



MASTERS BRASIL 2003



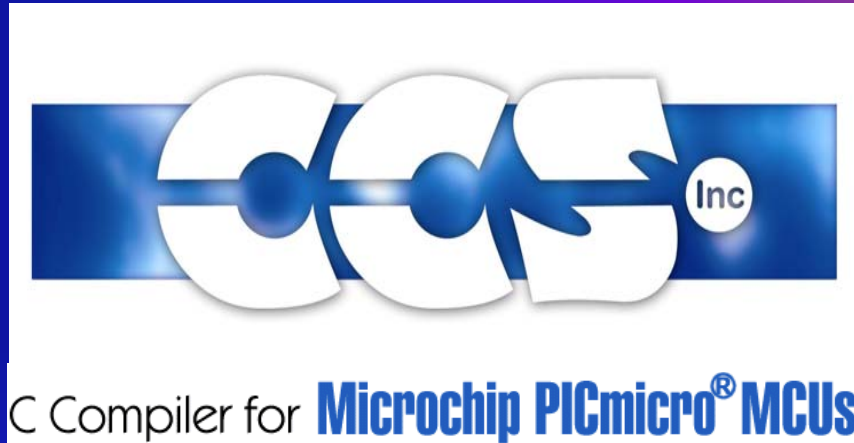
Introdução a Linguagem C para PICmicro[®] Programação de PIC em C CCS





Introdução a Linguagem C para PICmicro[®]

Programação de PIC em C CCS





Agenda

- ✦ Praticidade de Programação em C
- ✦ Uso do compilador no MPLAB e editor da CCS
- ✦ Comandos da Linguagem C
- ✦ Funções do Compilador padrão ANSI
- ✦ As funções especiais para o microcontrolador PIC
- ✦ A migração do programa desenvolvido em C para outro modelo de PIC
- ✦ Biblioteca de funções da CCS
- ✦ Simulação e Depuração de programas



Agenda

- ✦ **Praticidade de Programação em C**
- ✦ Uso do compilador no MPLAB e editor da CCS
- ✦ Comandos da Linguagem C
- ✦ Funções do Compilador padrão ANSI
- ✦ As funções especiais para o microcontrolador PIC
- ✦ A migração do programa desenvolvido em C para outro modelo de PIC
- ✦ Biblioteca de funções da CCS
- ✦ Simulação e Depuração de programas



Praticidade de Programação em C

Programação estruturada

Programação com o uso de Bibliotecas

Possibilidade de inserir código em Assembly

Implementação de rotinas de delay (atraso)

Criação de expressões matemáticas com variáveis de 8,16,32 bits e ponto flutuante

Criação de rotinas de interrupção

Seleção de Banco de RAM

Seleção de Página de memória de Programa

Configuração dos Periféricos



Exemplo de estrutura de programa em C

```
MPLAB IDE v6.22 - [C:\My Documents\Seminário C CCS\modelo.c]
File Edit View Project Debugger Programmer Tools Configure Window Help

1  /* ****
2  *                               *
3  *      Programação em C - Modelo de Estrutura      *
4  *                               *
5  *      CENTRO DE TREINAMENTO - MOSAICO ENGENHARIA  *
6  *                               *
7  *      TEL: (0XX11) 4992-8775          SITE: www.mosaico-eng.com.br  *
8  *                               *      E-MAIL: mosaico@mosaico-eng.com.br  *
9  * **** */
10 *
11 *      VERSÃO : 1.0
12 *      DATA : 05/06/2003
13 * **** */
14 /* ****
15 *                               *
16 *      Descrição geral
17 *                               *
18 * **** */
19 *
20 *      DEFINIÇÃO DAS VARIÁVEIS INTERNAS DO PIC
21 * **** */
22 #include <16f877A.h> // microcontrolador utilizado
23
24 *
25 *      Configurações para gravação
26 * **** */
27 #fuses xt,wdt,noprotect,put,brownout,nolvp,nocpd,nowrt // configuração dos fusíveis
28
29 *
30 *      Configura Conversor AD para 10 bits
31 * **** */
32 #device adc=10

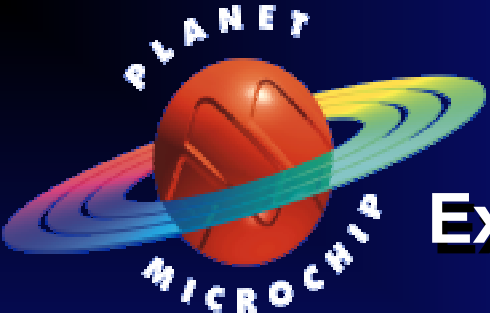
PIC16F877A 0x0fcf Ln 1, Col 1 INS
Start Microsoft PowerPoint - [Ap... MPLAB IDE v6.22 - [...]
```



Exemplo de estrutura de programa em C

```
MPLAB IDE v6.22 - [C:\My Documents\Seminário C CCS\modelo.c]
File Edit View Project Debugger Programmer Tools Configure Window Help

33 /* ****
34 *                               Carrega biblioteca Matemática          *
35 *                               **** */
36 #include <math.h>                // microcontrolador utilizado
37
38 /* ****
39 *                               Definições para uso de Rotinas de Delay *
40 *                               **** */
41 #use delay(clock=4000000, RESTART_WDT)
42
43 /* ****
44 *                               Inicializa USART                       *
45 *                               **** */
46 #use rs232(baud=9600,xmit=pin_c6,rcv=pin_c7)
47
48 /* ****
49 *                               Inicializa Master I2C                   *
50 *                               **** */
51 #use i2c(master,sda=pin_c4, scl=pin_c3, SLOW, RESTART_WDT, FORCE_HW)
52
53 /* ****
54 *                               Inicializa EEPROM                       *
55 *                               **** */
56 #from 0x2100 = { }              // carrega eeprom com valor inicial
57
58 /* ****
59 *                               Constantes internas                     *
60 *                               **** */
61 //A definição de constantes facilita a programação e a manutenção.
62
63 /* ****
64 *                               Definição e inicialização das variáveis *
65 *                               **** */
```



Exemplo de estrutura de programa em C

```
MPLAB IDE v6.22 - [C:\My Documents\Seminário C CCS\modelo.c]
File Edit View Project Debugger Programmer Tools Configure Window Help

63  /* ****
64  *          Definição e inicialização das variáveis          *
65  * **** */
66  //Neste bloco serão definidas as variáveis globais do programa.
67
68  /* ****
69  *          Definição e inicialização dos port's          *
70  * **** */
71  #use fast_io(a)
72  #use standard_io(b)
73  #use fixed_io(c)
74
75  /* ****
76  *          Declaração dos flags de software          *
77  * **** */
78  //A definição de flags ajuda na programação e economiza memória RAM.
79
80  /* ****
81  *          ENTRADAS          *
82  * **** */
83  // As entradas devem ser associadas a nomes para facilitar a programação e
84  // futuras alterações do hardware.
85
86  /* ****
87  *          SAÍDAS          *
88  * **** */
89  // As saídas devem ser associadas a nomes para facilitar a programação e
90  // futuras alterações do hardware.
91
92  /* ****
93  *          Tabela de Valores          *
94  * **** */

PIC16F877A 0x0fc Ln 93, Col 5 INS
Start MPLAB IDE v6.22 - [... Corel PHOTO-PAINT 11 - ... 2:59 AM
```



Exemplo de estrutura de programa em C

```
MPLAB IDE v6.22 - [C:\My Documents\Seminário C CCS\modelo.c]
File Edit View Project Debugger Programmer Tools Configure Window Help

92  /* *****
93  *                               Tabela de Valores
94  *                               ***** */
95
96  /* *****
97  *                               Configuração do Microcontrolador
98  *                               ***** */
99  // A função principal.
100
101  void main()
102  {
103      // configura microcontrolador
104      setup_adc_ports (no_analogs);
105      setup_counters(rtcc_internal , rtcc_div_8);
106      setup_timer_2 (t2_div_by_16,per,1);
107
108      // configura os tris
109      set_tris_a(0b11011111);
110
111      // inicializa os ports
112      porta=(0b00000000);
113      portb=(0b00001111);
114      portc=(0b00000000);
115
116      // habilita a interrupção de TMR0
117      enable_interrupts (GLOBAL|INT_RTCC|INT_TMR2);
118
119      #priority timer0,timer2          // prioridade para a int de trm0
120
121  /* *****
122  *                               Rotina principal
123  *                               ***** */
```

PIC16F877A 0x0fcf Ln 122, Col 5 INS

Start MPLAB IDE v6.22 - [...] Corel PHOTO-PAINT 11 - ... 3:01 AM



Exemplo de estrutura de programa em C

```
MPLAB IDE v6.22 - [C:\My Documents\Seminário C CCS\modelo.c]
File Edit View Project Debugger Programmer Tools Configure Window Help

121  /* ****
122  *                               Rotina principal
123  * **** */
124  while(TRUE) // rotina principal
125  {
126  RESTART_WDT(); // incia o watch-dog timer
127  }
128
129
130  /* ****
131  *                               Rotina de Tratamento de interrupção de TMR0
132  * **** */
133  #int_rtcc
134  void trata_int_tar0()
135  {
136  // aqui você implementa a sua rotina de tratamento de interrupção
137  }
138
139  /* ****
140  *                               Rotina de Tratamento de interrupção de TMR2
141  * **** */
142  #int_timer2
143  void trata_int_tar2()
144  {
145  // aqui você implementa a sua rotina de tratamento de interrupção
146  }
147
148  |
```

PIC16F877A 0x0fcf Ln 148, Col 1 INS

Start MPLAB IDE v6.22 - [...] Corel PHOTO-PAINT 11 - ... 3:02 AM



Exemplo de código ASM em um programa em C

```
MPLAB IDE v6.22 - [C:\Treinamento c\codigos\C1_e03_1.c*]
File Edit View Project Debugger Programmer Tools Configure Window Help

160  /* ***** | *****
161  *
162  *          Tratamento do Botão 0
163  * ***** | *****
164
165  if (!botao_UP)                // testa botão 0
166  {                             // botão 0 pressionado
167      if (!(bt_UP_press))      // o botão já foi tratado ?
168      {                         // ainda não. tem que passar pelo filtro.
169
170          if (filtro0 !=0)     // acabou o filtro do botão 0 ?
171          {
172              filtro0--;       // não, então decrementa o filtro
173          }
174
175          else                 // acabou. Faz o tratamento do botão
176          {
177              bt_UP_press = 1;  // marca que o botão está pressionado
178              if (contador != max)
179              {
180                  contador++;   // incrementa o contador
181                  portd = tabela(contador); //
182              }
183          }
184      }
185  else
186  {                             // botão 0 liberado
187      filtro0 = t_filtro;      // inicia o filtro do botão 0
188      bt_UP_press = 0;         // marca que o botão foi liberado
189  }
190
```

PIC16F877A 0x0fcf Ln 160, Col 18 INS



Exemplo de código ASM em um programa em C

```
MPLAB IDE v6.22 - [C:\Treinamento c\codigos\C1_e03_1.c*]
File Edit View Project Debugger Programmer Tools Configure Window Help

103 // Rotina de conversão para display
104 int tabela(int indice){
105     #asm
106         call    ini_convert
107         movwf  _return_
108         goto   fim_convert
109     ini_convert:
110         MOVF   indice,w           // COLOCA CONTADOR EM W
111         ANDLW  0B00001111        // MASCARA VALOR DE CONTADOR
112         ADDWF  0x02,F
113         //PGFEDCBA
114         RETLW  0B00111111        // 0H - RETORNA SÍMBOLO CORRETO 0
115         RETLW  0B00000110        // 1H - RETORNA SÍMBOLO CORRETO 1
116         RETLW  0B01011011        // 2H - RETORNA SÍMBOLO CORRETO 2
117         RETLW  0B01001111        // 3H - RETORNA SÍMBOLO CORRETO 3
118         RETLW  0B01100110        // 4H - RETORNA SÍMBOLO CORRETO 4
119         RETLW  0B01101101        // 5H - RETORNA SÍMBOLO CORRETO 5
120         RETLW  0B01111101        // 6H - RETORNA SÍMBOLO CORRETO 6
121         RETLW  0B00000111        // 7H - RETORNA SÍMBOLO CORRETO 7
122         RETLW  0B01111111        // 8H - RETORNA SÍMBOLO CORRETO 8
123         RETLW  0B01101111        // 9H - RETORNA SÍMBOLO CORRETO 9
124         RETLW  0B01110111        // AH - RETORNA SÍMBOLO CORRETO A
125         RETLW  0B01111100        // BH - RETORNA SÍMBOLO CORRETO b
126         RETLW  0B00111001        // CH - RETORNA SÍMBOLO CORRETO C
127         RETLW  0B01011110        // DH - RETORNA SÍMBOLO CORRETO d
128         RETLW  0B01111001        // EH - RETORNA SÍMBOLO CORRETO E
129         RETLW  0B01110001        // FH - RETORNA SÍMBOLO CORRETO F
130     fim_convert:
131     #endasm
132 }
133
```

PIC16F877A 0x0fcf Ln 124, Col 1 INS



Agenda

- ✦ Praticidade de Programação em C
- ✦ **Uso do compilador no MPLAB e editor da CCS**
- ✦ Comandos da Linguagem C
- ✦ Funções do Compilador padrão ANSI
- ✦ As funções especiais para o microcontrolador PIC
- ✦ A migração do programa desenvolvido em C para outro modelo de PIC
- ✦ Biblioteca de funções da CCS
- ✦ Simulação e Depuração de programas



Uso do Compilador da CCS no MPLAB

The screenshot shows the MPLAB IDE v6.22 interface. The title bar reads "MPLAB IDE v6.22 - [C:\My Documents\Seminário C CCS\modelo.c]". The menu bar includes File, Edit, View, Project, Debugger, Programmer, Tools, Configure, Window, and Help. The Project menu is open, showing options like Project Wizard..., New..., Open..., Close, Set Active Project, Quickbuild (no .asm file) Alt+F10, Clean, Build Options..., Find in Project Files..., Save Project, Save Project As..., Add Files to Project..., Remove File From Project, Select Language Tool suite..., and Set Language Tool Locations... The main editor window displays C code for a PIC16F628A microcontroller. The code includes comments in Portuguese and defines various configurations for the device, such as fuses and ADC settings.

```
1  /* ****
2  *
3  *
4  *
5  *
6  *
7  * TEL:
8  *
9  *
10 *
11 *
12 *
13 *
14 /* ****
15 *
16 *
17 *
18 /* ****
19 *
20 *
21 #include <16f877A.h> // microcontrolador utilizado
22
23
24 *
25 *
26 #fuses xt,wdt,noprotect,put,brownout,nolvp,nocpd,nowrt // configuração dos fusíveis
27
28 *
29 *
30 *
31 #device adc=10
32
```

At the bottom of the IDE, the status bar shows "PIC16F628A", "0x19ff", "Ln 1, Col 1", and "INS". The Windows taskbar at the bottom shows the Start button, "MPLAB IDE v6.22 - [...]", "Corel PHOTO-PAINT 11 - [...]", and the system tray with the time "5:08 AM".



Uso do Compilador da CCS no MPLAB

The screenshot shows the MPLAB IDE v6.22 interface. A 'Project Wizard' dialog box is open, displaying a 'Welcome!' message and instructions to click 'Next'. The background of the dialog features the 'MICROCHIP MPLAB IDE' logo. Below the dialog, a code editor window shows a project configuration file with the following content:

```
1  
2  
3  
4  
5  
6  
7  
8  
9  
1  
1  
1  
1  
1  
1  
1  
1  
22  
23 /* *****  
24 *          Configurações para gravação          *  
25 * *****  
26 #fuses xt,wdt,noprotect,put,brownout,nolvp,nocpd,nowrt // configuração dos fusíveis  
27  
28 /* *****  
29 *          Configura Conversor AD para 10 bits    *  
30 * *****  
31 #device adc=10  
32
```

The status bar at the bottom of the IDE shows 'PIC16F628A', '0x19ff', 'Ln 1, Col 1', and 'INS'. The Windows taskbar at the very bottom includes the Start button, 'MPLAB IDE v6.22 - [...]', 'Corel PHOTO-PAINT 11 - [...]', and the system tray with the time '5:21 AM'.



Uso do Compilador da CCS no MPLAB

A screenshot of the MPLAB IDE v6.22 interface. The "Project Wizard" dialog box is open, showing "Step One: Select a device" with a dropdown menu set to "PIC16F877A". The background shows a source code editor with C code for PIC16F628A, including comments in Portuguese and preprocessor directives like #fuses and #device.

MPLAB IDE v6.22 - IC:\My Documents\Seminaro C.CCS\modelo.c

Project Wizard

Step One:
Select a device

Device:
PIC16F877A

```
1  
2  
3  
4  
5  
6  
7  
8  
9  
10  
11  
12  
13  
14  
15  
16  
17  
18  
19  
20  
21  
22  
23  
24  
25  
26  
27  
28  
29  
30  
31  
32  
33  
34  
35  
36  
37  
38  
39  
40  
41  
42  
43  
44  
45  
46  
47  
48  
49  
50  
51  
52  
53  
54  
55  
56  
57  
58  
59  
60  
61  
62  
63  
64  
65  
66  
67  
68  
69  
70  
71  
72  
73  
74  
75  
76  
77  
78  
79  
80  
81  
82  
83  
84  
85  
86  
87  
88  
89  
90  
91  
92  
93  
94  
95  
96  
97  
98  
99  
100  
101  
102  
103  
104  
105  
106  
107  
108  
109  
110  
111  
112  
113  
114  
115  
116  
117  
118  
119  
120  
121  
122  
123  
124  
125  
126  
127  
128  
129  
130  
131  
132  
133  
134  
135  
136  
137  
138  
139  
140  
141  
142  
143  
144  
145  
146  
147  
148  
149  
150  
151  
152  
153  
154  
155  
156  
157  
158  
159  
160  
161  
162  
163  
164  
165  
166  
167  
168  
169  
170  
171  
172  
173  
174  
175  
176  
177  
178  
179  
180  
181  
182  
183  
184  
185  
186  
187  
188  
189  
190  
191  
192  
193  
194  
195  
196  
197  
198  
199  
200  
201  
202  
203  
204  
205  
206  
207  
208  
209  
210  
211  
212  
213  
214  
215  
216  
217  
218  
219  
220  
221  
222  
223  
224  
225  
226  
227  
228  
229  
230  
231  
232  
233  
234  
235  
236  
237  
238  
239  
240  
241  
242  
243  
244  
245  
246  
247  
248  
249  
250  
251  
252  
253  
254  
255  
256  
257  
258  
259  
260  
261  
262  
263  
264  
265  
266  
267  
268  
269  
270  
271  
272  
273  
274  
275  
276  
277  
278  
279  
280  
281  
282  
283  
284  
285  
286  
287  
288  
289  
290  
291  
292  
293  
294  
295  
296  
297  
298  
299  
300  
301  
302  
303  
304  
305  
306  
307  
308  
309  
310  
311  
312  
313  
314  
315  
316  
317  
318  
319  
320  
321  
322  
323  
324  
325  
326  
327  
328  
329  
330  
331  
332  
333  
334  
335  
336  
337  
338  
339  
340  
341  
342  
343  
344  
345  
346  
347  
348  
349  
350  
351  
352  
353  
354  
355  
356  
357  
358  
359  
360  
361  
362  
363  
364  
365  
366  
367  
368  
369  
370  
371  
372  
373  
374  
375  
376  
377  
378  
379  
380  
381  
382  
383  
384  
385  
386  
387  
388  
389  
390  
391  
392  
393  
394  
395  
396  
397  
398  
399  
400  
401  
402  
403  
404  
405  
406  
407  
408  
409  
410  
411  
412  
413  
414  
415  
416  
417  
418  
419  
420  
421  
422  
423  
424  
425  
426  
427  
428  
429  
430  
431  
432  
433  
434  
435  
436  
437  
438  
439  
440  
441  
442  
443  
444  
445  
446  
447  
448  
449  
450  
451  
452  
453  
454  
455  
456  
457  
458  
459  
460  
461  
462  
463  
464  
465  
466  
467  
468  
469  
470  
471  
472  
473  
474  
475  
476  
477  
478  
479  
480  
481  
482  
483  
484  
485  
486  
487  
488  
489  
490  
491  
492  
493  
494  
495  
496  
497  
498  
499  
500  
501  
502  
503  
504  
505  
506  
507  
508  
509  
510  
511  
512  
513  
514  
515  
516  
517  
518  
519  
520  
521  
522  
523  
524  
525  
526  
527  
528  
529  
530  
531  
532  
533  
534  
535  
536  
537  
538  
539  
540  
541  
542  
543  
544  
545  
546  
547  
548  
549  
550  
551  
552  
553  
554  
555  
556  
557  
558  
559  
560  
561  
562  
563  
564  
565  
566  
567  
568  
569  
570  
571  
572  
573  
574  
575  
576  
577  
578  
579  
580  
581  
582  
583  
584  
585  
586  
587  
588  
589  
590  
591  
592  
593  
594  
595  
596  
597  
598  
599  
600  
601  
602  
603  
604  
605  
606  
607  
608  
609  
610  
611  
612  
613  
614  
615  
616  
617  
618  
619  
620  
621  
622  
623  
624  
625  
626  
627  
628  
629  
630  
631  
632  
633  
634  
635  
636  
637  
638  
639  
640  
641  
642  
643  
644  
645  
646  
647  
648  
649  
650  
651  
652  
653  
654  
655  
656  
657  
658  
659  
660  
661  
662  
663  
664  
665  
666  
667  
668  
669  
670  
671  
672  
673  
674  
675  
676  
677  
678  
679  
680  
681  
682  
683  
684  
685  
686  
687  
688  
689  
690  
691  
692  
693  
694  
695  
696  
697  
698  
699  
700  
701  
702  
703  
704  
705  
706  
707  
708  
709  
710  
711  
712  
713  
714  
715  
716  
717  
718  
719  
720  
721  
722  
723  
724  
725  
726  
727  
728  
729  
730  
731  
732  
733  
734  
735  
736  
737  
738  
739  
740  
741  
742  
743  
744  
745  
746  
747  
748  
749  
750  
751  
752  
753  
754  
755  
756  
757  
758  
759  
760  
761  
762  
763  
764  
765  
766  
767  
768  
769  
770  
771  
772  
773  
774  
775  
776  
777  
778  
779  
780  
781  
782  
783  
784  
785  
786  
787  
788  
789  
790  
791  
792  
793  
794  
795  
796  
797  
798  
799  
800  
801  
802  
803  
804  
805  
806  
807  
808  
809  
810  
811  
812  
813  
814  
815  
816  
817  
818  
819  
820  
821  
822  
823  
824  
825  
826  
827  
828  
829  
830  
831  
832  
833  
834  
835  
836  
837  
838  
839  
840  
841  
842  
843  
844  
845  
846  
847  
848  
849  
850  
851  
852  
853  
854  
855  
856  
857  
858  
859  
860  
861  
862  
863  
864  
865  
866  
867  
868  
869  
870  
871  
872  
873  
874  
875  
876  
877  
878  
879  
880  
881  
882  
883  
884  
885  
886  
887  
888  
889  
890  
891  
892  
893  
894  
895  
896  
897  
898  
899  
900  
901  
902  
903  
904  
905  
906  
907  
908  
909  
910  
911  
912  
913  
914  
915  
916  
917  
918  
919  
920  
921  
922  
923  
924  
925  
926  
927  
928  
929  
930  
931  
932  
933  
934  
935  
936  
937  
938  
939  
940  
941  
942  
943  
944  
945  
946  
947  
948  
949  
950  
951  
952  
953  
954  
955  
956  
957  
958  
959  
960  
961  
962  
963  
964  
965  
966  
967  
968  
969  
970  
971  
972  
973  
974  
975  
976  
977  
978  
979  
980  
981  
982  
983  
984  
985  
986  
987  
988  
989  
990  
991  
992  
993  
994  
995  
996  
997  
998  
999  
1000
```

PIC16F628A 0x19ff Ln 1, Col 1 INS

Start MPLAB IDE v6.22 - [...] Corel PHOTO-PAINT 11 - [...] 7 5:27 AM



Uso do Compilador da CCS no MPLAB

The screenshot shows the MPLAB IDE v6.22 interface. The 'Project Wizard' dialog box is open, displaying 'Step Four: Add any existing files your project'. The file tree on the left shows a folder named 'Seminário C CCS' containing files like 'Apresentação', 'modelo.c', and several 'projeto-wizard' files. The 'Add >>' button is highlighted. To the right, a list box contains the path 'c:\My Documents\Seminário C CC'. Below the list box, there is a checkbox and the instruction 'Check the box to copy the file to the project directory'. At the bottom of the dialog are buttons for '< Back', 'Next >', 'Cancel', and 'Help'. The background shows a code editor with C code for PIC16F628A, including configuration for fuses and the ADC device.

```
/* *****  
 *                               *  
 *      Configurações para gravação *  
 *                               *  
 * *****/
```

```
#fuses xt,wdt,noprotect,put,brownout,nolvp,nocpd,nowrt // configuração dos fusíveis
```

```
/* *****  
 *                               *  
 *      Configura Conversor AD para 10 bits *  
 *                               *  
 * *****/
```

```
#device adc=10
```

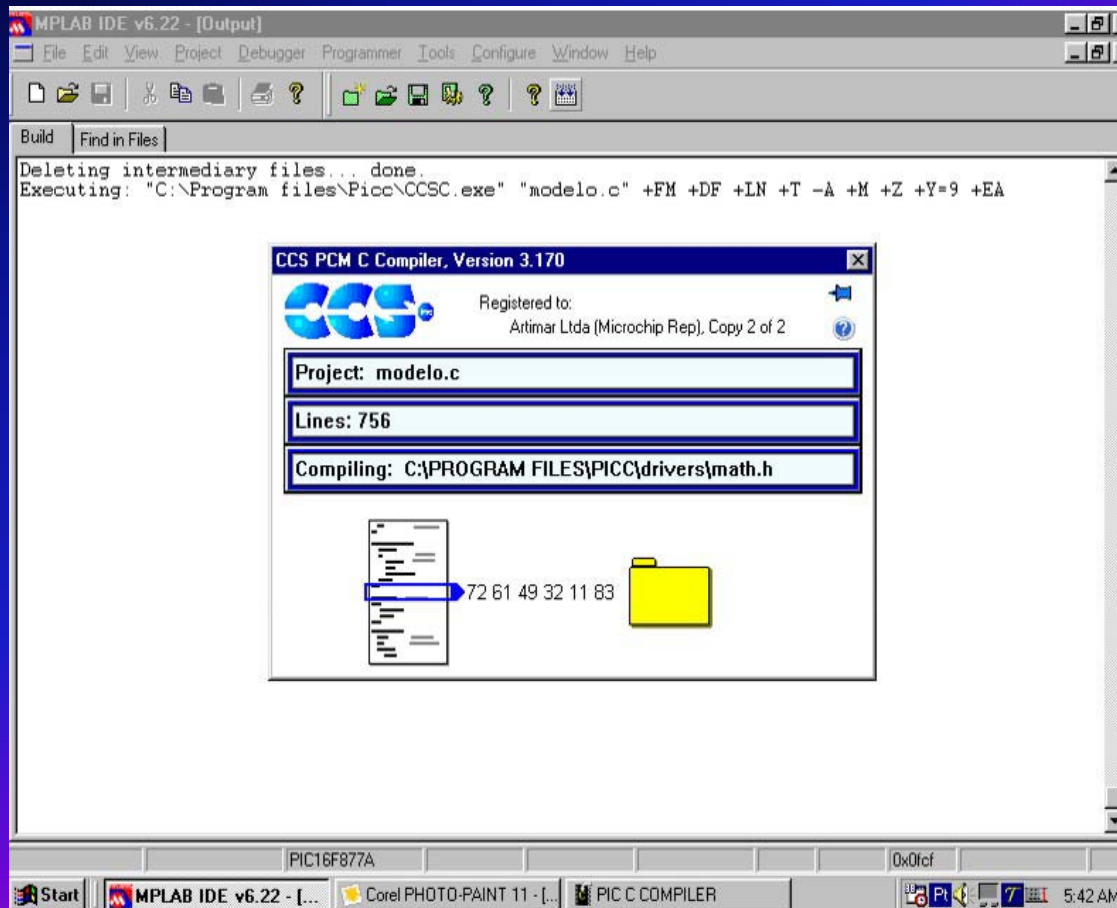
PIC16F628A 0x19ff Ln 1, Col 1 INS

Start MPLAB IDE v6.22 - [...]
Corel PHOTO-PAINT 11 - [...]

5:38 AM



Uso do Compilador da CCS no MPLAB





Uso do Compilador da CCS no MPLAB

MPLAB IDE v6.22 - [C:\ModuloCvc1_e01.c]

File Edit View Project Debugger Programmer Tools Configure Window Help

CCS PCM C Compiler, Version 3.170

Registered to:
Artimar Ltda (Microchip Rep), Copy 2 of 2

Project: c1_e01.c

Files: 2, Statements: 19, Time: 1 Sec, Lines: 412

Output files: ERR HEX SYM LST COF PJT TRE \$\$\$

0 Errors, 1 Warnings, Time: 1 Seconds

ROM: 1%

RAM: 3%

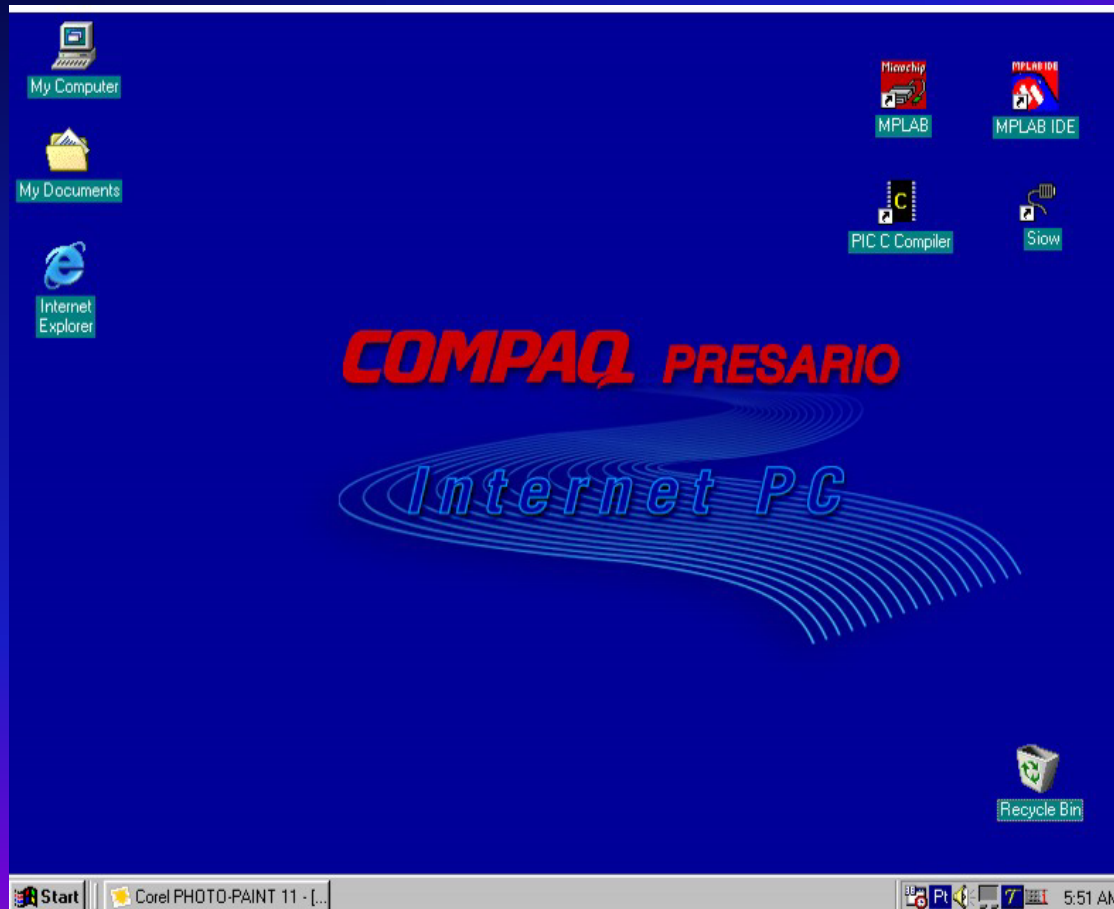
```
1  /* ****
2  *
3  *
4  *
5  *
6  *
7  * TEL:
8  *
9  * ****
10 *
11 * VERSE
12 *
13 * ****
14 /* ****
15 *
16 * ****
17 // Este s
18 // este p
19
20 /* ****
21 *
22 *      DEFINIÇÃO DAS VARIÁVEIS INTERNAS DO PIC
23 * *****/
24 #include    <16f877a.h>    // microcontrolador utilizado
25 #device icd=true
26
27 /* ****
28 *
29 *      Configurações para gravação
30 * *****/
31 #fuses xt,wdt,noprotect,put,brownout,nolvp,nocpd,nowrt // configuração dos fusíveis
32
```

PIC16F877A 0x01cf Ln 1, Col 77 INS

Start MPLAB IDE v6.22 - [C:\M... Corel PHOTO-PAINT 11 - [... PIC C COMPILER 5:46 AM

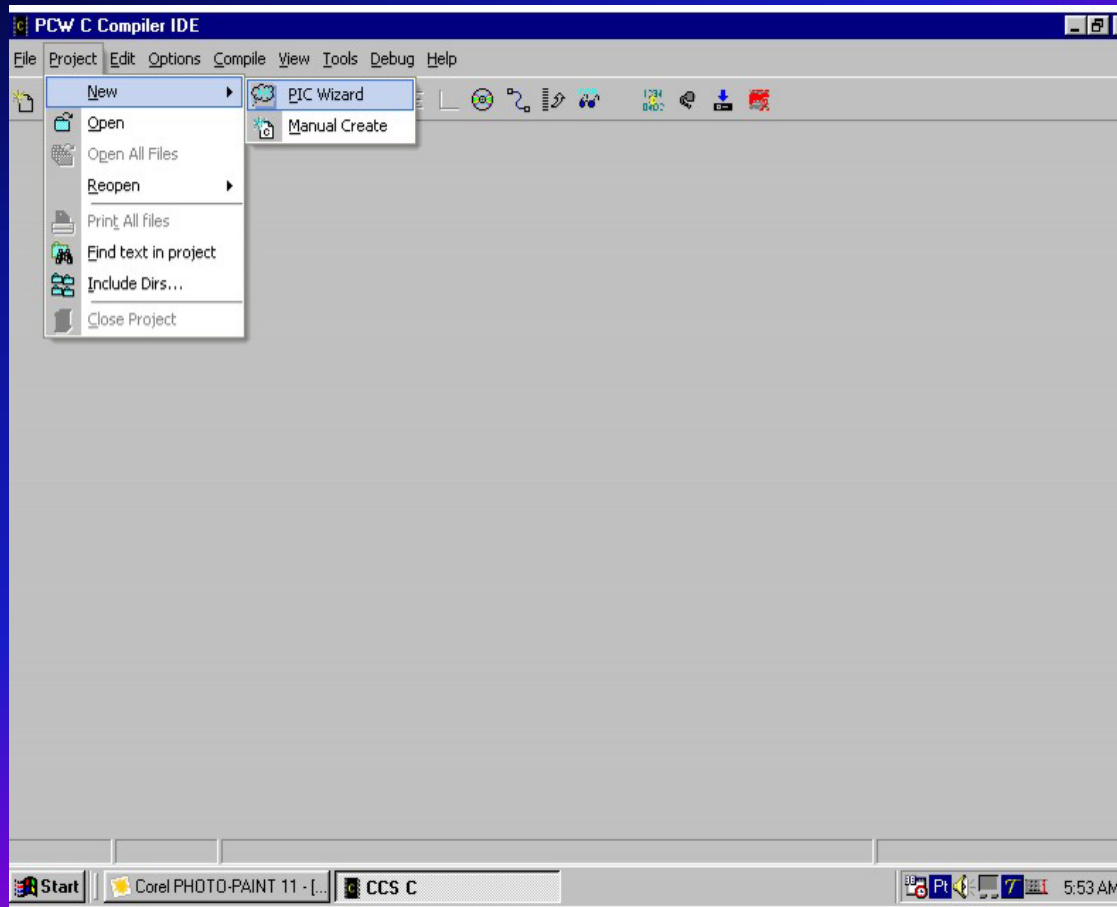


Uso do Compilador com o Editor da CCS



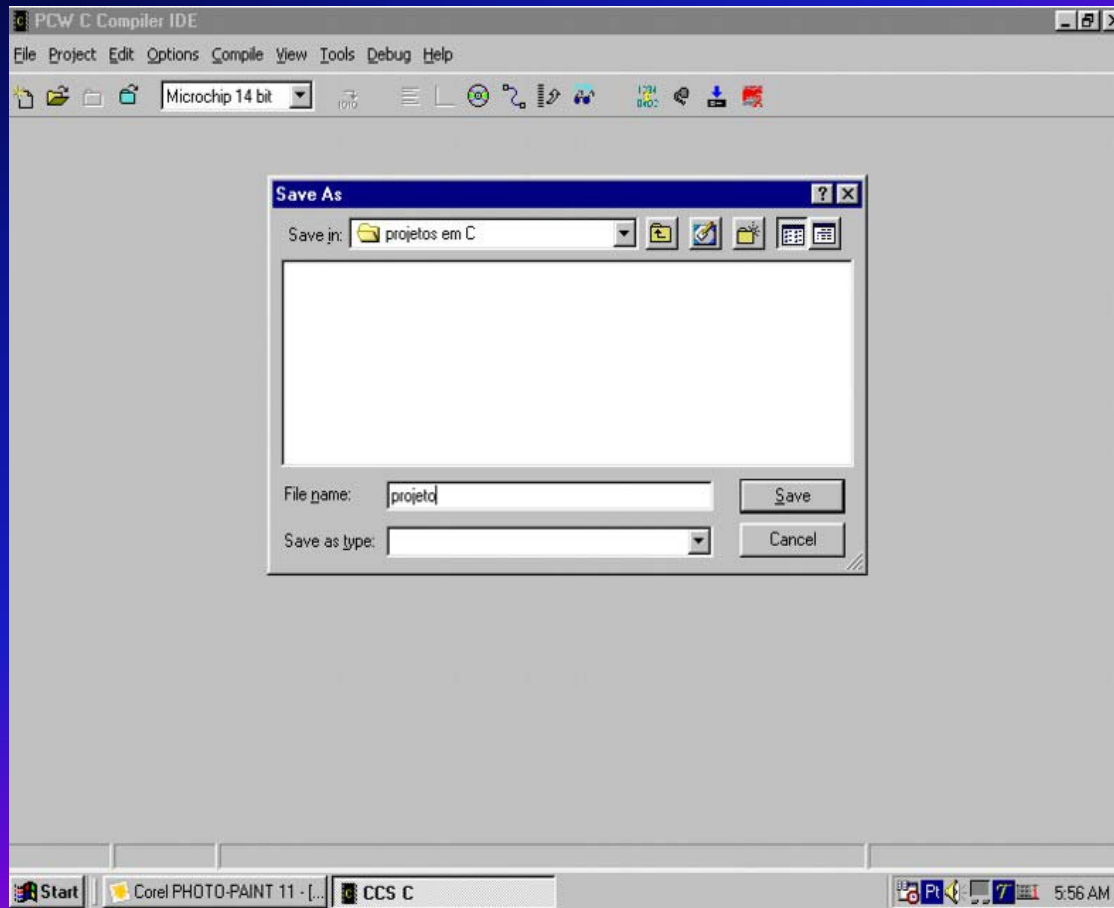


Uso do Compilador com o Editor da CCS



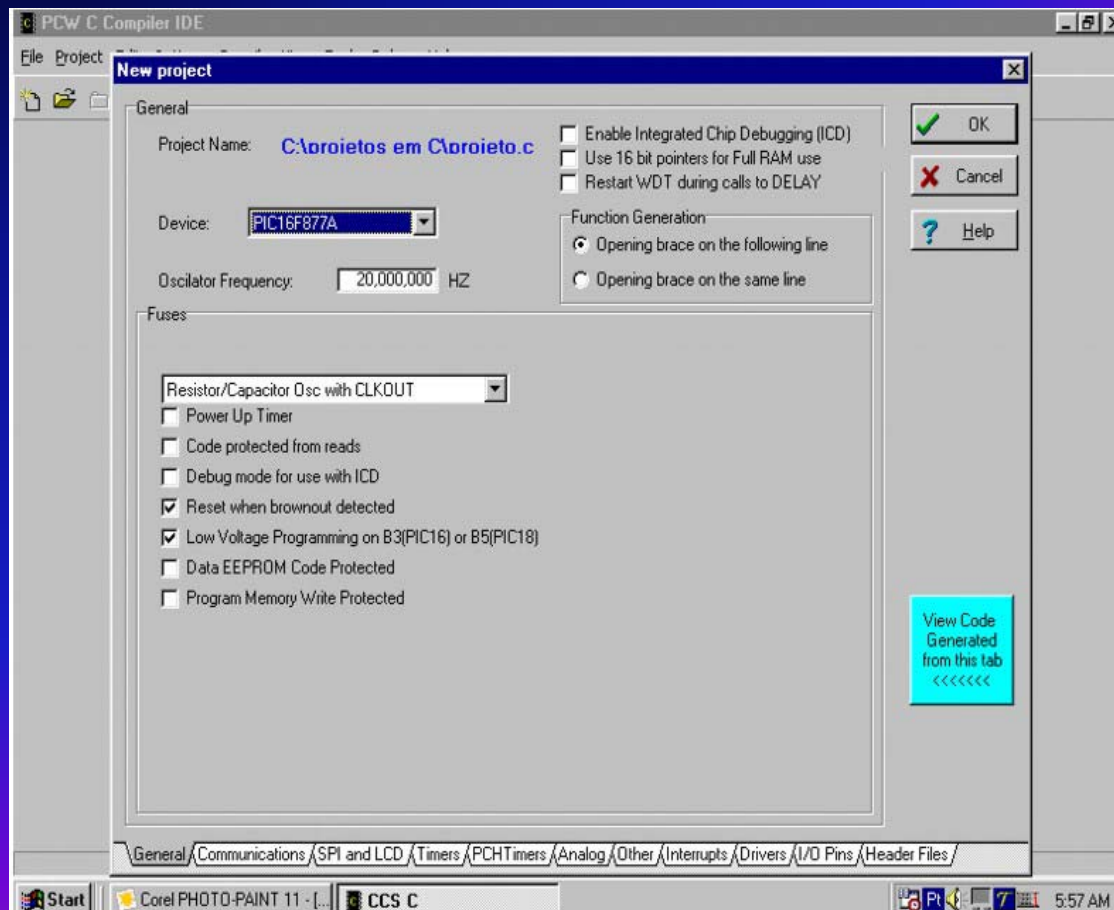


Uso do Compilador com o Editor da CCS



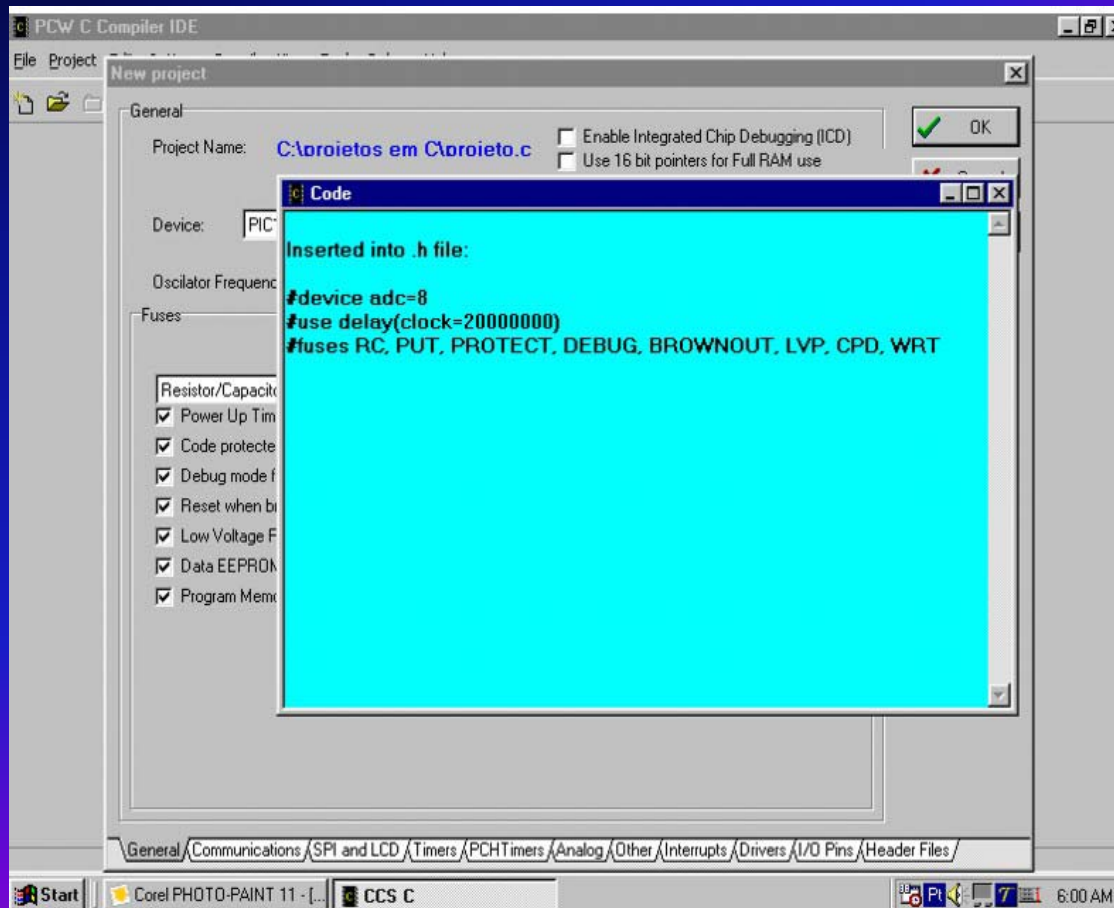


Uso do Compilador com o Editor da CCS



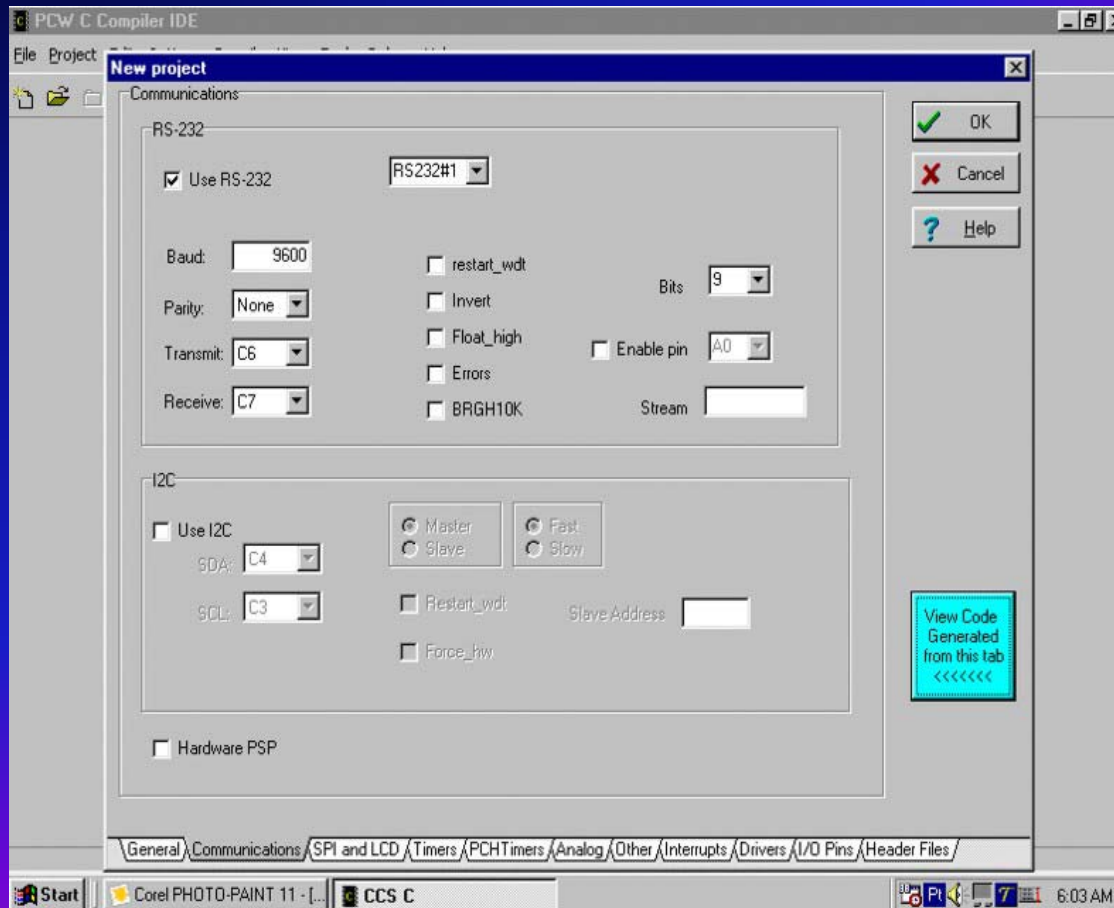


Uso do Compilador com o Editor da CCS



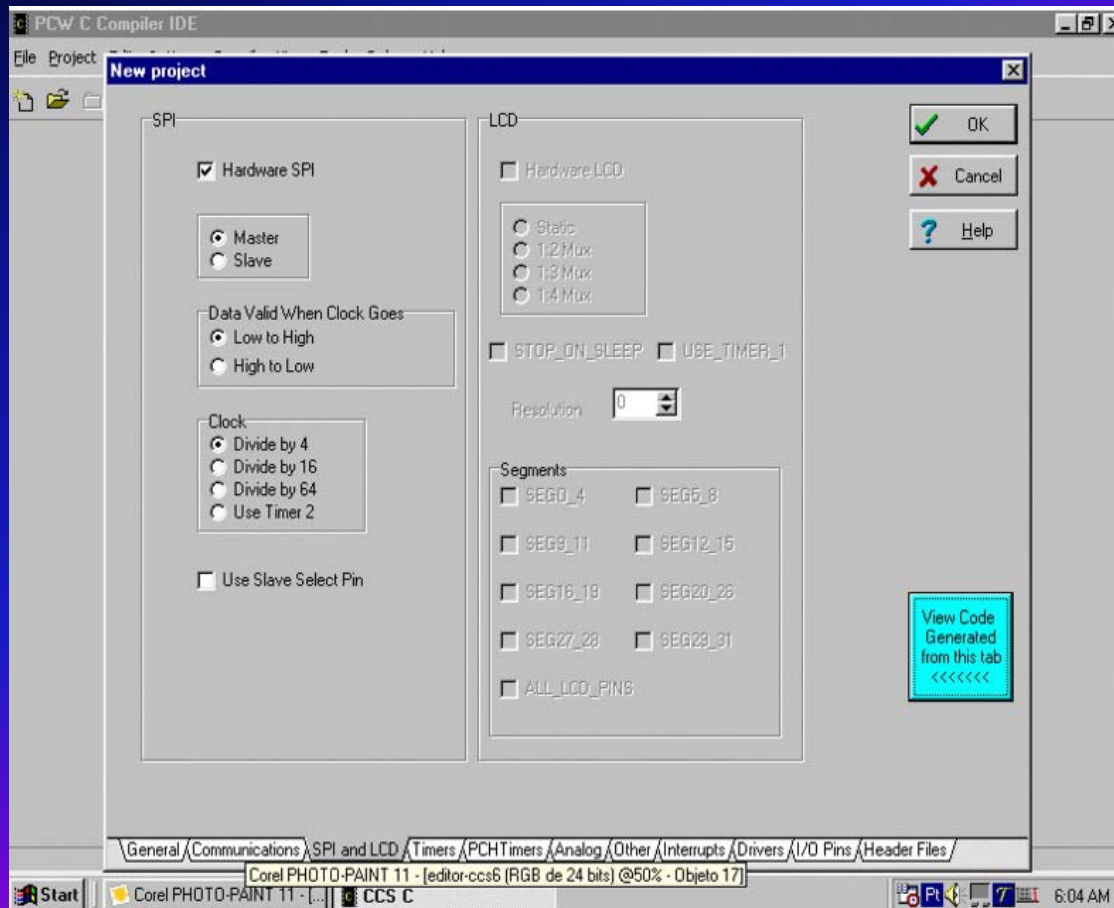


Uso do Compilador com o Editor da CCS



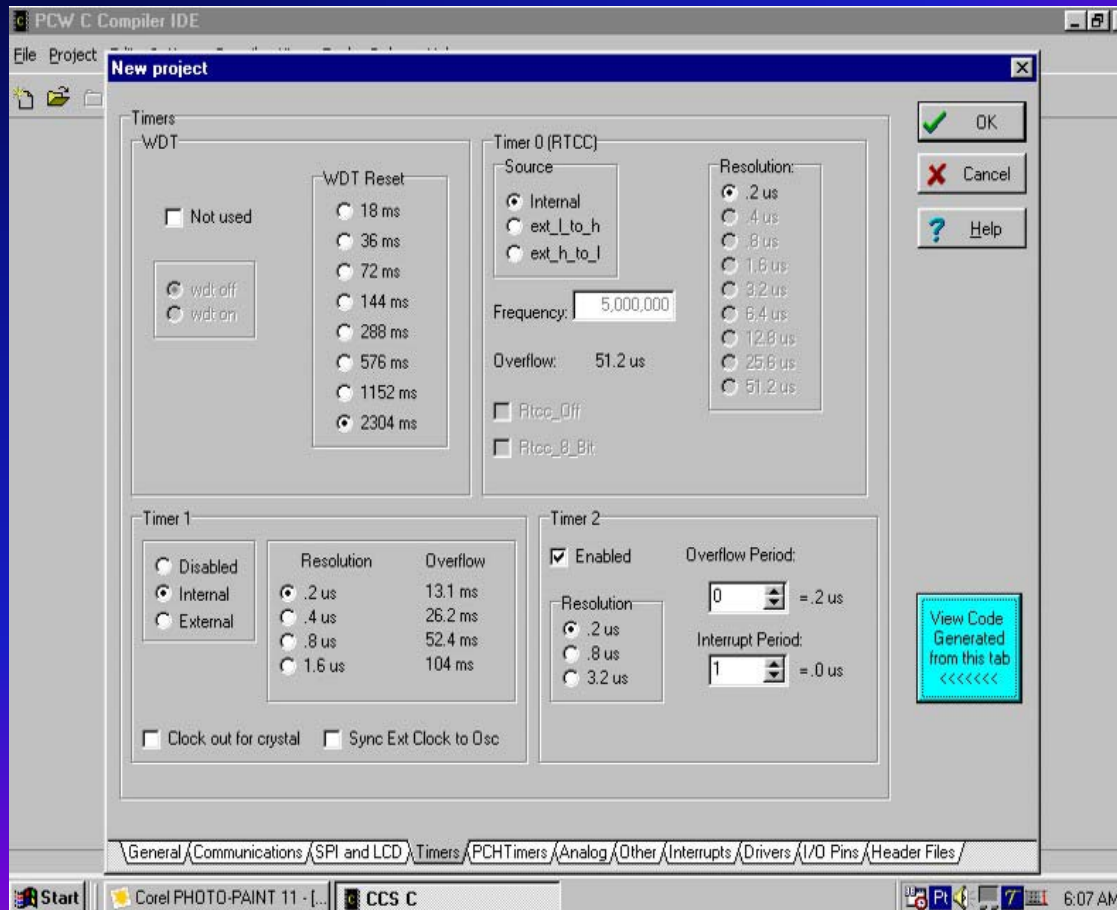


Uso do Compilador com o Editor da CCS





Uso do Compilador com o Editor da CCS





Uso do Compilador com o Editor da CCS

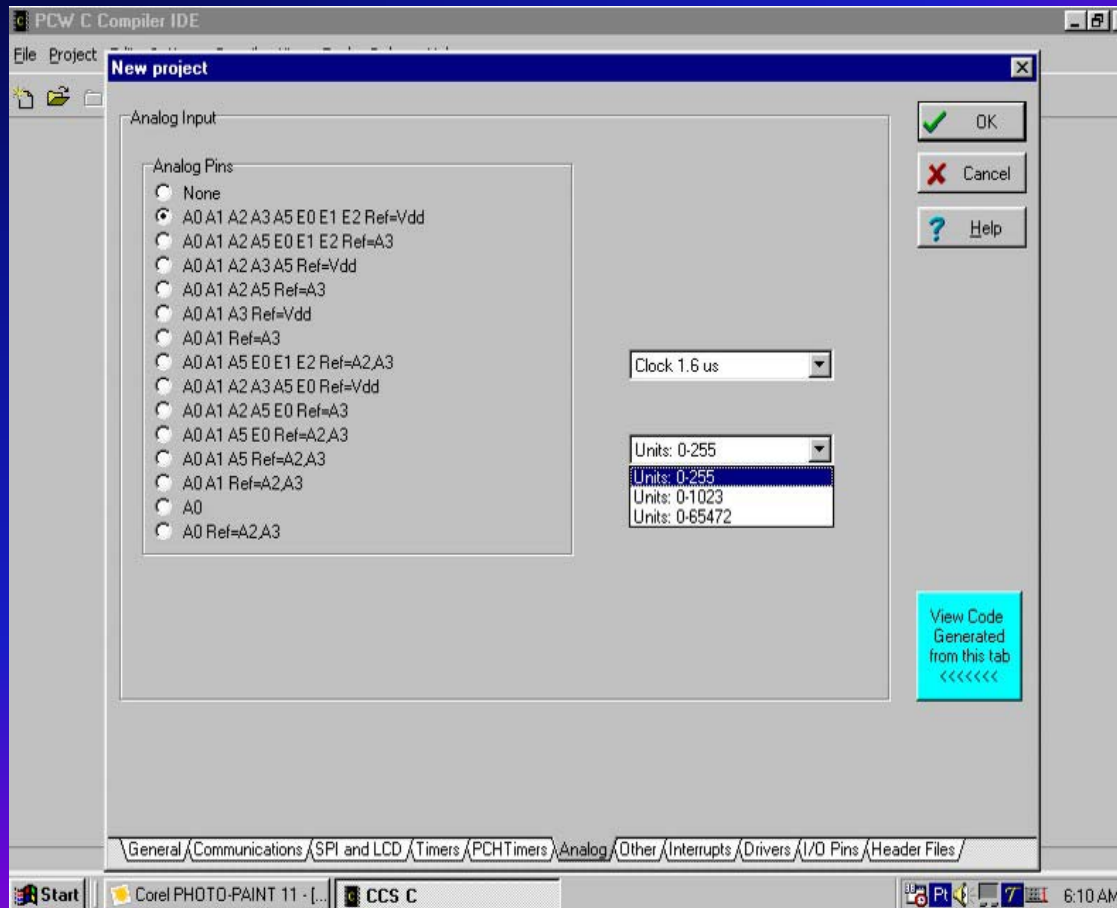
A screenshot of the CCS C Compiler IDE interface. The main window is titled "New project" and shows configuration options for "Timers". Under "Timer 0 (RTCC)", the "Resolution" is set to "100ns". Under "Timer 1", the "Source" is set to "Internal". A "Code" window is open in the foreground, displaying the following code:

```
Inserted into .c file in main():  
  
setup_counters(RTCC_INTERNAL_WDT_2304MS);  
setup_timer_1(T1_INTERNAL|T1_DIV_BY_1);  
setup_timer_2(T2_DIV_BY_1,0.1);
```

The IDE's taskbar at the bottom shows the Start button, open applications including "Corel PHOTO-PAINT 11" and "CCS C", and the system tray with the time "6:06 AM".



Uso do Compilador com o Editor da CCS





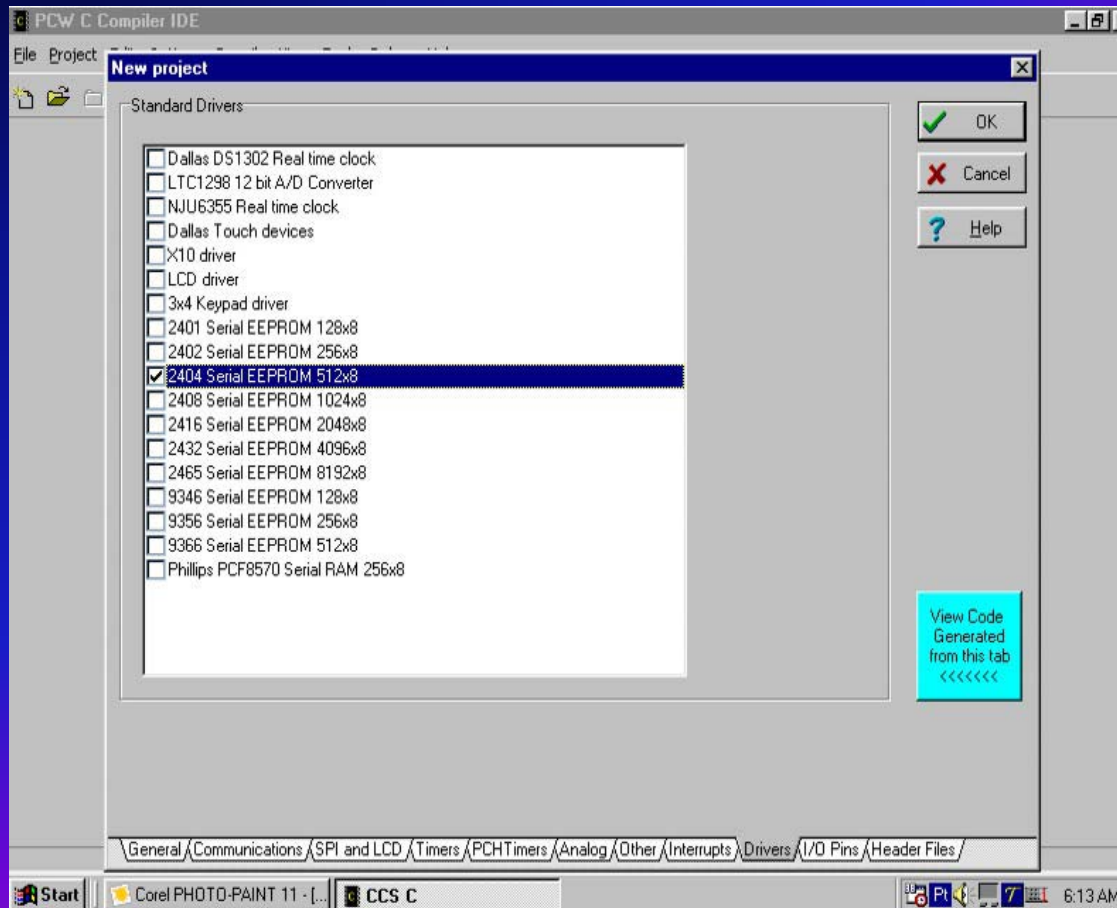
Uso do Compilador com o Editor da CCS

The screenshot shows the PCW C Compiler IDE interface. The 'New project' dialog box is open, with the 'Interrupts' tab selected. The 'Code' window is open, displaying the generated C code for the project. The code is as follows:

```
Inserted into .c file before main():  
  
#int_RTCC  
RTCC_isr()  
{  
  
}  
  
#int_RB  
RB_isr()  
{  
  
}  
  
Inserted into .c file in main():  
  
enable_interrupts(INT_RTCC);  
enable_interrupts(INT_RB);
```



Uso do Compilador com o Editor da CCS





Uso do Compilador com o Editor da CCS

PCW C Compiler IDE

File Project

New project

Pin names

Pins	I/O Type	Identifiers
A0	Not used	PIN_A0.[Comp1+].[ADC]
A1	Not used	PIN_A1.[Comp2+].[ADC]
A2	Not used	PIN_A2.[Comp2-].[ADC]
A3	Not used	PIN_A3.[Comp1-].[ADC]
A4	Input	PIN_A4
A5	Not used	PIN_A5.[ADC]
B0	Input	PIN_B0
B1	Input	PIN_B1
B2	Input	PIN_B2
B3	Input	PIN_B3
B4	Input	PIN_B4
B5	Input	PIN_B5
B6	Input/Output	PIN_B6
B7	Input/Output	PIN_B7
C0	Output	PIN_C0
C1	Output	PIN_C1

Enable Pullups on port B

List multiple identifiers by separating with commas

OK Cancel Help

View Code Generated from this tab <<<<<<<

General / Communications / SPI and LCD / Timers / PCHTimers / Analog / Other / Interrupts / Drivers / I/O Pins / Header Files

Start Corel PHOTO-PAINT 11 - [CCS C 6:15 AM



Uso do Compilador com o Editor da CCS

```
PCW C Compiler IDE
File Project Edit Options Compile View Tools Debug Help
Microchip 14 bit
projeto.c
1 #include "C:\projetos em C\projeto.h"
2 #include <2404.C>
3 #include <math.h>
4 #ZERO_RAM
5 #int_RTCC
6 RTCC_isr()
7 {
8
9 }
10
11 #int_RB
12 RB_isr()
13 {
14
15 }
16
17 void main()
18 {
19     setup_adc_ports(ALL_ANALOG);
20     setup_adc(ADC_CLOCK_DIV_8);
21     setup_psp(PSP_DISABLED);
22     setup_spi(SPI_MASTER|SPI_I_TO_H|SPI_CLK_DIV_4);
23     setup_counters(RTCC_INTERNAL, WDT_2304MS);
24     setup_timer_1(T1_INTERNAL|T1_DIV_BY_1);
25     setup_timer_2(T2_DIV_BY_1, 0, 1);
26     setup_comparator(NC_NC_NC_NC);
27     setup_vref(FALSE);
28     init_ext_eeprom();
29     enable_interrupts(INT_RTCC);
30     enable_interrupts(INT_RB);
31     enable_interrupts(global);
32 }
33
15:2 c:\projetos em c\projeto.c projeto
Start Corel PHOTO-PAINT 11 - [...] CCS C 6:19 AM
```



Agenda

- ✦ Praticidade de Programação em C
- ✦ Uso do compilador no MPLAB e editor da CCS
- ✦ **Comandos da Linguagem C**
- ✦ Funções do Compilador padrão ANSI
- ✦ As funções especiais para o microcontrolador PIC
- ✦ A migração do programa desenvolvido em C para outro modelo de PIC
- ✦ Biblioteca de funções da CCS
- ✦ Simulação e Depuração de programas



Comandos da Linguagem C

Comando:	Exemplo
<pre>if (expr) {bloco de comandos;} else {bloco de comandos;}</pre>	<pre>if (x==25) x=1; else x=x+1;</pre>
<pre>switch (expr) { case cexpr: stmt; default:stmt; }</pre>	<pre>switch (cmd) { case 0:printf("cmd 0"); break; case 1:printf("cmd 1"); break; default:printf("badcmd");}</pre>



Comandos da Linguagem C

<pre>while (expr) stmt;</pre>	<pre>While{ (get_rtcc()!=0) putc('n'); }</pre>
<pre>Do{ stmt }while (expr);</pre>	<pre>do { putc(c=getc()); } while (c!=0);</pre>
<pre>for(exp1;exp2;exp3) stmt;</pre>	<pre>for (i=1;i<=10;++i){ printf("%u\r\n",i); }</pre>
<pre>goto label;</pre>	<pre>#byte portb = 0x06 loop: portb = portb ^ 0x01 goto loop;</pre>



Agenda

- ✦ Praticidade de Programação em C
- ✦ Uso do compilador no MPLAB e editor da CCS
- ✦ Comandos da Linguagem C
- ✦ **Funções do Compilador padrão ANSI**
- ✦ As funções especiais para o microcontrolador PIC
- ✦ A migração do programa desenvolvido em C para outro modelo de PIC
- ✦ Biblioteca de funções da CCS
- ✦ Simulação e Depuração de programas



Funções do Compilador padrão ANSI

Funções Matemáticas

- ❖ `abs()` → Retorna o valor absoluto de um sinal.
- ❖ `exp()` → Retorna o valor de e^x ($e = 2,718282$).
- ❖ `floor()` → Retorna o menor número inteiro que é menor que o valor especificado.
- ❖ `ceil()` → Retorna o menor número inteiro que é maior que o valor especificado.
- ❖ `labs()` → Retorna o valor absoluto de um número inteiro.
- ❖ `modf()` → Retorna a parte inteira e a parte fracionária de um número fracionário.
- ❖ `fmod()` → Retorna o resto da divisão de dois números fracionários.



Funções do Compilador padrão ANSI

Funções Matemáticas

- ❖ `fabs()` → Retorna o valor absoluto de um número fracionário (ponto flutuante).
- ❖ `log()` → Calcula o Ln (logaritmo natural) do número especificado.
- ❖ `log10()` → Calcula o valor do logaritmo de base 10 do número especificado.
- ❖ `frexp()` → Decompõe o número em uma mantissa na faixa de 0.5 a 1 e um expoente inteiro tal que o número será igual à mantissa * 2^{exp} .
- ❖ `ldexp()` → Multiplica um valor por 2^{exp} .
- ❖ `sqrt()` → Retorna a Raiz quadrada do valor especificado.



Funções do Compilador padrão ANSI

Funções Matemáticas

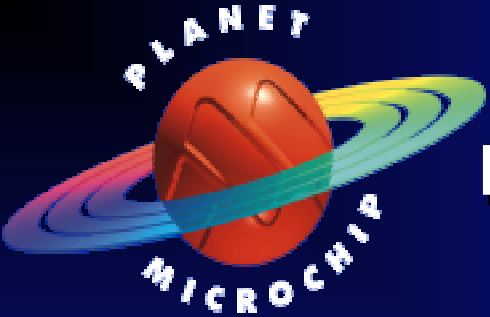
- ❖ $\cos()$ → Retorna o valor do cosseno.
- ❖ $\arccos()$ → Retorna o valor do arco cosseno.
- ❖ $\cosh()$ → Retorna o valor do cosseno hiperbólico.
- ❖ $\sin()$ → Retorna o valor do seno.
- ❖ $\arcsin()$ → Retorna o valor do arco seno.
- ❖ $\sinh()$ → Retorna o valor do seno hiperbólico.
- ❖ $\tan()$ → Retorna o valor da tangente.
- ❖ $\tanh()$ → Retorna o valor da tangente hiperbólica
- ❖ $\arctan()$ → Retorna o arco tangente.



Funções do Compilador padrão ANSI

Funções de Manipulação de Caracteres

- ❖ `atol()`, `atol32()`, `atof()` → Converte uma string em um valor inteiro de 8, 16 ou 32 bits, ou em um float.
- ❖ `tolower()` → Converte um caractere no seu correspondente minúsculo.
- ❖ `toupper()` → Converte um caractere no seu correspondente maiúsculo.
- ❖ `isalnum()` → Verifica se o caractere é uma letra ou número.
- ❖ `isalpha()` → Verifica se o caractere é uma letra.
- ❖ `isdigit()` → Verifica se o caractere é um número.
- ❖ `islower()` → Verifica se o caractere é uma letra minúscula.
- ❖ `isupper()` → Verifica se o caractere é uma letra maiúscula.



Funções do Compilador padrão ANSI

Funções de Manipulação de Caracteres

- ❖ `isspace()` → Verifica se o caractere é um espaço.
- ❖ `isxdigit()` → Verifica se o caractere é um dígito hexadecimal.
- ❖ `iscntrl()` → Verifica se é um caractere de controle.
- ❖ `isgraph()` → Verifica se é um caractere gráfico.
- ❖ `isprint()` → Verifica se é um caractere que pode ser impresso.
- ❖ `ispunct()` → Verifica se o caractere é diferente de letra ou número.
- ❖ `ismoung()` → Verifica se o caractere está contido em uma string.
- ❖ `strlen()` → Retorna a quantidade de caracteres em uma string.



Funções do Compilador padrão ANSI

Funções de Manipulação de Caracteres

- ❖ `strcpy()` → Copia o conteúdo de uma string para outra.
- ❖ `strncpy()` → Copia um determinado número de caracteres de uma string para outra.
- ❖ `strcmp()`, `strncmp()`, `stricmp()` → Compara duas strings e retorna: maior, menor ou igual.
- ❖ `strcat()` → Concatena uma string dentro de outra.
- ❖ `strstr()` → Localiza uma string dentro de outra.
- ❖ `strchr()`, `strrchr()` → localiza um caractere em uma string.
- ❖ `strtok()` → Retorna um ponteiro para a próxima ocorrência de uma palavra em um string.



Funções do Compilador padrão ANSI

Funções de Manipulação de Caracteres

- ❖ `strspn()`, `strcspn()` → Conta o número de caracteres presentes em uma string e que aparecem também em outra.
- ❖ `strpbrk()` → Localiza um caractere de uma string em outra string.
- ❖ `strlwr()` → Converte os caracteres de uma string em minúsculo.
- ❖ `sprintf()` → Imprime uma string ou constante de caracteres em outra string.



Funções do Compilador padrão ANSI

Operadores

- ❖ + → Soma.
- ❖ - → Subtração.
- ❖ * → Multiplicação.
- ❖ / → Divisão.
- ❖ % → Resto da divisão
- ❖ = → Atribuição
- ❖ & → Operação AND bit a bit
- ❖ | → Operação IOR bit a bit
- ❖ ^ → Operação XOR bit a bit
- ❖ ! → Operação NEGATION bit a bit



Funções do Compilador padrão ANSI

Operadores

- ❖ << → Rotação a esquerda
- ❖ >> → Rotação a direita
- ❖ ++ → Incremento
- ❖ -- → Decremento
- ❖ == → Igual
- ❖ != → Diferente
- ❖ > → Maior
- ❖ < → Menor
- ❖ >= Maior ou igual
- ❖ <= Menor ou igual



Agenda

- ✦ Praticidade de Programação em C
- ✦ Uso do compilador no MPLAB e editor da CCS
- ✦ Comandos da Linguagem C
- ✦ Funções do Compilador padrão ANSI
- ✦ **As funções especiais para o microcontrolador PIC**
- ✦ A migração do programa desenvolvido em C para outro modelo de PIC
- ✦ Biblioteca de funções da CCS
- ✦ Simulação e Depuração de programas



Funções Especiais para o Microcontrolador

Funções para RS232 I/O

- ❖ `getc()`, `getch()`, `getchar()`, `gets()`
- ❖ `fgetc()`, `fgets()`, `get_string()`
- ❖ `putc()`, `putchar()`, `puts()`
- ❖ `fputc()`, `fputs()`
- ❖ `printf()`, `fprintf()`
- ❖ `kbhit()`, `set_uart_speed()`
- ❖ `perror()`, `assert()`,



Funções Especiais para o Microcontrolador

Funções para I2C I/O

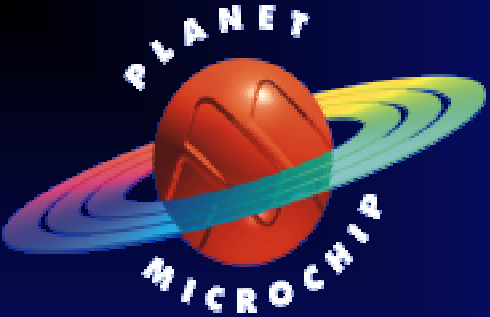
- ❖ `i2c_start()`
- ❖ `i2c_stop()`
- ❖ `i2c_read()`
- ❖ `I2c_write()`
- ❖ `I2c_poll()`



Funções Especiais para o Microcontrolador

Funções para SPI I/O

- ❖ `setup_spi()`
- ❖ `spi_read()`
- ❖ `spi_write()`
- ❖ `spi_data_is_in()`



Funções Especiais para o Microcontrolador

Funções para manipulação de Memória

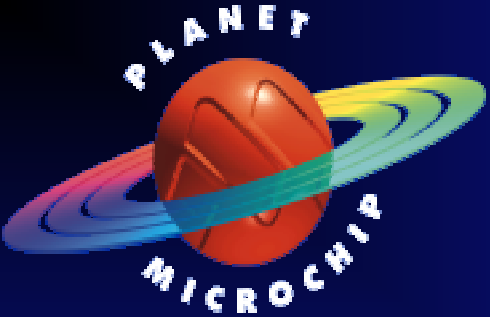
- ❖ `memset()`
- ❖ `memcpy()`
- ❖ `offsetof()`
- ❖ `offsetofbit()`



Funções Especiais para o Microcontrolador

Funções para Delay (atraso)

- ❖ Delay_cycles()
- ❖ Delay_us()
- ❖ Delay_ms()



Funções Especiais para o Microcontrolador

Funções para manipulação de bit e byte

- ❖ `shift_right()`, `shift_left()`
- ❖ `rotate_right()`, `rotate_left()`
- ❖ `bit_clear()`, `bit_set()`, `bit_test()`
- ❖ `swap()`
- ❖ `make8()`, `make16()`, `make32()`



Funções Especiais para o Microcontrolador

Funções para manipulação de I/O (Entrada e Saída)

- ❖ `output_low()`, `output_high()`, `output_float()`, `output_bit()`
- ❖ `output_a()`, `output_b()`, `output_c()`, `output_d()`, `output_e()`
- ❖ `input()`
- ❖ `input_a()`, `input_b()`, `input_c()`, `input_d()`, `input_e()`
- ❖ `port_b_pullups()`
- ❖ `set_tris_a()`, `set_tris_b()`, `set_tris_c()`, `set_tris_d()`, `set_tris_e()`



Funções Especiais para o Microcontrolador

Funções Analógicas

- ❖ `setup_comparator()`
- ❖ `setup_vref()`
- ❖ `setup_adc()`
- ❖ `setup_adc_ports()`
- ❖ `set_adc_channel()`
- ❖ `read_adc()`



Funções Especiais para o Microcontrolador

Funções para manipulação de Timers

- ❖ `setup_timer_0()`, `setup_rtcc()`, `setup_counter()`
- ❖ `setup_timer_1()`, `setup_timer_2()`, `setup_timer_3()`
- ❖ `set_rtcc()`, `set_timer_0()`, `set_timer_1()`, `set_timer_2()`, `set_timer_3()`
- ❖ `get_rtcc()`, `get_timer0()`, `get_timer1()`, `get_timer2()`, `get_timer3()`
- ❖ `setup_wdt()`, `restart_wdt()`



Funções Especiais para o Microcontrolador

Funções para o módulo CCP

- ❖ `setup_ccp1()`, `setup_ccp2()`
- ❖ `set_pwm1_duty()`, `set_pwm2_duty()`



Funções Especiais para o Microcontrolador

Funções para manipulação da memória EEPROM e FLASH

- ❖ `read_eeprom()`
- ❖ `write_eeprom()`
- ❖ `read_program_eeprom()`
- ❖ `write_program_eeprom()`
- ❖ `read_calibration()`



Funções Especiais para o Microcontrolador

Funções para controle do processador

- ❖ `sleep()`
- ❖ `reset_cpu()`, `restart_cause()`
- ❖ `enable_interrupts()`, `disable_interrupts()`
- ❖ `ext_int_edge()`
- ❖ `read_bank()`, `write_bank()`
- ❖ `label_address()`, `goto_address()`



SLIDE PRÊMIO

Envie Um E-mail

Para Microchip@artimar.com.br com o seguinte assunto:

“200 – Premio Seminario Microchip 2003”

Inclua o Nome do Participante e Endereço para envio.

Os Primeiros **5** participantes que enviarem
o e-mail acima receberão

1 Brinde Microchip

* Promoção Válida por Tempo Limitado e Somente 1 Brinde por Participante do Seminário



Funções Especiais para o Microcontrolador

Funções para manipulação da PSP

- ❖ `setup_psp()`
- ❖ `psp_input_full()`
- ❖ `psp_output_full()`
- ❖ `psp_overflow()`



Agenda

- ✦ Praticidade de Programação em C
- ✦ Uso do compilador no MPLAB e editor da CCS
- ✦ Comandos da Linguagem C
- ✦ Funções do Compilador padrão ANSI
- ✦ As funções especiais para o microcontrolador PIC
- ✦ **A migração do programa desenvolvido em C para outro modelo de PIC**
- ✦ Biblioteca de funções da CCS
- ✦ Simulação e Depuração de programas



Migração de programas em C para outros modelo de PIC

MPLAB IDE v6.22

File Edit View Project Debugger Programmer Tools Configure Window Help

Output

```
Build Find in Files
Deleting intermediary files... done.
Executing: "C:\Program files\Picc\CCSC.exe" "c1_e01.c" +FM +DF +LN +T -A +M +Z +Y=9 +EA
>>> Warning 203 "C:\ModuloC\c1_e01.c" Line 128(1,1): Condition always TRUE
Memory usage: ROM=1% RAM=3% - 3%
0 Errors, 1 Warnings.
Loaded C:\ModuloC\c1_e01.cof
BUILD SUCCEEDED
```

C:\ModuloC\c1_e01.c

```
17 // Este software está preparado para demonstrar o controle dos pinos de I/O
18 // este primeiro programa demonstrará o estado de um botão por meio de um led.
19
20 /* * * * * *
21 * Definição das variáveis internas do PIC
22 * * * * *
23 #define 16f877A
24 //#define 16f874A
25
26 #ifdef 16f877A
27 #include <16f877A.h> // microcontrolador utilizado
28 #else
29 #include <16f874A.h> // microcontrolador utilizado
30 #endif
31
32 /* * * * * *
```

Tamanho do arquivo: 0 KB

PIC16F877A

0xa3e9

Clique no objeto para selecionar



Migração de programas em C para outros modelo de PIC

The screenshot displays the MPLAB IDE v6.22 environment. The top menu bar includes File, Edit, View, Project, Debugger, Programmer, Tools, Configure, Window, and Help. Below the menu is a toolbar with various icons for file operations and debugging. The main workspace is divided into two panes:

- Output Window:** Shows the results of a build process. The text includes:

```
Deleting intermediary files... done.
Executing: "C:\Program files\Picc\CCSC.exe" "c1_e01.c" +FM +DF +LN +T -A +M +Z +Y=9 +EA
>>> Warning 203 "C:\ModuloC\c1_e01.c" Line 128(1,1): Condition always TRUE
Memory usage: ROM=2% RAM=3% - 3%
0 Errors, 1 Warnings.
Loaded C:\ModuloC\c1_e01.cof
BUILD SUCCEEDED
```
- Source Code Window (C:\ModuloC\c1_e01.c):** Shows the C source code with line numbers 17 through 32. The code includes comments in Portuguese and C preprocessor directives for PIC microcontroller models:

```
17 // Este software está preparado para demonstrar o controle dos pinos de I/O
18 // este primeiro programa demonstrará o estado de um botão por meio de um led.
19
20 /* *****
21 * Definição das variáveis internas do PIC
22 * *****
23 // #define 16f877A
24 #define 16f874A
25
26 #ifdef 16f877A
27 #include <16f877A.h> // microcontrolador utilizado
28 #else
29 #include <16f874A.h> // microcontrolador utilizado
30 #endif
31
32 /* *****
```

The status bar at the bottom indicates the current file is PIC16F877A, located at 0xa33b, line 20, column 45. The file size is 116 KB.



Agenda

- ✦ Praticidade de Programação em C
- ✦ Uso do compilador no MPLAB e editor da CCS
- ✦ Comandos da Linguagem C
- ✦ Funções do Compilador padrão ANSI
- ✦ As funções especiais para o microcontrolador PIC
- ✦ A migração do programa desenvolvido em C para outro modelo de PIC
- ✦ **Biblioteca de funções da CCS**
- ✦ Simulação e Depuração de programas



Biblioteca de funções da CCS (Drivers)

Biblioteca de Funções para EEPROM

- ❖ 2401.c
- ❖ 2402.c
- ❖ 2404.c
- ❖ 2408.c
- ❖ 24128.c
- ❖ 2416.c
- ❖ 24256.c
- ❖ 2432.c
- ❖ 2465.c



Biblioteca de funções da CCS (Drivers)

Biblioteca de Funções para EEPROM

- ❖ 25160.c
- ❖ 25320.c
- ❖ 25640.c
- ❖ 25C080.c
- ❖ 9346.c
- ❖ 9356.c
- ❖ 9356SPI.c
- ❖ 9366.c
- ❖ AT25256.c



Biblioteca de funções da CCS (Drivers)

Biblioteca de Funções para EEPROM

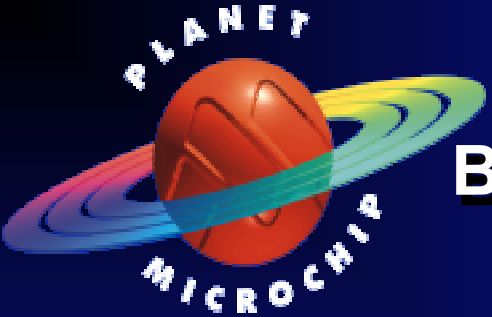
- ❖ AT29C1024.c
- ❖ CE51X.c
- ❖ CE62X.c
- ❖ CE67X.c



Biblioteca de funções da CCS (Drivers)

Biblioteca de Funções para RAM Serial

- ❖ 68HC68R1.c
- ❖ 68HC68R2.c



Biblioteca de funções da CCS (Drivers)

Biblioteca de Funções para expansão de I/O

- ❖ 74165.c
- ❖ 74595.c



Biblioteca de funções da CCS (Drivers)

Biblioteca de Funções para conversor Analógico Digital

- ❖ AD7705.c
- ❖ AD7715.c
- ❖ AD8320.c
- ❖ AD8400.c



Biblioteca de funções da CCS (Drivers)

Biblioteca de Funções para comunicação serial

- ❖ CRC.c
- ❖ Input.c
- ❖ Loader.c
- ❖ PIC_USB.h
- ❖ S7600.h
- ❖ SC28I19X.c
- ❖ Sntp.h
- ❖ USB.h
- ❖ USB.c



Biblioteca de funções da CCS (Drivers)

Biblioteca de Funções Sensores de temperatura

- ❖ DS1302.c
- ❖ DS1621.c
- ❖ DS1621M.c
- ❖ DS1631.c
- ❖ DS1624.c



Biblioteca de funções da CCS (Drivers)

Biblioteca de Funções teclado e lcd

- ❖ KDB.c
- ❖ LCD.c



Agenda

- ✦ Praticidade de Programação em C
- ✦ Uso do compilador no MPLAB e editor da CCS
- ✦ Comandos da Linguagem C
- ✦ Funções do Compilador padrão ANSI
- ✦ As funções especiais para o microcontrolador PIC
- ✦ A migração do programa desenvolvido em C para outro modelo de PIC
- ✦ Biblioteca de funções da CCS
- ✦ **Simulação e Depuração de programas**



Simulação e Depuração de Programas

MPLAB IDE v6.22

File Edit View Project Debugger Programmer Tools Configure Window Help

Watch

Address	Symbol Name	Value
0006	PORTB	00000000
0005	PORTA	00000000

Simulator Stimulus

File Stimulus Pin Stimulus

Add Row Delete Row Edit Complete

Type	Enable	Pin	Action	High cycles
Asynch	Fire	RA4	Toggle	

C:\ModuloC\c1_e01.c

```
120     portd=0x00;           // limpa
121     porte=0x00;         // limpa
122
123     /* *****
124     *           Loop principal
125     * *****
126
127     while(TRUE)
128     {
129     →   RESTART_WDT();
130       if(!botao)         // testa
131       {
132         led = 1;         // Se botão = 0, então led = 1
133       }
134       else led=0;       // caso contrário, led = 0
135     }
```

MPLAB SIM PIC16F877A pc:0x36 W:0xff Z dc c 0x9f5c Ln 129, Col 1 INS



Simulação e Depuração de Programas

MPLAB IDE v6.22

File Edit View Project Debugger Programmer Tools Configure Window Help

Watch

Add SFR: ADCON0 Add Symbol: botao

Address	Symbol Name	Value
0006	PORTB	00000000
0005	PORTA	00000000

Simulator Stimulus

File Stimulus Pin Stimulus

Add Row Delete Row Edit Complete

Type	Enable	Pin	Action	High cycles
Asynch	Fire	RA4	Toggle	

C:\ModuloC\c1_e01.c

```

120     portd=0x00;           // limpa
121     porte=0x00;         // limpa
122
123     /* *****
124     *                               Lab principal
125     * *****
126
127     while (TRUE)
128     {
129         RESTART_WDT();
130         if (!botao)       // testa
131         {
132             led = 1;      // Se botão = 0, então led = 1
133         }
134         else led=0;      // caso contrário, led = 0
135     }
  
```

MPLAB SIM PIC16F877A pc:0x39 W:0xff Z dc c 0x9f5c Ln 132, Col 1 INS



Simulação e Depuração de Programas

MPLAB IDE v6.22

File Edit View Project Debugger Programmer Tools Configure Window Help

Watch

Add SFR ADCON0 Add Symbol botao

Address	Symbol Name	Value
0006	PORTB	00000001
0005	PORTA	00000000

Watch 1 Watch 2 Watch 3 Watch 4

Simulator Stimulus

File Stimulus Pin Stimulus

Add Row Delete Row Edit Complete

Type	Enable	Pin	Action	High cycles
Asynch	Fire	RA4	Toggle	

C:\ModuloC\c1_e01.c

```
125 *****
126
127     while(TRUE)
128     {
129         RESTART_WDT();
130         if(!botao) // testa 1
131         {
132             led = 1; // Se bot:
133         }
134         else led=0; // caso c
135
136
137     }
138
139     // FIM DO PROGRAMA
140
```

MPLAB SIM PIC16F877A pc:0x3c W:0xff Z dc c 0x9f5c Ln 137, Col 1 INS



Simulação e Depuração de Programas

MPLAB IDE v6.22

File Edit View Project Debugger Programmer Tools Configure Window Help

Watch

Address	Symbol Name	Value
0006	PORTB	00000001
0005	PORTA	00010000

Simulator Stimulus

File Stimulus Pin Stimulus

Add Row Delete Row Edit Complete

Type	Enable	Pin	Action	High cycles
Asynch	Fire	RA4	Toggle	

C:\ModuloC\c1_e01.c

```
125 *****
126
127     while (TRUE)
128     {
129         RESTART_WDT();
130         if (!botao) // testa
131         {
132             led = 1; // Se bot:
133         }
134         else led=0; // caso c
135     }
136
137
138     // FIM DO PROGRAMA
139
140
```

MPLAB SIM PIC16F877A pc:0x36 W:0xff Z dc c 0x9f5c Ln 129, Col 1 INS



Simulação e Depuração de Programas

MPLAB IDE v6.22

File Edit View Project Debugger Programmer Tools Configure Window Help

Watch

Add SFR ADCON0 Add Symbol botao

Address	Symbol Name	Value
0006	PORTB	00000000
0005	PORTA	00010000

Watch 1 Watch 2 Watch 3 Watch 4

C:\ModuloC\c1_e01.c

```
125 *****
126
127     while(TRUE)
128     {
129         RESTART_WDT();
130         if(!botao) // testa
131         {
132             led = 1; // Se bot:
133         }
134         else led=0; // caso c
135     }
136
137     // FIM DO PROGRAMA
138
139
140
```

Simulator Stimulus

File Stimulus Pin Stimulus

Add Row Delete Row Edit Complete

Type	Enable	Pin	Action	High cycles
Asynch	Fire	RA4	Toggle	

MPLAB SIM PIC16F877A pc:0x3c W:0xff Z dc c 0x9f5c Ln 137, Col 1 INS



Simulação e Depuração de Programas

MPLAB IDE v6.22

File Edit View Project Debugger Programmer Tools Configure Window Help

Watch

Add SFR: ADCON0 | Add Symbol: botao

Address	Symbol Name	Value
0006	PORTB	00000001
0005	PORTA	00000000

Watch 1 | Watch 2 | Watch 3 | Watch 4

C:\ModuloC\c1_e01.c

```

125 *****
126
127     while (TRUE)
128     {
129         RESTART_WDT();
130         if (!botao) // testa b
131         {
132             led = 1; // Se botá
133         }
134         else led=0; // caso cc
135
136
137     }
138
139     // FIM DO PROGRAMA
  
```

Simulator Stimulus

File Stimulus | Pin Stimulus

Add Row | Delete Row | Edit Complete

Type	Enable	Pin	Action	High cycles
Asynch	Fire	RA4	Toggle	

Output

Build | Find in Files | MPLAB ICD 2

MPLAB SIM | MPLAB ICD 2 | PIC16F877A | pc:0x37 | W:0xff | Z dc c | 0x9f5c



Simulação e Depuração de Programas

PCW C Compiler IDE

File Project Edit Options Compile View Tools Debug Help

Microchip 14 bit

ICD Control Program V2.3

Download To Target

Test Target

Run Program

Update ICD Firmware

Test ICD

Configure Port

Check COMM

Help

Advanced...

Ready PIC16F877A

CCS 1.07 Rev #34 I/20

Device:Com1 - 9600

44

45

28:55

c:\module\c1_e01.c

PC=0000 W=00 ICD Control Pr

c1_e01

Start CCS C ICD Interface Program 9:55 PM

Debug

Peripherals Eval

Data EE Stack Watches

Monitor Break Log Breaks

Debug Configure RAM ROM

ICD-Serial

Compile Reload True

Mouse over eval False

ICD F/W CCS 1.07

Processor PIC16F877A

Port Com1

Baud Rate 9600

Break Style Hardware

Breakpoints 1

Apply Abort

When TRUE the target will be reloaded after every compile



MASTERS BRASIL 2003



**A Equipe Microchip Brasil
Agradece a Sua Presença!**

