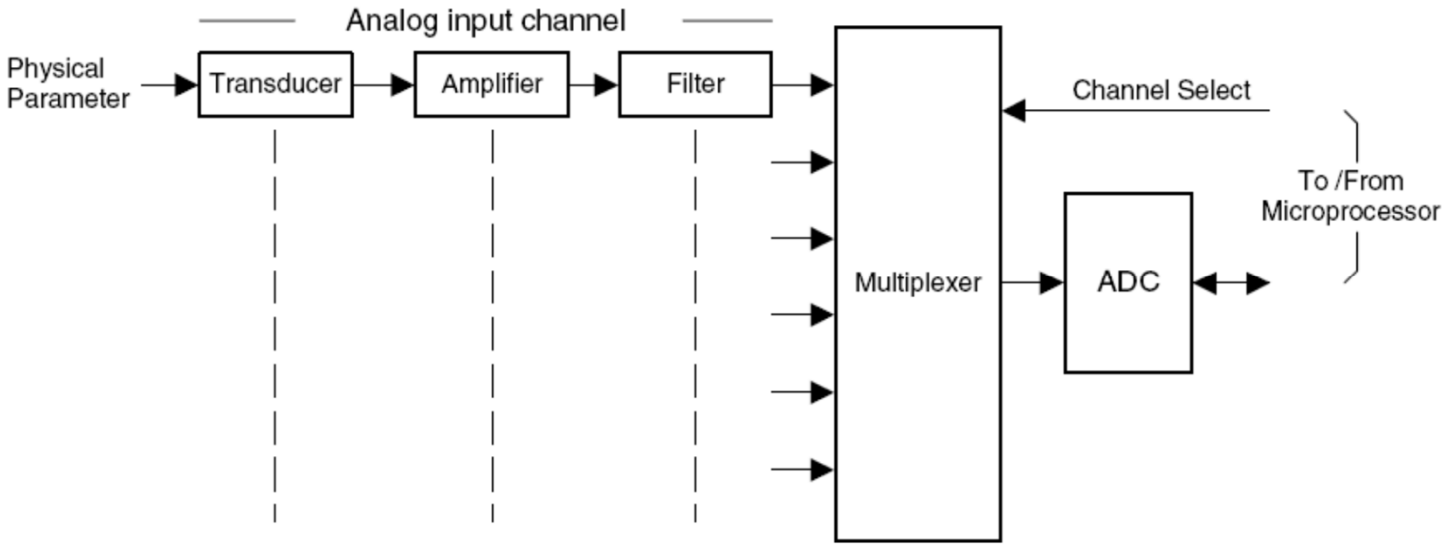
- **Timpi de conversie, diagrame de timp**



Condition	Sample & Hold (Cycles from Start of Conversion)	Conversion Time (Cycles)
First conversion	13.5	25
Normal conversions, single ended	1.5	13
Auto Triggered conversions	2	13.5
Normal conversions, differential	1.5/2.5	13/14

Convertor Analog/Digital

- **Utilizare generală**
- Măsurarea unei mărimi fizice
- Transducer – transformă marimea fizică într-o tensiune (**Senzor**)

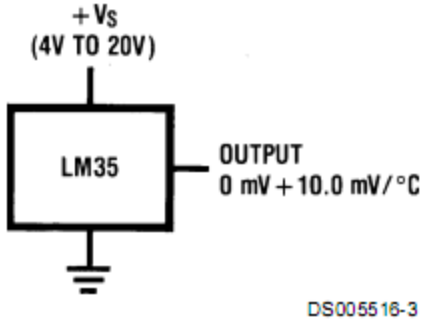


Convertor Analog/Digital

- **Măsurarea temperaturii**
- Senzor: National Semiconductor LM 35
- Voutput = T * 0.01 V
- Intrare single ended:

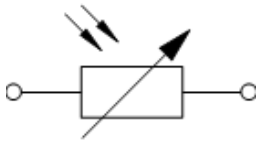
$$ADC = \frac{V_{IN} * 1024}{V_{REF}}$$

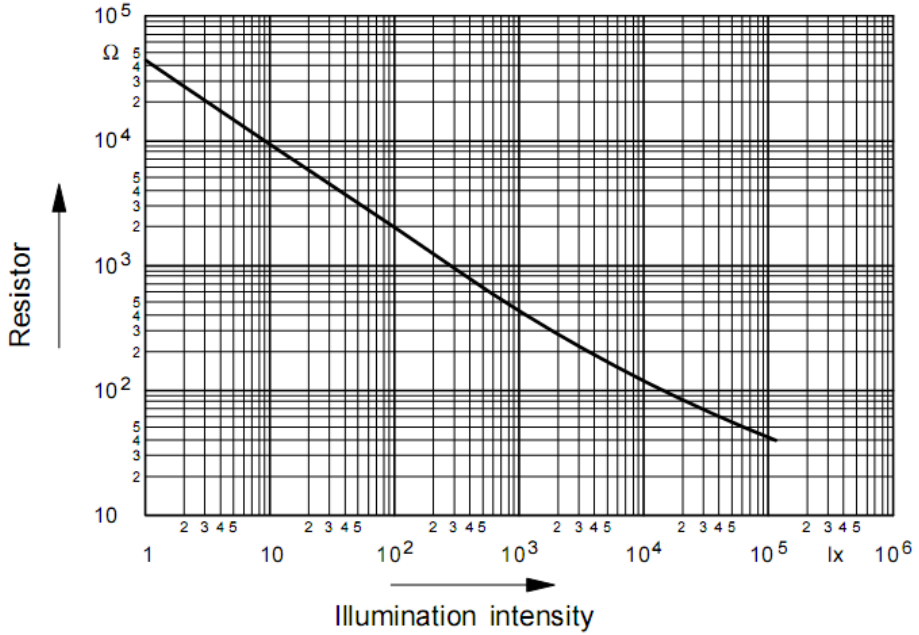
- Daca ADLAR = 1, putem scrie: ADCH = Vin * 256 / Vref
- ADCH = Voutput * 256 / Vref
- ADCH = T * 2,56 / Vref
- Dacă se alege Vref ca **tensiunea interna de referinta 2,56 V**, atunci **ADCH = T** (in grade Celsius)

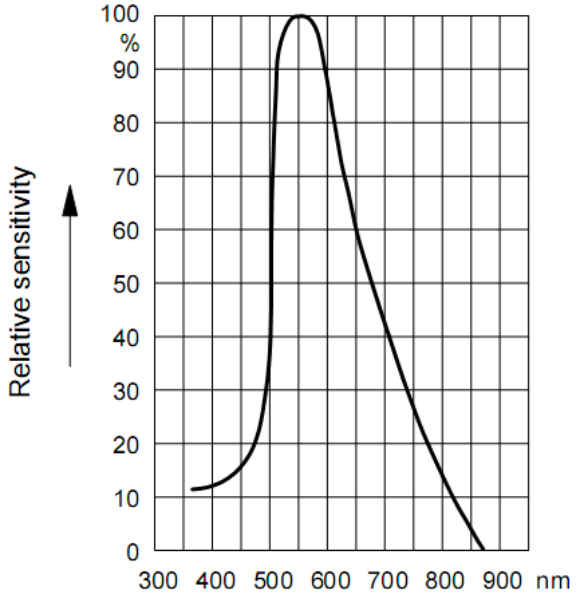


Convertor Analog/Digital

- **Măsurarea intensității luminoase**
  - Soluția cea mai simplă: folosirea unei fotorezistențe (sulfură de cadmiu)
  - Rezistența scade pe măsură ce intensitatea luminoasă crește

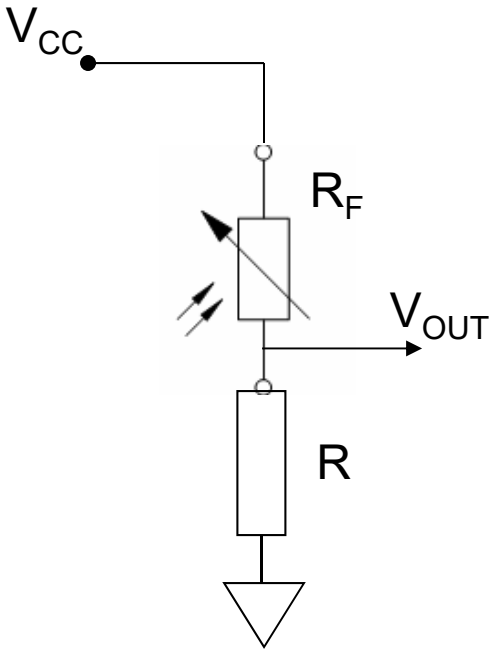






Convertor Analog/Digital

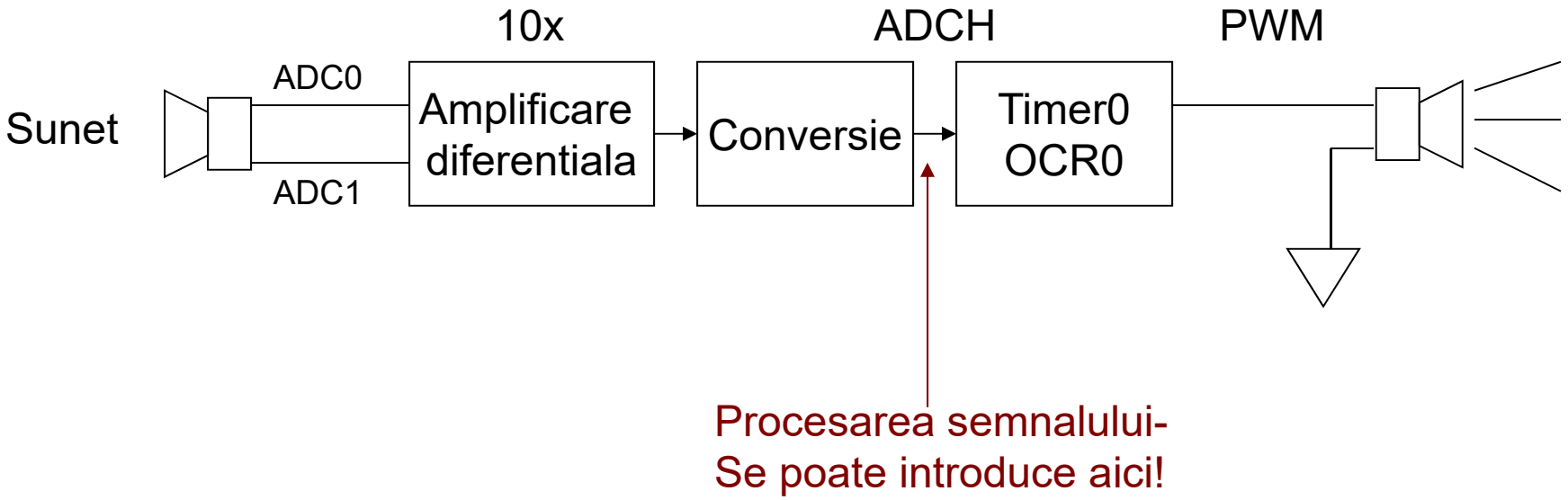
- **Măsurarea intensității luminoase (continuare)**
  - Fotorezistența împreună cu o rezistență fixă formează un divizor de tensiune
  - Relația dintre RF și intensitatea luminoasă trebuie calibrată
  - V_{OUT} se introduce la o ADC intrare single ended, GND comun



$$V_{OUT} = \frac{V_{CC} R}{R + R_F}$$

Convertor Analog/Digital

- **Preluare și redare semnal sonor – exemplu detaliat**
  - Se folosește o intrare analogică diferențială (ADC0 și ADC1), pentru că oferă amplificare internă (1x, 10x, 200x)
  - Sursa de sunet: ieșire audio PC, sau MP3 player. Se poate folosi și un microfon dinamic, cu rezultate foarte slabe
  - Redarea la iesire: folosirea Timer0, modul PWM phase correct
  - Amplitudinea citita de catre ADC este folosita pentru a varia lățimea pulsului PWM



Convertor Analog/Digital

- **Preluare și redare semnal sonor – codul sursă**

```
.org 0x0020
    rjmp    timer0_overflow     ; Timer Overflow ISR
main:
    …… Initializare stiva !
    ldi r16, 0b01100001         ; PWM phase correct, viteza maxima
    out TCCR0, r16
    ldi r16, 0xff
    out DDRB, r16               ; Iesire OC0
    ldi r16, 0b00000001         ; Activare intrerupere overflow timer
    out TIMSK, r16
    rcall ADC_init              ; Initializar ADC – va fi detaliat
    sei
loop:
    rcall sample_adc            ; Porneste o conversie
    rcall wait_adc              ; Asteptare terminare conversie
    rcall ADC_read              ; Citire rezultat in r16
    rjmp loop

timer0_overflow:
    out OCR0, r16               ; Modifica valoarea lui OCR0
    reti
```

Convertor Analog/Digital

- **Preluare și redare semnal sonor – codul sursă**

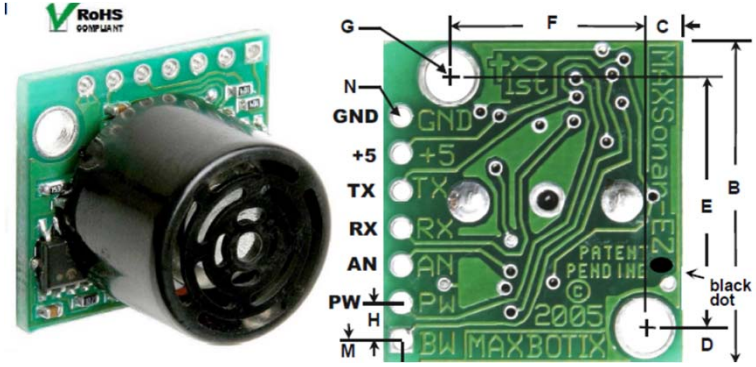
```
ADC_init:
    ldi r16, 0b11101001        ; ADC1+, ADC0-, Amplif 10x, ADLAR=1, Vref=2,56 V intern
    out  ADMUX, r16
    ldi r16, 0b10000000        ; Activare convertor, Viteza maxima
    out  ADCSRA, r16
    ret
sample_adc:
    sbi  ADCSRA, ADSC          ; ADC start, setare bit ADSC din ADCSRA
    ret
wait_adc:
    sbic ADCSRA, ADSC          ; Cand bitul ADSC devine 0, conversia e terminata
    rjmp wait_adc
    ret
ADC_read:
    in   r16, ADCH             ; ADCH este o aproximare pe 8 biti a rezultatului
    mov r18, r16
    andi r18, 0x80             ; Daca rezultatul e negativ, il inversam
    breq ok
    neg r16
ok:
    ret
```

Convertor Analog/Digital

- **Exemplu: masurarea distantei cu senzor Sonar**
  - LV-MaxSonar®-EZ0™ High Performance Sonar Range Finder [7]

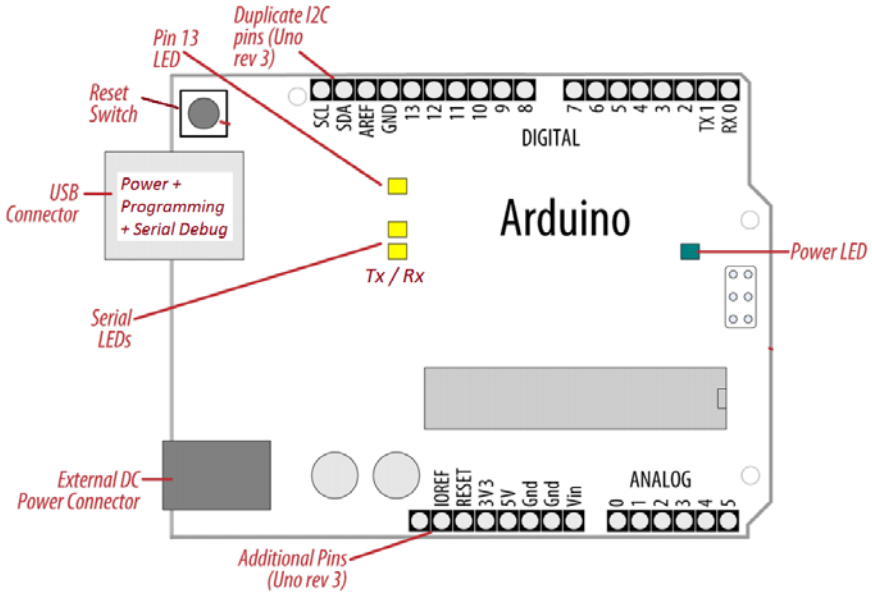
AN – Ieșire analogică, cu factor de scalare (Vcc/512) per inch. O alimentare de 3.3V produce ~6.4mV/in ≈ **2.56 mV/cm**
Domeniu de distanta: 6-in (15 cm) .. 254 in (645 cm)
rezolutie 1-inch

$$ADC = \frac{V_{IN} \times 1024}{V_{REF}} = \frac{2.56mV \times d[cm] \times 1024}{2.56[V]} \approx d[cm]$$




```
ADC_Init:
  ldi r16, 0b11000011          ; Vref=2,56 V internal, ADLAR=0 (Data Shift right – full 1024 bit resolution),
  out ADMUX, r16               ; ADC3 single ended
  ldi r16, 0b10000000          ; Activare ADC, viteza maxima
  out ADCSRA, r16
ret
ADC_read:
  in r20, ADCL                 //ADC citire registru inferior
  in r21, ADCH                 //ADC citire registru superior        // r21:r20 = d[cm]
ret
```

Procesare semnal analogic cu Arduino



Arduino UNO: A0 .. A5

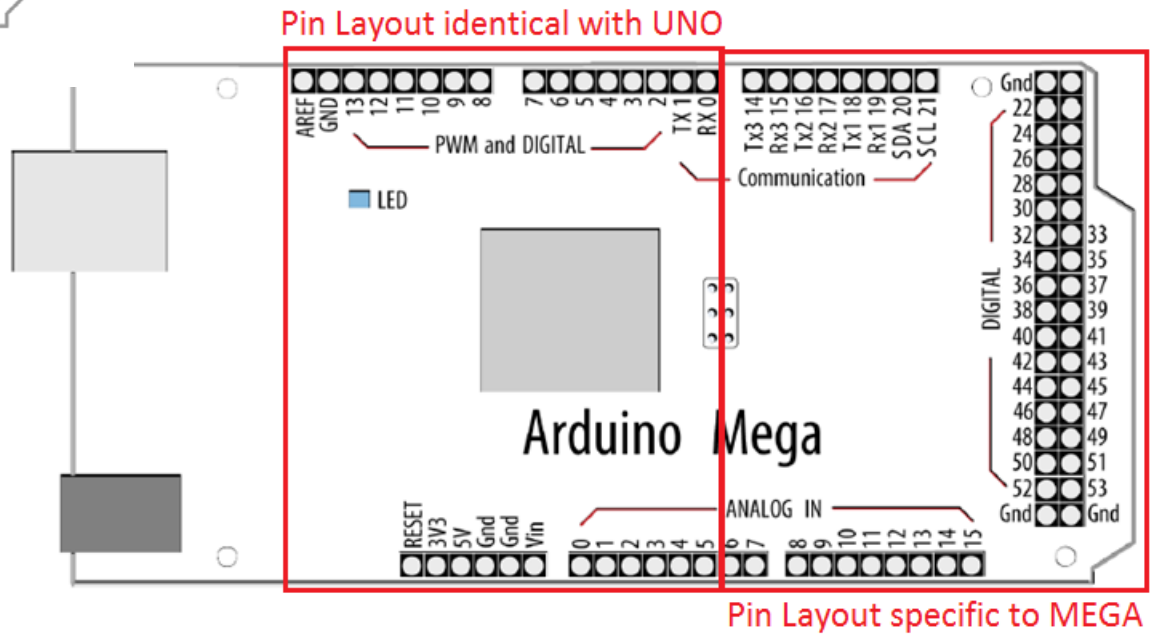
Arduino MEGA: A0 .. A15

- Pinii analogici sunt intrări pentru convertorul ADC al microcontrollerului.
- ADC are rezoluția de 10 biți, returnând valori între 0 și 1023

Alți pini:

A_{REF} (intrare) – tensiune de referință externă pentru ADC

IOREF (ieșire) – tensiune de referință pentru shield-uri



Procesare semnal analogic cu Arduino

- **Funcția principală a pinilor analogici**: **citirea semnalelor analogice**
- Pinii analogici au și funcția de pin digital de uz general, ca și pinii digitali. Exemplu:
  - pinMode(A0, OUTPUT);
  - digitalWrite(A0, HIGH);
- Pinii analogici au de asemenea rezistențe pull up resistors, care funcționează în același fel ca rezistențele pinilor digitali. Ele sunt activate scriind HIGH pe pinul configurat ca intrare.
  - digitalWrite(A0, HIGH); // activare pullup la A0 configurat ca *input*.
- Activarea unei rezistențe pull up va influența valorile citite cu analogRead() !!!
- **Funcții:**
  - **analogRead**(pin) – citeste o valoare de pe un pin analogic
  - **analogReference**(type) – configurează tensiunea de referință care va fi folosită pentru intrarea analogică (i.e. valoarea maximă a tensiunii de intrare masurabilă pe pin-ul analogic)

- **analogReference(type)** – configurează tensiunea de referință care va fi folosită pentru intrarea analogica.
- **type** – stabilește referința folosită:
  - DEFAULT: tensiunea referinta implicita, de 5 V (pentru UNO & MEGA)
  - INTERNAL: tensiune internă de referință, 1.1 V la UNO (*nu exista la Arduino Mega*)
  - INTERNAL1V1: tensiune internă de referință 1.1V (doar *Arduino Mega*)
  - INTERNAL2V56: tensiune internă 2.56V (doar *Arduino Mega*)
  - EXTERNAL: tensiune de referință externa, aplicată la pinul AREF (**0 … 5V**).

- După schimbarea tensiunii de referință, prima citire cu analogRead() poate fi eronată !!!
- **Nu folosiți o tensiune de referință externa negativă (<0V) sau mai mare de 5V pe pinul AREF! Dacă folosiți o tensiune externă de referință, configurați referința ca externă apelând analogReference() înainte de a apela funcția analogRead().** In caz contrar, veți pune in contact tensiunea de referință internă, generată în mod activ, cu tensiunea externă, putând cauza scurtcircuit și distrugerea microcontrollerului. !!!

Procesare semnal analogic cu Arduino

- int *digital_value* **analogRead**(*pin*) – citește o valoare de pe pinul analogic specificat
- O valoare analogica intre 0 .. RANGE va produce un numar *digital_value* intre 0 și 1023.
- Rezoluția de măsurare este deci: RANGE volți / 1024 unități.
- Pentru referința DEFAULT (5V) rezoluția devine:

  resolutionADC = .0049 volti (4.9 mV) / unitate.
- Pentru a converti valoarea citită *digital_value* la tensiunea analogică:
- ***Voltage*** = ***resolutionADC*** * ***digital_value***
- Durează aproximativ 100 microsecunde (0.0001 s) pentru a citi o intrare analogică, astfel incât rata maximă de citire este 10000 valori pe secundă.

- Dacă pinul analogic nu este conectat la nimic, valoarea returnată de analogRead() va fluctua în functie de mai multi factori (e.g. ce tensiuni sunt pe ceilalți pini analogici, apropierea mâinii de placă…) !!!

Procesare semnal analogic cu Arduino

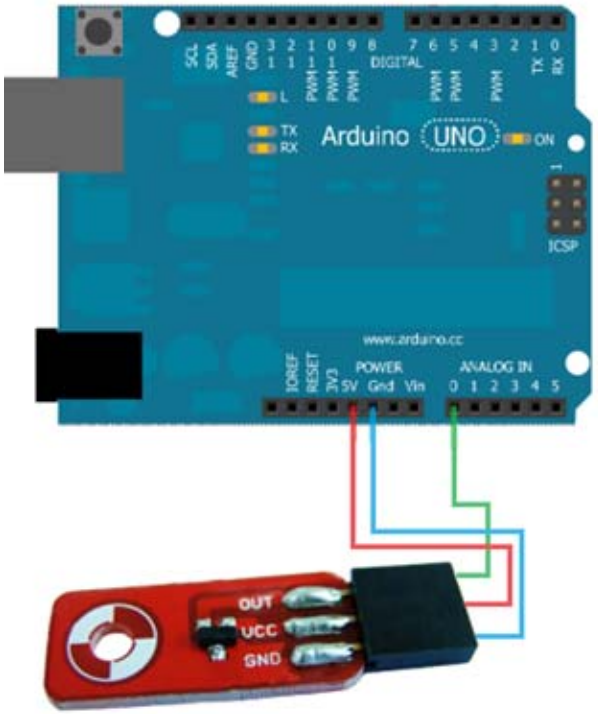
- **Exemplu** – Citirea tensiunii unui potențiometru conectat la intrarea analogică (http://arduino.cc/en/Reference/AnalogRead)

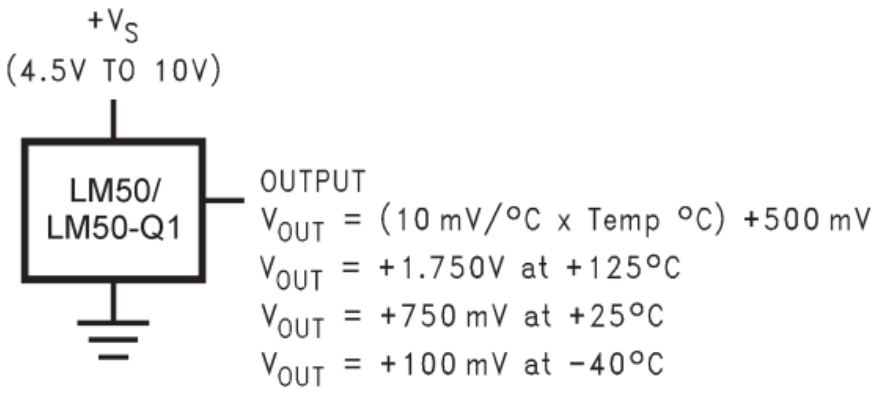
```
int analogPin = 3;                  // aici se va conecta cursorul potentiometrului
                                    // celelalte terminale ale potentiometrului se conecteaza la +5V si GND
int val = 0;                        // variabila in care se va citi valoarea analogica
float voltage;                      // tensiunea calculata, in [mV]
float resolutionADC = 4.9;          // rezolutia in mV pentru referinta implicita de 5 V
void setup()
{
  Serial.begin(9600);
}

void loop()
{
  val = analogRead(analogPin);      // citire intrare analogica, pentru referinta 5 V
  voltage = val * resolutionADC;    // conversie in mV
  Serial.print("Digital value = ");
  Serial.println(val);              //  transmitere la PC
  Serial.print("Voltage [mV] = ");
  Serial.println(voltage);
}
```

Procesare semnal analogic cu Arduino

- **Senzor de temperatură folosind LM50 [5]**
- Caracteristici:
  - Iesire liniara +10.0 mV/°C = 0.01V/°C
  - Domeniu de temperaturi −40°C … +125°C
  - Deplasament constant +500 mV pentru citirea temperaturilor negative
  - Circuitul LM50 este inclus in senzorul de temperatura Brick [6]





- **Exemplu** – Citire temperatură de la senzor, face media a 10 citiri consecutive, și trimite către PC

```
float resolutionADC  = .0049 ;       // rezolutia implicita (pentru referinta 5V) = 0.049 [V] / unitate
float resolutionSensor  = .01 ;      // rezolutie senzor = 0.01V/°C

void setup() {
   Serial.begin(9600);
}
void loop(){
   Serial.print("Temp [C]: ");
   float temp = readTempInCelsius(10, 0);  // citeste temperatura de 10 ori, face media
   Serial.println(temp);                   // afisare
   delay(200);
}
float readTempInCelsius(int count, int pin) { // citeste temperatura de count ori de pe pinul analogic pin
   float sumTemp = 0;
   for (int i =0; i < count; i++) {
     int reading = analogRead(pin);
     float voltage = reading * resolution;
     float tempCelsius = (voltage - 0.5) / resolutionSensor ;  // scade deplasament, converteste in grade C
     sumTemp = sumTemp + tempCelsius;                          // suma temperaturilor
   }
   return sumTemp / (float)count;                              // media returnata
}
```

- **Exemplu** – Măsurare distanțe cu sonarul LV EZ0 (rezolutie 10mV / inch ≅ 0.01V)

```
const int sensorPin = 1;                // Iesire sonar conectata la A1
float resolutionADC  = .0049 ;          // rezolutia implicita (pentru referinta 5V) = 0.049 [V] / unitate
float resolutionSensor  = .01 ;         // rezolutie senzor = 0.01V/ inch

void setup() {
   Serial.begin(9600);
}
void loop() {
   float distance = readDistance(10, sensorPin );                    // distanta in inch, media a 10 citiri
   Serial.print("Distance [inch]: ");  Serial.println(distance);      // afisare distanta in inch
   Serial.print("Distance [cm]: ");  Serial.println(distance*2.54); // afisare distanta in cm, 1 inch=2.54 cm
   delay(200);
}
float readDistance(int count, int pin) {
  // citeste de 10 ori distanta, si face media
  float sumDist = 0;
  for (int i =0; i < count; i++) {
      int reading = analogRead(pin);
      float voltage = reading * resolution;
      float distance = voltage / resolutionSensor;  // conversie tensiune in distanta
      sumDist = sumDist + distance;
   }
   return sumDist / (float)count;
}
```

Referințe

1. Atmel ATmega640/V-1280/V-1281/V-2560/V-2561/V datasheet
2. Atmel Atmega64 datasheet
3. http://arduino.cc/en/
4. http://arduino.cc/en/Reference/AnalogRead
5. http://www.ti.com/lit/ds/symlink/lm50.pdf
6. http://www.robofun.ro/senzori/vreme/senzor-temperatura-brick
7. http://maxbotix.com/documents/LV-MaxSonar-EZ_Datasheet.pdf