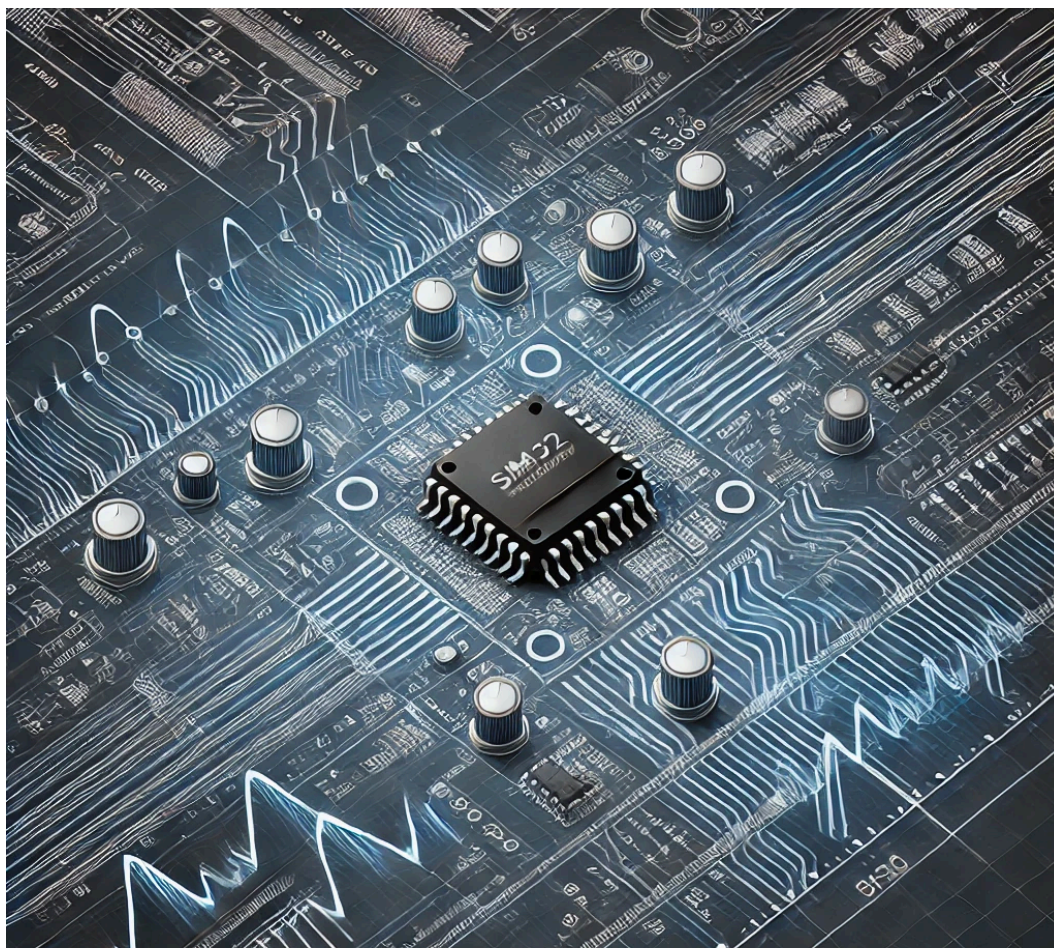


# IMPLEMENTACIÓN DE RETARDOS NO BLOQUEANTES EN STM32



**Curso:** Técnicas Digitales II

**Profesores:** Ing. Rubén Darío Mansilla, Ing. Lucas Abdala

**Integrantes:**

- Barrientos Lucas
- Cuellar Agustin
- Vera Monasterio Candela

**Fecha de entrega:** Noviembre 2024

## **6. Informe**

### **6.1 Introducción**

Las funciones no bloqueantes son una técnica importante en el desarrollo de aplicaciones embebidas, especialmente en microcontroladores, ya que permiten que el programa principal siga ejecutándose mientras se espera por una operación de tiempo, como un retardo o una lectura de entrada. Esto mejora la eficiencia y responsividad de la aplicación, evitando que el programa se detenga durante los tiempos de espera. En esta actividad, se implementará un driver de funciones no bloqueantes utilizando el temporizador SysTick del microcontrolador STM32, y se aplicará este driver en la modificación de aplicaciones desarrolladas previamente.

#### **Objetivo general**

El objetivo general de esta actividad es que el alumno logre implementar un módulo de software para trabajar con retardos no bloqueantes, aplicando este concepto en aplicaciones prácticas concretas utilizando el STM32CubeIDE.

#### **Importancia**

El uso de funciones no bloqueantes es crucial para el desarrollo de aplicaciones embebidas eficientes y responsivas. Al evitar que el programa se detenga durante los tiempos de espera, se puede mejorar la experiencia del usuario, la capacidad de respuesta del sistema y la utilización del procesador. Esto es especialmente importante en aplicaciones en tiempo real, donde la capacidad de respuesta rápida es fundamental.

## Descripción del Proceso

### Desarrollo del Driver Debounce

**Paso 1:** Se implementaron las funciones auxiliares necesarias para usar retardos no bloqueantes en un archivo fuente **main.c** con su correspondiente archivo de cabecera **main.h**.

**Paso 2:** Se crearon los archivos fuente **API\_Delay.c** y **API\_Delay.h** para armar el driver de funciones no bloqueantes, moviendo las definiciones de tipos de variables y declaraciones de funciones al archivo de cabecera **delay.h** y las implementaciones al archivo fuente **delay.c**.

**Paso 3:** Cada alumno tomó una aplicación desarrollada en la Actividad de Formación Práctica 2 y realizó las modificaciones necesarias para integrar el nuevo driver de funciones no bloqueantes, reemplazando el uso de la función bloqueante **HAL\_Delay()** por las nuevas funciones no bloqueantes desarrolladas.

## 6.2 Aplicaciones desarrolladas

### Modificación de las aplicaciones

#### Aplicacion 1

Se desarrolló el driver de retardos no bloqueantes implementando:

- **API\_delay.h:** Define la estructura **delay\_t** y los prototipos de funciones (**delayInit**, **delayRead**, **delayWrite**)
- **API\_delay.c:** Implementa la lógica de las funciones utilizando **HAL\_GetTick()** para el conteo de tiempo
- Se reemplazaron los retardos bloqueantes **HAL\_Delay()** por el nuevo sistema

Se realizó la integración en las aplicaciones previas:

- Para la App 1.1 no fue necesaria la implementación del driver al ser una secuencia simple sin interacción

## **Aplicacion 2**

Se modificó la aplicación original que utilizaba **HAL\_Delay()** por un sistema de retardos no bloqueantes. Los principales cambios incluyeron:

- Implementación de una estructura **delay\_t** para manejar el retardo de 200ms.
- Simplificación del código mediante un array de LEDs para mejor manejo de la secuencia.
- Se mantuvo la funcionalidad de la secuencia de LEDs (verde->azul->rojo) pero con mayor eficiencia al eliminar el bloqueo.
- Se eliminó temporalmente la función de cambio de dirección mediante pulsador para esta versión, quedando pendiente su implementación con el nuevo sistema.

## **Aplicacion 3**

Se modificó la aplicación que originalmente utilizaba **HAL\_Delay()** implementando múltiples retardos no bloqueantes. Los cambios principales fueron:

- Implementación de 5 estructuras **delay\_t** diferentes:
  - **delay\_sequence**: Para las secuencias 1, 2 y 4.
  - **delay\_led1, delay\_led2, delay\_led3**: Para la secuencia 3 con diferentes tiempos.

- Se mantuvo la funcionalidad del pulsador para cambiar entre secuencias.
- Se agregó una variable static bool **init\_delay** para inicializar los retardos solo una vez en cada secuencia.

## **Aplicacion 4**

Se modificó la aplicación que controlaba la frecuencia de parpadeo de los LEDs mediante **HAL\_Delay()**. Los cambios principales fueron:

- Implementación de una única estructura **delay\_t** (sharedDelay) para controlar el parpadeo sincronizado de los tres LEDs.
- Se creó un array **delayDurations[]** con los cuatro tiempos predefinidos (100ms, 250ms, 500ms, 1000ms).
- Se simplificó el control de frecuencia eliminando el switch-case, reemplazándolo por acceso directo al array.
- Se mejoró la detección del pulsador implementando una máquina de estados simple con una variable static.
- Se utiliza **delayWrite()** para cambiar dinámicamente el tiempo de retardo sin necesidad de reinicializar la estructura.

## **Autor de cada aplicación:**

- **Aplicación 1.1:** Barrientos Lucas
- **Aplicación 1.2:** Vera Monasterio Candela
- **Aplicación 1.3:** Cuellar Agustin
- **Aplicación 1.4:** Barrientos Lucas

## Observaciones:

### Dificultades encontradas

En esta fase del proyecto, realizamos el proceso de compilación y depuración para la aplicación 1. Durante el proceso de **compilación**, encontramos un **error**: el compilador no podía encontrar la ruta del driver que contenía la carpeta API, que era lo que nos pasaba en toda la AFP2. Resolvimos este problema de la misma manera que lo hicimos en el trabajo práctico anterior, siguiendo estos pasos:

1. Ir al menú "Project" y seleccionar "Properties".
2. En la ventana de propiedades, navegar a "C/C++ General > Paths and Symbols".
3. Seleccionar la pestaña "Includes".
4. Asegurarse de que la ruta **Drivers/API/Inc** esté agregada para el compilador. Esto le indicará al compilador que busque el archivo de cabecera en esa carpeta.
5. Agregar la ruta completa de **Drivers/API/Inc** en la sección del compilador C.

Después de realizar estos ajustes, el programa funcionó correctamente. Lo probamos en la placa y todo funcionó bien.

### 6.3 Link al repositorio grupal

[https://github.com/codecuellar/Grupo\\_6\\_TDII\\_2024](https://github.com/codecuellar/Grupo_6_TDII_2024)

En las siguientes capturas de pantalla podemos observar como queda cada aplicación una vez finalizada. Los problemas que se presentaron fueron solucionados después de agregar las direcciones que nos recomendó la IA haciendo que se pueda compilar correctamente, sin ningún error.



The screenshot displays the Keil IDE interface. On the left, the Project Explorer shows the project structure for 'AFP\_3\_App\_1\_1\_Grupo\_6\_2024', including folders for Binaries, Includes, Core, Inc, and Src. The main.c file is open in the Source Code Editor, showing a header section with comments and a main function. The Build Console at the bottom shows the output of the build process, indicating a successful build with 0 errors and 0 warnings.

The screenshot displays the STM32CubeIDE environment. The main editor shows the `main.c` file with the following content:

```

1  /** USER CODE BEGIN Header */
2
3  #include "main.h"
4  #include "string.h"
5  #include "API_GPIO.h"
6  #include "API_Delay.h"
7
8  /* Copyright (c) 2024 STMICROELECTRONICS.
9   * All rights reserved.
10
11  * This software is licensed under terms that can be found in the LICENSE file
12  * in the root directory of this software component.
13  * If no LICENSE file comes with this software, it is provided AS-IS.
14
15  */
16
17  /** USER CODE END Header */
18
19  #include "main.h"
20 #include "string.h" // So include driver GPIO
21 #include "API_GPIO.h"
22 #include "API_Delay.h"
23
24 /* Private includes -----------------------------------------------------*/
25 /* USER CODE BEGIN Includes */
26
27 /* USER CODE END Includes */
28
29 /* Private typedef -----------------------------------------------------*/
30 /* USER CODE BEGIN PTD */
31
32

```

The Build Targets window on the right shows the following targets:

- main.h
- string.h
- API\_GPIO.h
- API\_Delay.h
- TxConfig\_ETH\_TxPacketConfigTypeDef
- DMARxDescTab\_ETH\_DMADescTypeDef
- DMATxDescTab\_ETH\_DMADescTypeDef
- heth\_ETH\_HandleTypeDef
- huart3\_UART\_HandleTypeDef
- hpcd\_USB\_OTG\_FS\_PCD\_HandleTypeDef
- SystemClockConfig(void) void
- MX\_ETH\_Init(void) void
- MX\_USART3\_UART\_Init(void) void
- MX\_USB\_OTG\_FS\_PCD\_Init(void) void
- main(void) int
- SystemClockConfig(void) void
- MX\_ETH\_Init(void) void
- MX\_USART3\_UART\_Init(void) void
- MX\_USB\_OTG\_FS\_PCD\_Init(void) void
- Error\_Handler(void) void
- assert\_failed(uint8\_t,uint32\_t) void

The Build Console at the bottom shows the following output:

```

1  CDT Build Console [AFP_3_App_1_2_Grupo_6_2024]
2  Finished building: default.size.stdout
3
4  Finished building: AFP_3_App_1_2_Grupo_6_2024.list
5
6  21:52:14 Build Finished. 0 errors, 0 warnings. (took 34s.34ms)

```

## App 1.3

```

7  * @attention
8
9  * Copyright (c) 2024 STMicroelectronics.
10 * All rights reserved.
11 *
12 * This software is licensed under terms that can be found in the LICENSE file
13 * in the root directory of this software component.
14 * If no LICENSE file comes with this software, it is provided AS-IS.
15 *
16 *****
17 */
18 // USER CODE END Header */
19 // Includes -----*/
20 #include "main.h"
21 #include "API_GPIO.h"
22 #include "API_Delay.h"
23 #include "string.h"
24
25 // Private variables -----*/
26 // USER CODE BEGIN PV */
27 int direction = 1; // 1 para adelante, -1 para atrás
28 int current_led = 0;
29 uint8_t button_state = 0;
30 uint8_t last_button_state = 0;
31 int num_leds = 3;
32 delay_t led_delay; // Estructura para el delay de los LEDs
33
34 // USER CODE END PV */
35
36 // Private function prototypes -----*/
37 void SystemClock_Config(void);
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

```

## App 1.4

```

7  * @attention
8
9  * Copyright (c) 2024 STMicroelectronics.
10 * All rights reserved.
11 *
12 * This software is licensed under terms that can be found in the LICENSE file
13 * in the root directory of this software component.
14 * If no LICENSE file comes with this software, it is provided AS-IS.
15 *
16 *****
17 */
18 // USER CODE END Header */
19 // Includes -----*/
20 #include "main.h"
21 #include "string.h"
22 #include "API_GPIO.h"
23 #include "stm32f4xx_hal.h"
24 #include "API_Delay.h"
25
26 // Private includes -----*/
27 // USER CODE BEGIN Includes */
28
29 // USER CODE END Includes */
30
31 // Private typedef -----*/
32 // USER CODE BEGIN PTD */
33
34 // USER CODE END PTD */
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

```