

Andressa Tolfo Meira RA 165417

Thiago Suzuqui Lodi RA 165478

Relatório da prova prática: Projeto PWM pertinente à disciplina de Microcontroladores do curso de Engenharia de Computação e Controle e Automação da Universidade Católica Dom Bosco.

Docente: Edvaldo Francisco Freitas Lima

Objetivos

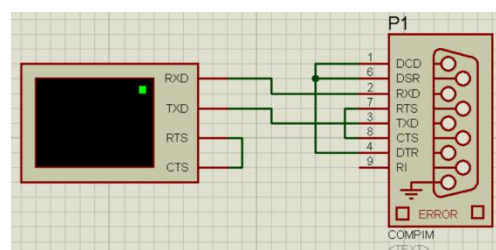
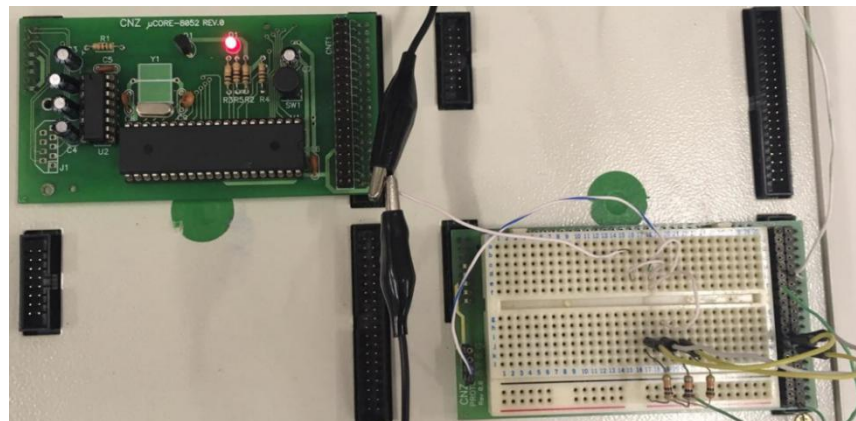
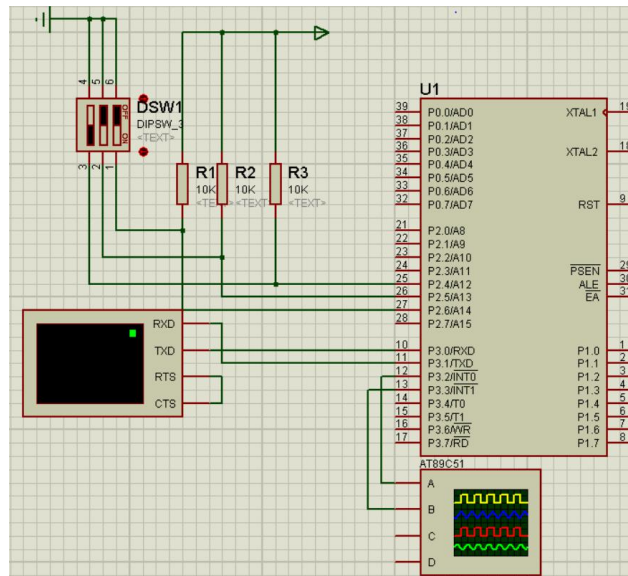
Este experimento tem como objetivo apresentar o funcionamento do PWM a partir de chaves dip-switch ligadas a porta P2 do microcontrolador gerando assim, sinais pré-definidos na porta P3 e implementar também por meio da serial uma interface que incremente e decmente o valor recebido do PWM.

Material Utilizado

- Software EdSim51;
- Software uCore;
- Osciloscópio;
- Multímetro;
- Kit CNZ com módulos uCore-8052 e Proto;

Desenvolvimento

O desenvolvimento do programa baseou-se nos experimentos de PWM e comunicação serial feitos no laboratório, a partir da proposta de projeto foi definido a pinagem do microcontrolador e rotinas de configuração, com a criação de um loop principal que têm a função de ler a combinação do dip-switch e desvia a sub-rotina que a ela corresponde. Logo em seguida, é configurado o tempo que será gerado o sinal de PWM e a largura de pulso. Após a execução do sinal, aguarda o comando da serial para prosseguir com o programa. Para a execução prática do projeto, foi utilizado o Kit CNZ com os módulos 8052 e proto, para montar o circuito.



Fluxograma

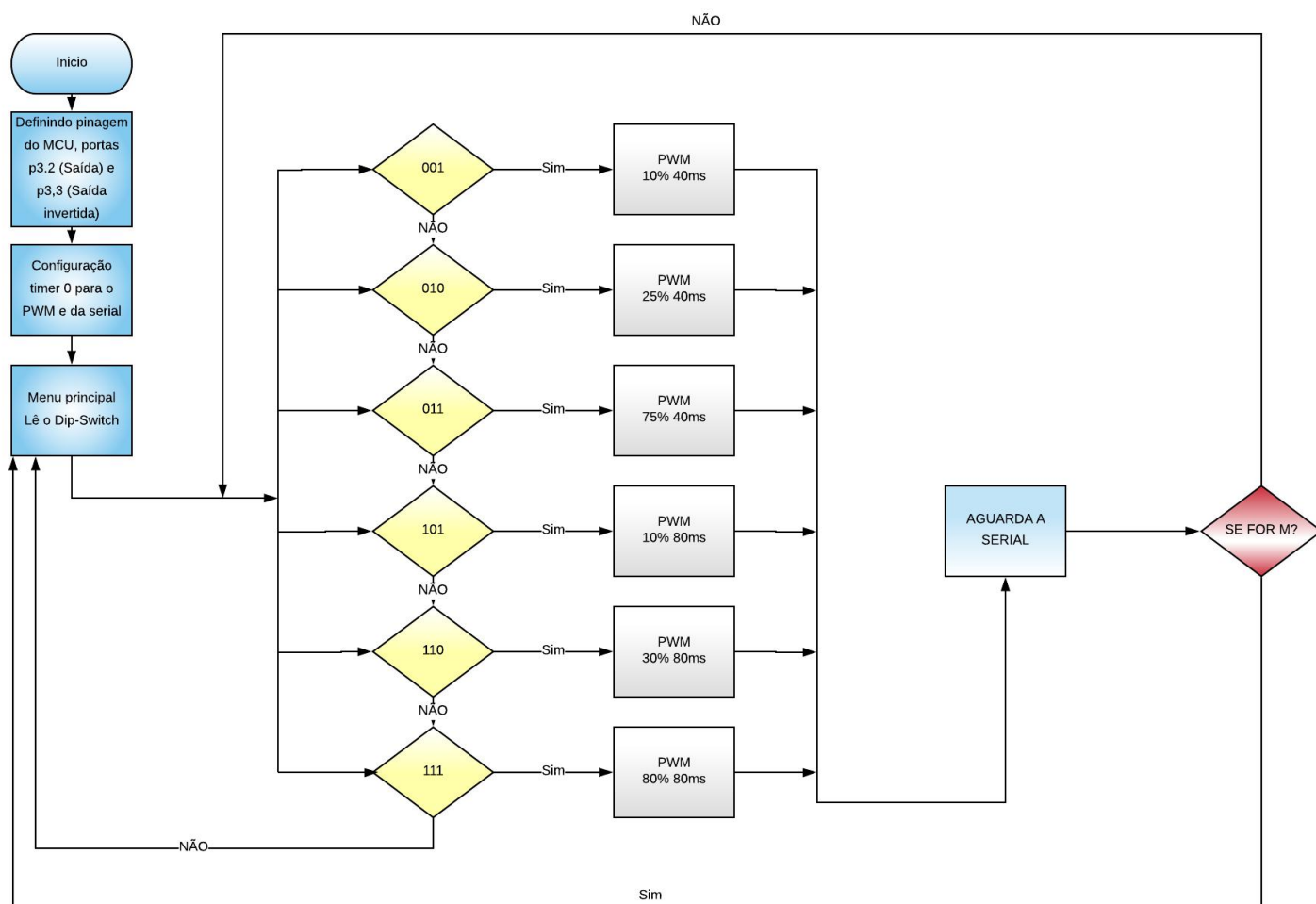


Tabela - cálculo PWM

	padrao	periodo	pct high	us high	hex high	pct low	us low	hex low
.001	65535	40000	5%	2000	F82F	95%	38000	6B8F
	65535	40000	10%	4000	F05F	90%	36000	735F
	65535	40000	15%	6000	E88F	85%	34000	7B2F
	65535	40000	20%	8000	E0BF	80%	32000	82FF
.010	65535	40000	25%	10000	D8EF	75%	30000	8ACF
	65535	40000	30%	12000	D11F	70%	28000	929F
	65535	40000	70%	28000	929F	30%	12000	D11F
	65535	40000	75%	30000	8ACF	25%	10000	D8EF
.011	65535	40000	80%	32000	82FF	20%	8000	E0BF
	65535	80000	5%	4000	F05F	48%	38000	6B8F
	65535	80000	10%	8000	E0BF	45%	36000	735F
	65535	80000	15%	12000	D11F	43%	34000	7B2F
.101	65535	80000	25%	20000	B1DF	75%	60000	159F
	65535	80000	30%	24000	A23F	70%	56000	253F
	65535	80000	35%	28000	929F	65%	52000	34DF
	65535	80000	75%	60000	159F	25%	20000	B1DF
.111	65535	80000	80%	64000	5FF	20%	16000	C17F
	65535	80000	43%	34000	7B2F	15%	12000	D11F

2x

Programa em Assembly

```
1 ;*****
2 ;
3 ;          PROJETO: PWM - Controlado por dip-switch e serial
4 ;*****
5 ;
6 ;
7 ; -> Projeto para criar diferentes pulsos de PWM a partir de uma selecao via dip-switch, tendo a
8 ;     possibilidade de aumentar ou diminuir esse valor a partir de comandos enviados via serial.
9 ;
10 ;-----|
11 ;|          Combinacoes dip-switch          |
12 ;|-----|
13 ;|Entrada      Período      PWM|
14 ;|1001         40ms        10%|
15 ;|1010         40ms        25%|
16 ;|1011         40ms        75%|
17 ;|1101         80ms        10%|
18 ;|1110         80ms        30%|
19 ;|1111         80ms        80%|
20 ;|-----|
21 ;
22 ; -> bits de entrada definidos no port P2 sendo P2.4(msb) P2.5 P2.6
23 ; -> para definicao do periodo do pwm eh preciso fazer o calculo 65535 - periodo(alta/baixo)
24 ;
25 ; -> atraves da comunicacao serial eh possivel enviar os caracteres 'a', 'd' ou 'm' ao MCU para aplicar os comandos:
26 ;     'a' - aumenta o sinal do pwm em + 5% do selecionado via dip-switch
27 ;     'd' - diminui o sinal do pwm em - 5% do selecionado via dip-switch
28 ;     'm' - retorna ao menu de leitura do dip-switch
29 ;
30 ;
31 ; -> uso dos registradores:
32 ;     R0 controla o tempo de PWM
33 ;     R1 armazena a parte alta do periodo on do PWM
34 ;     R2 armazena a parte baixa do periodo on do PWM
35 ;     R3 armazena a parte alta do periodo of do PWM
36 ;     R4 armazena a parte baixa do periodo of do PWM
37 ;
38 ;
39 ;definindo pinagem do MCU
40 SAIDA      EQU      P3.2          ;Saída do PWM
41 saida_inv  equ      P3.3          ;Saída do PWM invertido
42 dip        equ      P2           ;entrada do dip-switch
43
44          ORG      0000H
45          jmp      INICIO
46          ORG      0040H
47
48 ;configuracao timer 0 e da serial
49 INICIO:
50          CLR      SM0             ;|
51          SETB     SM1             ;| coloca serial em 8bits UART mode
52          MOV      TMOD,#21H       ;| coloca timer 0 configurado no modo 1 (16 bits)
53                                     ;| coloca timer 1 em 8-bit e recarga automatica
54          MOV      TCON,#00H       ;| TIMER DESLIGADO
55          MOV      TH1,#0FDH       ;|
56          MOV      TLL,#0FDH       ;| baud rate de 9600
57          SETB     TR1             ;| start timer 1
58          SETB     REN             ;| enable read mode
59 ;fim configuracao
60
61 ;loop principal
62 le:
63          MOV      A,dip           ; Le dipswitch
64          ORL      A,#10001111B    ; mascara para os bits que interessam, p2.6 p2.5 p2.4
65          CJNE     A,#11001111B,a1 ; 001
66          MOV      R0,#200         ; tempo de PWM
67          MOV      r1,#0FDH        ;| periodo 40ms
68          MOV      r2,#05FH        ;| 10% on
69          MOV      r3,#073H        ;|
70          MOV      r4,#05FH        ;| 90% of
71
72          jmp      dez_pc_40ms
73
74
75          jmp      dez_pc_40ms
76 a1:
77          CJNE     A,#10101111B,a2 ; 010
78
79          MOV      R0,#200         ; tempo de PWM
80          mov      r1,#0D8H        ;| periodo 40ms
81          mov      r2,#05FH        ;| 25% on
82          mov      r3,#08AH        ;|
83          mov      r4,#0CFH        ;| 75% of
84
85          jmp      vinte_cinco_pc_40ms
86 a2:
87          CJNE     A,#11101111B,a3 ; 011
88
89          MOV      R0,#200         ; tempo de PWM
90          mov      r1,#08AH        ;| periodo 40ms
91          mov      r2,#0CFH        ;| 25% on
92          mov      r3,#0D8H        ;|
93          mov      r4,#0EFH        ;| 75% of
94
95          jmp      setenta_cinco_pc_40ms
96 a3:
97          CJNE     A,#11011111B,a4 ; 101
98
99          MOV      R0,#200         ; tempo de PWM
100         mov      r1,#0E0H        ;| periodo 80ms
101         mov      r2,#0BFH        ;| 10% on
102         mov      r3,#073H        ;|
103         mov      r4,#05FH        ;| 90% of
104
105         jmp      dez_pc_80ms
106 a4:
107         CJNE     A,#10111111B,a5 ; 110
108
109         MOV      R0,#100         ; tempo de PWM
110         mov      r1,#0A2H        ;| periodo 80ms
111         mov      r2,#03FH        ;| 30% on
```

```

112      mov     r3,#025H      ;|
113      mov     r4,#03FH      ;| 70% of
114
115      jmp     trinta_pc_80ms
116
117      a5:     CJNE     A,#11111111B,le      ; 111
118
119      MOV     R0,#200        ; tempo de PWM
120      mov     r1,#005H      ;| periodo 80ms
121      mov     r2,#0FFH      ;| 80% on
122      mov     r3,#0C1H      ;|
123      mov     r4,#07FH      ;| 20% of
124
125      jmp     oitenta_pc_80ms
126 ;fim loop principal
127
128 ;sub-rotinas para gerar pulsos do pwm
129
130 ;*****
131 ;*      10% (HIGH)
132 ;*
133 ;*      90% (LOW)
134 ;* periodo 40ms / 40000us
135 ;*****
136 dez_pc_40ms:
137      MOV     TH0,r1        ; CARREGA TIMER 0 COM 4ms 10%
138      MOV     TL0,r2        ; EM ALTO
139      SETB    TR0           ; LIGA TIMER 0
140      SETB    SAIDA        ; POE SAIDA EM UM
141      CLR     saida_inv
142
143      JNB     TF0,$         ; ESPERA OVERFLOW
144      CLR     TR0           ; DESLIGA TIMER
145      CLR     TF0          ; ZERA FLAG DE OVERFLOW
146
147      MOV     TH0,r3        ; CARREGA TIMER 0 COM 36ms 90%
148      MOV     TL0,r4        ; EM baixo
149
150      SETB    TR0           ; LIGA TIMER0
151      CLR     SAIDA        ; POE SAIDA EM NIVEL ZERO
152      setb    saida_inv
153
154      JNB     TF0,$         ; ESPERA OVERFLOW DO TIMER
155      CLR     TR0           ; DESLIGA TIMER
156      CLR     TF0          ; ZERA FLAG DE OVERFLOR DO TIMER
157
158      DJNZ    R0,dez_pc_40ms ; FAZ ENSTA ONDA POR 5 SEGUNDOS
159      call    serial_dez_pc_40ms
160
161      jmp     le
162
163 ;*****
164 ;*      25% (HIGH)
165 ;*
166 ;*      75% (LOW)
167 ;* periodo 40ms / 40000us
168 ;*****
169 vinte_cinco_pc_40ms:
170      MOV     TH0,r1        ; CARREGA TIMER 0 COM 10000 us / 10ms / 25%
171      MOV     TL0,r2        ; EM ALTO
172      SETB    TR0           ; LIGA TIMER 0
173      SETB    SAIDA        ; POE SAIDA EM UM
174      clr     saida_inv
175
176      JNB     TF0,$         ; ESPERA OVERFLOW
177      CLR     TR0           ; DESLOGA TIMER
178      CLR     TF0          ; ZERA FLAG DE OVERFLOW
179
180      MOV     TH0,r3        ; CARREGA TIMER 0 COM 30000us / 30ms / 75%
181      MOV     TL0,r4        ; EM BAIXO
182      SETB    TR0           ; LIGA TIMER 0
183      CLR     SAIDA        ; POE SAIDA EM NIVEL ZERO
184      setb    saida_inv
185
186      JNB     TF0,$         ; ESPERA OVERFLOW
187
188      CLR     TR0           ; DESLIGA TIMER
189      CLR     TF0          ; ZERA FLAG DE OVERFLOW DO TIMER
190      DJNZ    R0,vinte_cinco_pc_40ms
191      call    serial_vinte_cinco_pc_40ms
192      jmp     le
193
194 ;*****
195 ;*      75% (HIGH)
196 ;*
197 ;*      25% (LOW)
198 ;* periodo 40ms / 40000us
199 ;*****
200 setenta_cinco_pc_40ms:
201      MOV     TH0,r1        ; CARREGA TIMER 0 COM 30000us / 30ms / 75%
202      MOV     TL0,r2        ; EM alta
203      SETB    TR0           ; LIGA TIMER 0
204      SETB    SAIDA        ; COLOCA SAIDA EM um
205      CLR     saida_inv
206
207      JNB     TF0,$         ; ESPERA OVERFLOW
208      CLR     TR0           ; DESLIGA TIMER
209      CLR     TF0          ; ZERA FLAG DE OVERFLOW
210
211      MOV     TH0,r3        ; CARREGA TIMER 0 10000us / 10ms / 25%
212      MOV     TL0,r4        ; EM BAIXA
213      SETB    TR0           ; LIGA TIMER
214      CLR     SAIDA        ; COLOCA SAIDA EM zero
215      setb    saida_inv
216
217      JNB     TF0,$         ; ESPERA OVERFLOW
218
219      CLR     TR0           ; DESLIGA TIMER
220      CLR     TF0          ; ZERA FLAG DE OVERFLOW
221      DJNZ    R0,setenta_cinco_pc_40ms
222

```

```

223      call      serial_setenta_cinco_pc_40ms
224
225      jmp      le
226
227
228      ;*****
229      ;*      10% (HIGH)
230      ;*
231      ;* periodo 80ms / 80000us
232      ;*
233      ;*****
234      dez_pc_80ms:
235      ;parte alta
236      MOV      TH0,r1      ; CARREGA TIMER 0 COM 8000us / 8ms / 10%
237      MOV      TL0,r2      ; EM alta
238      SETB     TR0         ; LIGA TIMER 0
239      SETB     SAIDA       ; COLOCA SAIDA EM um
240      CLR      saida_inv
241
242      JNB      TF0,$        ; ESPERA OVERFLOW
243      CLR      TR0         ; DESLIGA TIMER
244      CLR      TF0         ; ZERA FLAG DE OVERFLOW
245
246      ;parte baixa divide o periodo de 90% em dois de 45% e faz 2x o de 45%
247      ;primeira vez
248      MOV      TH0,r3      ; CARREGA TIMER 0 36000us / 36ms / 45%
249      MOV      TL0,r4      ; EM BAIXA
250      SETB     TR0         ; LIGA TIMER
251      CLR      SAIDA       ; COLOCA SAIDA EM zero
252      setb     saida_inv
253
254      JNB      TF0,$        ; ESPERA OVERFLOW
255      CLR      TR0         ; DESLIGA TIMER
256      CLR      TF0         ; ZERA FLAG DE OVERFLOW
257
258      ;segunda vez
259      MOV      TH0,r3      ; CARREGA TIMER 0 36000us / 36ms / 45%
260      MOV      TL0,r4      ; EM BAIXA
261      SETB     TR0         ; LIGA TIMER
262      CLR      SAIDA       ; COLOCA SAIDA EM zero
263      setb     saida_inv

```

```

264      JNB      TF0,$        ; ESPERA OVERFLOW
265      CLR      TR0         ; DESLIGA TIMER
266      CLR      TF0         ; ZERA FLAG DE OVERFLOW
267
268      DJNZ     R0,dez_pc_80ms
269
270      call     serial_dez_pc_80ms
271
272      jmp      le
273
274      ;*****
275      ;*      30% (HIGH)
276      ;*
277      ;* periodo 80ms / 80000us
278      ;*
279      ;*****
280      trinta_pc_80ms:
281      MOV      TH0,r1      ; CARREGA TIMER 0 COM 24000us / 24ms / 30%
282      MOV      TL0,r2      ; EM alta
283      SETB     TR0         ; LIGA TIMER 0
284      SETB     SAIDA       ; COLOCA SAIDA EM um
285      CLR      saida_inv
286
287      JNB      TF0,$        ; ESPERA OVERFLOW
288      CLR      TR0         ; DESLIGA TIMER
289      CLR      TF0         ; ZERA FLAG DE OVERFLOW
290
291      MOV      TH0,r3      ; CARREGA TIMER 0 56000us / 56ms / 70%
292      MOV      TL0,r4      ; EM BAIXA
293      SETB     TR0         ; LIGA TIMER
294      CLR      SAIDA       ; COLOCA SAIDA EM zero
295      setb     saida_inv
296
297      JNB      TF0,$        ; ESPERA OVERFLOW
298      CLR      TR0         ; DESLIGA TIMER
299      CLR      TF0         ; ZERA FLAG DE OVERFLOW
300
301      DJNZ     R0,trinta_pc_80ms
302
303      call     serial_trinta_pc_80ms
304

```

```

305      jmp      le
306
307      ;*****
308      ;*      80% (HIGH)
309      ;*
310      ;* periodo 80ms / 80000us
311      ;*
312      ;*****
313      oitenta_pc_80ms:
314      MOV      TH0,r1      ; CARREGA TIMER 0 COM 64000us / 64ms / 80%
315      MOV      TL0,r2      ; EM alta
316      SETB     TR0         ; LIGA TIMER 0
317      SETB     SAIDA       ; COLOCA SAIDA EM um
318      CLR      saida_inv
319
320      JNB      TF0,$        ; ESPERA OVERFLOW
321      CLR      TR0         ; DESLIGA TIMER
322      CLR      TF0         ; ZERA FLAG DE OVERFLOW
323
324      MOV      TH0,r3      ; CARREGA TIMER 0 16000us / 16ms / 20%
325      MOV      TL0,r4      ; EM BAIXA
326      SETB     TR0         ; LIGA TIMER
327      CLR      SAIDA       ; COLOCA SAIDA EM zero
328      setb     saida_inv
329
330      JNB      TF0,$        ; ESPERA OVERFLOW
331      CLR      TR0         ; DESLIGA TIMER
332      CLR      TF0         ; ZERA FLAG DE OVERFLOW
333
334      DJNZ     R0,oitenta_pc_80ms
335
336      call     serial_oitenta_pc_80ms
337
338      jmp      le
339      ;fim das sub-rotinas de gerar pwm
340
341
342      ;sub-rotina da serial
343      ;-----
344      ;      serial_dez_pc_40ms
345      ;-----
346      serial_dez_pc_40ms:

```



```

347         jnb         ri,$           ;aguarda receber cmd pela serial
348         mov         a,sbuf         ;salva o valor da serial no acc
349         clr         ri             ;limpa o flag da serial
350
351         CJNE        a,#6DH,cmp_serial_dez_pc_40ms
352         jmp         le             ;se teclar m volta pro menu
353 cmp_serial_dez_pc_40ms:
354         CJNE        a,#61H,dim_serial_dez_pc_40ms
355
356         mov         r0,#200        ; carrega tempo de PWM
357
358         mov         r1,#0E8h       ;|
359         mov         r2,#08Fh       ;|parte alta 15%
360         mov         r3,#07Bh       ;|
361         mov         r4,#02Fh       ;|parte baixa 85%
362
363         jmp         dez_pc_40ms
364 dim_serial_dez_pc_40ms:
365         mov         r0,#200        ; carrega tempo de PWM
366
367         mov         r1,#0F8h       ;|
368         mov         r2,#02Fh       ;|parte alta 5%
369         mov         r3,#06Bh       ;|
370         mov         r4,#08Fh       ;|parte baixa 95%
371
372         jmp         dez_pc_40ms
373
374
375 ;-----
376 ; serial_vinte_cinco_pc_40ms
377 ;-----
378 serial_vinte_cinco_pc_40ms:
379         jnb         ri,$           ;aguarda receber cmd pela serial
380         mov         a,sbuf         ;salva o valor da serial no acc
381         clr         ri             ;limpa o flag da serial
382
383         CJNE        a,#6DH,cmp_serial_vinte_cinco_pc_40ms
384         jmp         le             ;se teclar m volta pro menu
385 cmp_serial_vinte_cinco_pc_40ms:
386         CJNE        a,#61H,dim_serial_vinte_cinco_pc_40ms
387

```

```

388         mov         r0,#200        ; carrega tempo de PWM
389
390         mov         r1,#0D1h       ;|
391         mov         r2,#01Fh       ;|parte alta 30%
392         mov         r3,#092h       ;|
393         mov         r4,#09Fh       ;|parte baixa 70%
394
395         jmp         vinte_cinco_pc_40ms
396 dim_serial_vinte_cinco_pc_40ms:
397         mov         r0,#200        ; carrega tempo de PWM
398
399         mov         r1,#0E0h       ;|
400         mov         r2,#08Fh       ;|parte alta 20%
401         mov         r3,#082h       ;|
402         mov         r4,#0FFh       ;|parte baixa 80%
403
404         jmp         vinte_cinco_pc_40ms
405
406 ;-----
407 ; serial_setenta_cinco_pc_40ms
408 ;-----
409 serial_setenta_cinco_pc_40ms:
410         jnb         ri,$           ;aguarda receber cmd pela serial
411         mov         a,sbuf         ;salva o valor da serial no acc
412         clr         ri             ;limpa o flag da serial
413
414         CJNE        a,#6DH,cmp_serial_setenta_cinco_pc_40ms
415         jmp         le             ;se teclar m volta pro menu
416 cmp_serial_setenta_cinco_pc_40ms:
417         CJNE        a,#61H,dim_serial_setenta_cinco_pc_40ms
418
419         mov         r0,#200        ; carrega tempo de PWM
420
421         mov         r1,#082h       ;|
422         mov         r2,#0FFh       ;|parte alta 80%
423         mov         r3,#050h       ;|
424         mov         r4,#0BFh       ;|parte baixa 20%
425
426         jmp         setenta_cinco_pc_40ms
427 dim_serial_setenta_cinco_pc_40ms:
428         mov         r0,#200        ; carrega tempo de PWM
429

```

```

430         mov         r1,#092h       ;|
431         mov         r2,#09Fh       ;|parte alta 70%
432         mov         r3,#0D1h       ;|
433         mov         r4,#01Fh       ;|parte baixa 30%
434
435         jmp         setenta_cinco_pc_40ms
436
437
438 ;-----
439 ; serial_dez_pc_80ms
440 ;-----
441 serial_dez_pc_80ms:
442         jnb         ri,$           ;aguarda receber cmd pela serial
443         mov         a,sbuf         ;salva o valor da serial no acc
444         clr         ri             ;limpa o flag da serial
445
446         CJNE        a,#6DH,cmp_serial_dez_pc_80ms
447         jmp         le             ;se teclar m volta pro menu
448 cmp_serial_dez_pc_80ms:
449         CJNE        a,#61H,dim_serial_dez_pc_80ms
450
451         mov         r0,#200        ; carrega tempo de PWM
452
453         mov         r1,#0D1h       ;|
454         mov         r2,#01Fh       ;|parte alta 15%
455         mov         r3,#07Bh       ;|
456         mov         r4,#02Fh       ;|parte baixa 85%
457
458         jmp         dez_pc_80ms
459 dim_serial_dez_pc_80ms:
460         mov         r0,#200        ; carrega tempo de PWM
461
462         mov         r1,#0F0h       ;|
463         mov         r2,#05Fh       ;|parte alta 5%
464         mov         r3,#06Bh       ;|
465         mov         r4,#08Fh       ;|parte baixa 95%
466
467         jmp         dez_pc_80ms
468
469
470 ;-----

```

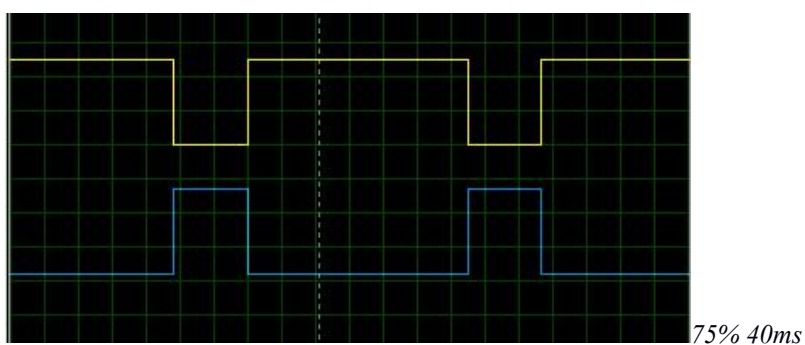
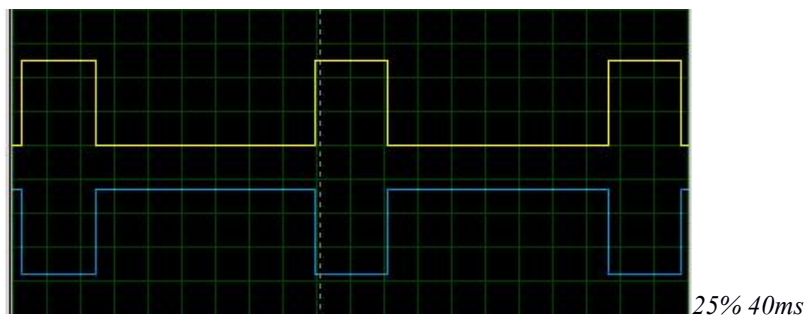
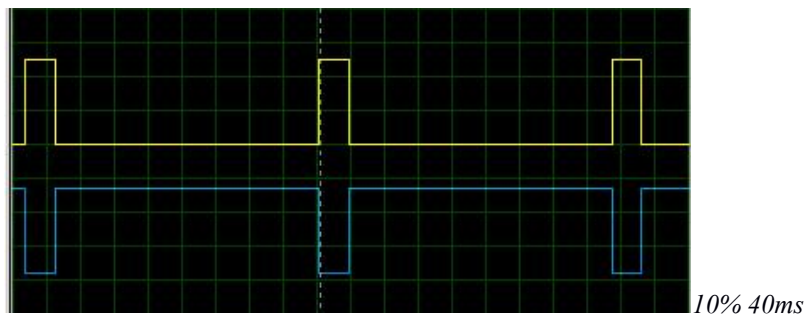


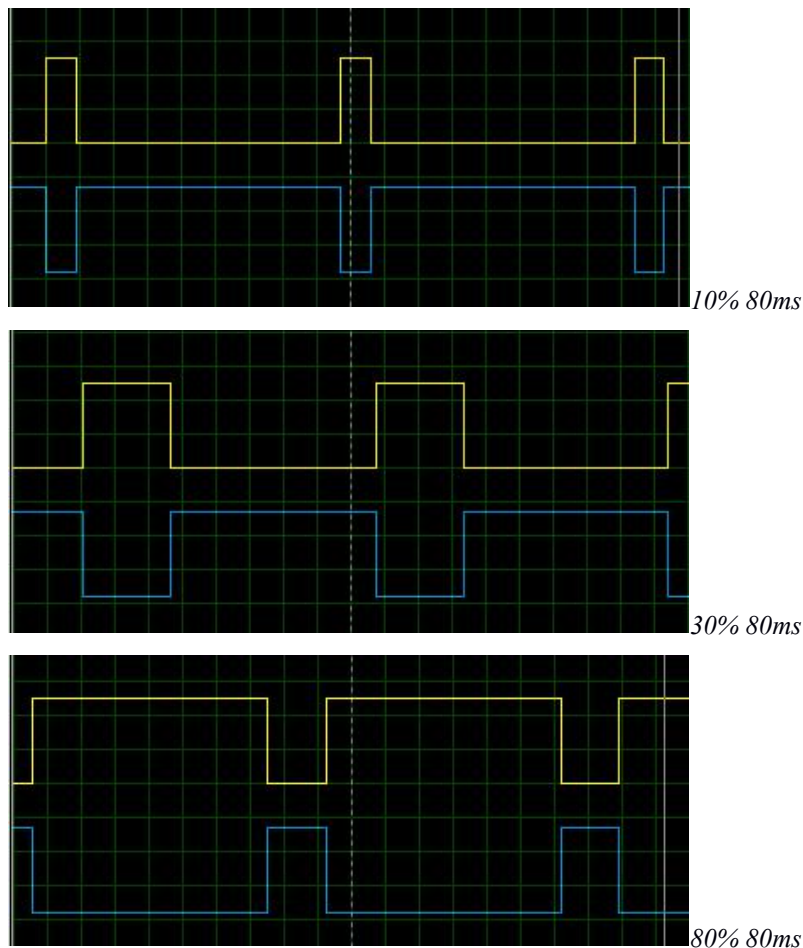
```

471 ; serial_trinta_pc_80ms
472 -----
473 serial_trinta_pc_80ms:
474     jnb     ri,$           ;aguarda receber cmd pela serial
475     mov     a,sbuf        ;salva o valor da serial no acc
476     clr     ri            ;limpa o flag da serial
477
478     CJNE    a,#0DH,cmp_serial_trinta_pc_80ms
479     jmp     ls            ;se teclar m volta pro menu
480 cmp_serial_trinta_pc_80ms:
481     CJNE    a,#61H,dim_serial_trinta_pc_80ms
482
483     mov     r0,#200       ; carrega tempo de PWM
484
485     mov     r1,#092h      ;|
486     mov     r2,#09Fh      ;|parte alta 35%
487     mov     r3,#034h      ;|
488     mov     r4,#0DFh      ;|parte baixa 65%
489
490     jmp     trinta_pc_80ms
491 dim_serial_trinta_pc_80ms:
492     mov     r0,#200       ; carrega tempo de PWM
493
494     mov     r1,#081h      ;|
495     mov     r2,#0DFh      ;|parte alta 25%
496     mov     r3,#015h      ;|
497     mov     r4,#09Fh      ;|parte baixa 75%
498
499     jmp     trinta_pc_80ms
500
501 -----
502 ; serial_oitenta_pc_80ms
503 -----
504 serial_oitenta_pc_80ms:
505     jnb     ri,$           ;aguarda receber cmd pela serial
506     mov     a,sbuf        ;salva o valor da serial no acc
507     clr     ri            ;limpa o flag da serial
508
509     CJNE    a,#0DH,cmp_serial_oitenta_pc_80ms
510     jmp     ls            ;se teclar m volta pro menu
511 cmp_serial_oitenta_pc_80ms:
512     CJNE    a,#61H,dim_serial_oitenta_pc_80ms
513
514     mov     r0,#200       ; carrega tempo de PWM
515
516     ;mov     r1,#092h      ;|
517     ;mov     r2,#09Fh      ;|parte alta 85%
518     ;mov     r3,#034h      ;|
519     ;mov     r4,#0DFh      ;|parte baixa 15%
520
521     jmp     oitenta_pc_80ms
522 dim_serial_oitenta_pc_80ms:
523     mov     r0,#200       ; carrega tempo de PWM
524
525     mov     r1,#015h      ;|
526     mov     r2,#09Fh      ;|parte alta 75%
527     mov     r3,#081h      ;|
528     mov     r4,#0DFh      ;|parte baixa 25%
529
530     jmp     oitenta_pc_80ms
531
532 -----
533 ;fim sub-rotina da serial
534
535

```

Formas de onda vistas pelo osciloscópio





OBS.: O projeto, contendo programa em assembly e hex, além de simulação no proteus, se encontra no repositório do GitHub e pode ser acessado através do endereço <https://github.com/thszk/microcontroller/tree/master/PWM>

Conclusão

O projeto abordou conceitos fundamentais da disciplina além de reforçar os conceitos vistos nas aulas teóricas e práticas, principalmente o desenvolvimento de firmwares para controle do projeto. O desenvolvimento do mesmo buscou atender a todos os requisitos apresentados na proposta inicial, enfrentando algumas dificuldades com relação à programação do micro, no entanto ao longo do período foram sendo sanadas e possibilitou a conclusão do projeto.