

Aluno: Ademir Camillo Junior
Professor: Rafael Rodrigues Obelheiro
Disciplina: ASC – PGCA
Trabalho Final

Plano de Trabalho

O Problema

Com o crescimento dos dispositivos conectados surge o tema chamado de IoT, *Internet of Things*, e com isso, surgem também diversas opções para os desenvolvedores criarem suas próprias soluções que farão parte desse cenário. Entretanto, saber qual equipamento escolher para desenvolver a sua solução pode ser um fator decisivo para o sucesso ou não do equipamento. Alguns fatores que podemos citar como importantes são *form factory* (tamanho do componente), memória, consumo de energia e processamento.

Objetivo

Analisar três micro controladores disponíveis e bastante utilizados por desenvolvedores de soluções IoT para identificar o desempenho (Velocidade e Tempo) de cada um deles referente a processamento de informações.

Métricas

Para realizar a avaliação de desempenho serão efetuados testes de ordenação de vetores em cada um dos micros controladores escolhidos, a fim de verificar o tempo gasto em **microsegundos** por cada um deles para realizar a ordenação de uma determinada quantidade de números. Em cada etapa, serão executados testes através dos seguintes procedimentos:

- Teste 1 - 50
 - a. Vetor com 50 registros
 - b. Ordenação através do algoritmo Bubble Sort
 - c. Repetição da ordenação 20 vezes
 - d. Coleta dos dados de tempo de cada ordenação
 - e. Vetor com 50 registros
 - f. Ordenação através do algoritmo Selection Sort
 - g. Repetição da ordenação 20 vezes
 - h. Coleta dos dados de tempo de cada ordenação
- Teste 2 – 100
 - a. Vetor com 100 registros
 - b. Ordenação através do algoritmo Bubble Sort
 - c. Repetição da ordenação 20 vezes
 - d. Coleta dos dados de tempo de cada ordenação
 - e. Vetor com 100 registros
 - f. Ordenação através do algoritmo Selection Sort
 - g. Repetição da ordenação 20 vezes
 - h. Coleta dos dados de tempo de cada ordenação
- Teste 3 – 150
 - a. Vetor com 150 registros
 - b. Ordenação através do algoritmo Bubble Sort
 - c. Repetição da ordenação 20 vezes

- d. Coleta dos dados de tempo de cada ordenação
- e. Vetor com 150 registros
- f. Ordenação através do algoritmo Selection Sort
- g. Repetição da ordenação 20 vezes
- h. Coleta dos dados de tempo de cada ordenação

Tabela de Coleta de Ddados

Bubble Sort - 50				
Qtde	ATMega328 16MHz	ATMega328 8MHz	Attiny 85 8MHz	Attiny85 1MHz
1	6664	13488	34688	278464
2	6864	13408	34632	279168
3	6592	13472	34808	278336
4	6784	13336	34616	278464
5	6744	13456	34640	278336
6	6768	13464	34896	277888
7	6748	13288	34672	276800
8	6748	13552	34744	278336
9	6720	13392	34760	277120
10	6700	13496	34784	278016
11	6716	13496	34712	278272
12	6584	13504	34880	279232
13	6704	13384	34768	277888
14	6632	13440	34856	279040
15	6736	13368	34952	277376
16	6712	13560	34696	278976
17	6864	13584	34784	278208
18	6720	13368	34888	277888
19	6696	13448	34784	277952
20	6748	13272	34744	279040
Média	6722	13439	34765	278240

Tabela 1. BubbleSort – 50 Registros (microseg)

Selection Sort - 50				
Qtde	ATMega328 16MHz	ATMega328 8MHz	Attiny 85 8MHz	Attiny85 1MHz
1	7716	15448	36776	294272
2	7720	15424	36784	294144
3	7728	15440	36784	294208
4	7724	15440	36768	294208
5	7708	15424	36776	294272
6	7736	15472	36776	294272
7	7708	15448	36760	294336
8	7712	15440	36776	294272
9	7728	15440	36776	294272
10	7716	15448	36784	294144
11	7720	15440	36752	294208

12	7720	15440	36776	294080
13	7716	15440	36768	294272
14	7712	15456	36784	294080
15	7724	15440	36776	294272
16	7720	15448	36768	294336
17	7712	15432	36784	294272
18	7720	15464	36768	294272
19	7728	15464	36768	294144
20	7728	15464	36776	294208
Média	7720	15446	36774	294227

Tabela 2. Selection Sort – 50 Registros (microseg)

Bubble Sort - 100				
Qtde	ATMega328 16MHz	ATMega328 8MHz	Attiny 85 8MHz	Attiny85 1MHz
1	16852	34144	76288	613376
2	16960	34392	76696	612480
3	16948	34160	76296	612224
4	17112	33784	77112	616576
5	17008	34296	76776	613504
6	16992	34520	76928	611136
7	16980	34192	77040	614912
8	16944	34248	76592	611264
9	16960	34152	77248	615872
10	17092	33592	76672	617408
11	16956	33912	76672	614400
12	16992	34352	76600	613312
13	17104	34184	77240	614080
14	17108	33600	76456	614848
15	17128	34072	76616	616192
16	17064	33928	76608	612096
17	17008	33808	76768	614592
18	17008	34192	76408	615296
19	17216	34056	77072	614144
20	17128	33600	76656	614656
Média	17028	34059	76737	614118

Tabela 3. BubbleSort – 100 Registros (microseg)

Selection Sort - 100				
Qtde	ATMega328 16MHz	ATMega328 8MHz	Attiny 85 8MHz	Attiny85 1MHz
1	20932	41856	84536	676480
2	20936	41872	84544	676352
3	20956	41888	84544	676224
4	20944	41872	84544	676544
5	20940	41872	84560	676288
6	20936	41864	84552	676352
7	20936	41872	84528	676160

8	20940	41904	84528	676288
9	20936	41920	84552	676352
10	20956	41896	84528	676160
11	20948	41872	84544	676480
12	20948	41856	84568	676224
13	20952	41880	84536	676160
14	20948	41888	84544	676480
15	20936	41888	84552	676416
16	20936	41872	84544	676096
17	20960	41856	84552	676288
18	20936	41864	84536	676480
19	20948	41888	84592	676288
20	20944	41864	84552	676416
Média	20943	41877	84547	676326

Tabela 4. Selection Sort – 100 Registros (microseg)

Bubble Sort - 150				
Qtde	ATMega328 16MHz	ATMega328 8MHz	Attiny 85 8MHz	Attiny85 1MHz
1	31388	62376	106056	833792
2	30972	61832	105208	845440
3	31196	61952	105512	839360
4	30944	61936	105520	838720
5	30876	62528	105440	843648
6	30640	61424	105536	838016
7	31176	62008	104760	848640
8	30820	61584	104888	841216
9	30896	62472	105520	840832
10	31044	61552	105640	844288
11	31184	62024	105440	844160
12	30716	62024	105736	841920
13	31484	62216	104816	844224
14	30736	61664	105056	845760
15	31172	61664	105304	840320
16	30880	61552	104520	841984
17	31184	62040	105648	845568
18	30892	62456	105512	843456
19	30896	61640	105704	847552
20	30956	62752	105408	840512
Média	31003	61985	105361	842470

Tabela 5. BubbleSort – 150 Registros (microseg)

Selection Sort - 150				
Qtde	ATMega328 16MHz	ATMega328 8MHz	Attiny 85 8MHz	Attiny85 1MHz
1	39636	79328	118568	947968
2	39664	79344	118552	947904
3	39648	79328	118536	947712

4	39664	79328	118528	947968
5	39664	79328	118568	947712
6	39668	79344	118568	947776
7	39660	79328	118528	947840
8	39664	79352	118568	947776
9	39664	79360	118536	947904
10	39652	79344	118584	947776
11	39676	79360	118536	948032
12	39672	79352	118576	947904
13	39672	79328	118560	947648
14	39668	79328	118552	947840
15	39652	79312	118536	947840
16	39648	79328	118568	947776
17	39640	79336	118560	947840
18	39672	79344	118552	947776
19	39656	79360	118576	947840
20	39668	79312	118544	947776
Média	39660	79337	118555	947830

Tabela 6. Selection Sort – 150 Registros (microseg)

Ambiente – Software e Hardware

Para realizar os testes serão utilizados 4 micro controladores diferentes: ATmega 328PU a 16MHz, ATmega 328P a 8MHz, Attiny85 a 8MHz, Attiny85 a 1MHz. Em cada um deles será carregado o mesmo código fonte com os vetores aleatórios criados e uma rotina para executar os testes, retornando os tempos de cada execução através da porta serial (USB) do computador. O tempo em microssegundos será calculado através da função interna dos microcontroladores AVR chamada `micro()` que retorna o tempo desde que o equipamento foi energizado. Assim, ao iniciar a ordenação conseguimos pegar esse tempo inicial e ao acabar diminuir do tempo final, resultando o tempo de ordenação.

O programa de teste será desenvolvido na IDE de Desenvolvimento do próprio Arduino, que permite a gravação do mesmo código para os quatro micros controladores. Neste software será implementado os algoritmos de ordenação, os vetores a serem ordenados, os testes (de 1 a 3) e as saídas de tempos na porta serial.

Algoritmos de Ordenação

Para realizar o teste, foram escolhidos dois algoritmos de ordenação bastante conhecidos no meio acadêmico: *Bubble Sort* e *Selection Sort*. Conforme descritos abaixo.

Bubble Sort

```
void sort(int a[], int size)
{
    for(int i=0; i<(size-1); i++)
    {
        for(int o=0; o<(size-(i+1)); o++)
        {
            if(a[o] > a[o+1]) {
                int t = a[o];
                a[o] = a[o+1];
            }
        }
    }
}
```

```

        a[o+1] = t;
    }
}
}

```

Selection Sort

```

void sort(int a[], int size)
{
    int i,j;
    for (j = 0; j < n-1; j++)
    {
        int iMin = j;
        for ( i = j+1; i < n; i++)
        {
            if (a[i] < a[iMin])
            {
                iMin = i;
            }
        }

        if(iMin != j)
        {
            temp=a[j];
            a[j]=a[iMin];
            a[iMin]=temp;
        }
    }
}

```

Micro Controladores

Abaixo a descrição completa dos micros controladores que serão utilizados e suas características principais.

MODELO	PROCESSADOR	CLOCK	SRAM	FLASH	VOLTAGEM
Arduino UNO	ATmega328PU	16MHz	2 KB	32 KB	5V
Arduino Pro Mini	ATmega328P	8MHz	1 KB	32 KB	3.3V
Attiny85	Atinny85	8MHz	0,5 KB	8 KB	1.8V