

```
Terminal
~
Register group: general
r0      0x1      1      r1      0xac3f55      11288405      r2      0x3e8      1000      r3      0x0      0
r4      0x1      1      r5      0x1      0      r6      0x20f14      134932      r7      0x3ffff058      1073737816
r8      0x102ec      66540      r9      0x407ffff5c      188218268      r10      0x0      0      r11      0x20f14      134932
r12     0x3f97000      107331744      sp      0x407ffde0      0      lr      0x3fe2d958      1071831384      pc      0x103fc      0x103fc <alinh>
cpsr    0x60000010      1010612752      fpscr   0x0      0      fpsid   0x410430f0      1090793712      fpexc   0x40000000      1073741824
AFSR0_EL1 0x0      0      AFSR1_EL1 0x0      0      DBGDIDR 0x3515f021      890630177      DBGD0SR 0x0      0
DBGVCR  0x0      0      DBGPCR  0x0      0      DBGVCR  0x0      0      DBGVCR  0x0      0
PAB      0x0      0      DBGVCR  0x0      0      DBGVCR  0x0      0      DBGVCR  0x0      0
DBGCMR   0x0      0      TEECR   0x0      0      MIDR_EL1 0x412fc0f1      1093648025      CTR      0x8444c004      -2075869180
TCHTR    0x0      0      TIDBR0_EL1 0x0      0      PMCCNTR 0x0      0      TLBTR    0x0      0
TTBR1_EL1 0x0      0      MIDR     0x412fc0f1      1093648025      TIDBR    0x0      0      MPIDR_EL1 0x0      0      -2147483648
TTBR2     0x0      0      REVIDR_EL1 0x0      0      MIDR     0x412fc0f1      1093648025      JIDR     0x0      0
CLIDR    0xa200023      169869347      DPFR     0x0      0      WFSAR   0x0      0      IFAR     0x0      0
JMCRR    0x0      0      AIDR     0x0      0      CSELR    0x0      0      ID_PFR2 0x10      16
VBAR     0x0      0      AMAIR0   0x0      0      AMAIR0   0x0      0      PMEVTYPER0 0x0      0

~$ disas -i 11288405
1      .text
2      .global main
3
4      main:
5          LDR R1, = 11288405      @ divisor
6          LDR R2, = 1000          @ divisor
7          MOV R3, #0              @ quociente
8          MOV R4, #1              @ variável que realiza a soma do quociente
9          alinhar_divisor:
10         CMP R1, R2, LSL #1      @ comparamos o divisor com o divisor (<= 1)
11         BLT divisao             @ se divisor < divisor, passamos para divisao
12         LSL R2, R2, #1          @ anteriormente foi apenas uma verificação, aqui realizamos de fato o shift left do divisor
13         LSL R4, R4, #1          @ valor somado ao quociente seria equivalente ao número de shifts
14         B alinhar_divisor
15
16      divisao:
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
```

- Comparação de se o dividendo é menor que o divisor e , se sim, pulamos para label que efetua a divisao

```

Register group: general
r0 0x1 1 r1 0xac3f55 11288405 r2 0x3e8 1000 r3 0x0 0
r4 0x1 1 r5 0x1 1 r6 0x20f14 134932 r7 0x3ffff058 1073737816
r8 0x103ec 66540 r9 0x407fff5c 1082130268 r10 0x0 0 r11 0x20f14 134932
r12 0x3ff97000 1073311744 sp 0x407ffde0 0x407ffde0 lr 0x3fe2d958 1071831384
cpsr 0x2000010 536870928 fpscr 0x0 0 fpsid 0x410430f0 1090793712 fpexc 0x40000000 1073741824
AFSR0_EL1 0x0 0 AFSR1_EL1 0x0 0 DBGDIDR 0x3515f021 890630177 DBGDSAR 0x0 0
DBGBVR 0x0 0 DBGBCR 0x0 0 DBGWVR 0x0 0 DBGWCR 0x0 0
PAR 0x0 0 TEECR 0x0 0 MIDR_EL1 0x412fc0f1 1093648625 CTR 0x8444c004 -2075869180
TCMTR 0x0 0 TBR0_EL1 0x0 0 PMCCNTR 0x0 0 TLSTR 0x0 0
TTBR1_EL1 0x0 0 MIDR 0x412fc0f1 1093648625 MPIDR_EL1 0x80000000 -2147483648
TTBR2 0x0 0 REVIDR_EL1 0x0 0 MIDR 0x412fc0f1 1093648625 JIDR 0x0 0
CLIDR 0xa200023 169869347 DFAR 0x0 0 WFIAR 0x0 0 IFAR 0x0 0
DCCR 0x0 0 AIDR 0x0 0 CSSELR 0x0 0 ID_PFR2 0x10 16
VBAR 0x0 0 AMAIR1 0x0 0 AMAIRO 0x0 0 PMEVTYPER0 0x0 0

divisao_11288405.s
1 .text
2 .global main
3
4 main:
5     LDR R1, = 11288405 @ dividendo
6     LDR R2, = 1000 @ divisor
7     MOV R3, #0 @ quociente
8     MOV R4, #1 @ variável que realiza a soma do quociente
9     alinhar_divisor: @ alinhamos o divisor para a "mesma" casa do divisor como fazemos com divisao de decimais normalmente, onde subtraímos o divisor dos números mais significativ
10    > CMP R1, R2, LSL #1 @ se dividendo < divisor, passamos para divisao
11    > BLT divisao @ anteriormente foi apenas uma verificação, aqui realizamos de fato o shift left do divisor
12    > LSL R2, R2, #1 @ anteriormente foi apenas uma verificação, aqui realizamos de fato o shift left do divisor
13    > LSL R4, R4, #1 @ valor somado ao quociente seria equivalente ao número de shifts
14    > B alinhar_divisor
15
16 divisao:

```

Remote Thread 1.102040 In: alinhar_divisor L11 PC: 0x10400

(gdb) b 4
Ponto de parada 1 at 0x103ec: file divisao_11288405.s, line 5.
(gdb) b 5
Nota: ponto de parada 1 also set at pc 0x103ec.
Ponto de parada 2 at 0x103ec: file divisao_11288405.s, line 5.
(gdb) c
Continuing.

Breakpoint 1, main () at divisao_11288405.s:5
(gdb) s
(gdb) s
(gdb) s
(gdb) s
(gdb) s
alinhar_divisor () at divisao_11288405.s:10
(gdb) s
(gdb) s
(gdb) s

- Como o dividendo era maior que o divisor, efetuamos de fato o shift e incrementamos o contador. Esta etapa será repetida N vezes até alinharmos dividendo e divisor e a condição (BLT divisao) ser positiva

```

Register group: general
r0 0x1 1 r1 0xac3f55 11288405 r2 0x7d0 2000 r3 0x0 0
r4 0x1 1 r5 0x1 1 r6 0x20f14 134932 r7 0x3ffff058 1073737816
r8 0x103ec 66540 r9 0x407fff5c 1082130268 r10 0x0 0 r11 0x20f14 134932
r12 0x3ff97000 1073311744 sp 0x407ffde0 0x407ffde0 lr 0x3fe2d958 1071831384
cpsr 0x2000010 536870928 fpscr 0x0 0 fpsid 0x410430f0 1090793712 fpexc 0x40000000 1073741824
AFSR0_EL1 0x0 0 AFSR1_EL1 0x0 0 DBGDIDR 0x3515f021 890630177 DBGDSAR 0x0 0
DBGBVR 0x0 0 DBGBCR 0x0 0 DBGWVR 0x0 0 DBGWCR 0x0 0
PAR 0x0 0 TEECR 0x0 0 MIDR_EL1 0x412fc0f1 1093648625 CTR 0x8444c004 -2075869180
TCMTR 0x0 0 TBR0_EL1 0x0 0 PMCCNTR 0x0 0 TLSTR 0x0 0
TTBR1_EL1 0x0 0 MIDR 0x412fc0f1 1093648625 MPIDR_EL1 0x80000000 -2147483648
TTBR2 0x0 0 REVIDR_EL1 0x0 0 MIDR 0x412fc0f1 1093648625 JIDR 0x0 0
CLIDR 0xa200023 169869347 DFAR 0x0 0 WFIAR 0x0 0 IFAR 0x0 0
DCCR 0x0 0 AIDR 0x0 0 CSSELR 0x0 0 ID_PFR2 0x10 16
VBAR 0x0 0 AMAIR1 0x0 0 AMAIRO 0x0 0 PMEVTYPER0 0x0 0

divisao_11288405.s
1 .text
2 .global main
3
4 main:
5     LDR R1, = 11288405 @ dividendo
6     LDR R2, = 1000 @ divisor
7     MOV R3, #0 @ quociente
8     MOV R4, #1 @ variável que realiza a soma do quociente
9     alinhar_divisor: @ alinhamos o divisor para a "mesma" casa do divisor como fazemos com divisao de decimais normalmente, onde subtraímos o divisor dos números mais significativ
10    > CMP R1, R2, LSL #1 @ se dividendo < divisor, passamos para divisao
11    > BLT divisao @ anteriormente foi apenas uma verificação, aqui realizamos de fato o shift left do divisor
12    > LSL R2, R2, #1 @ anteriormente foi apenas uma verificação, aqui realizamos de fato o shift left do divisor
13    > LSL R4, R4, #1 @ valor somado ao quociente seria equivalente ao número de shifts
14    > B alinhar_divisor
15
16 divisao:

```

Remote Thread 1.102040 In: alinhar_divisor L13 PC: 0x10408

(gdb) b 4
Ponto de parada 1 at 0x103ec: file divisao_11288405.s, line 5.
(gdb) b 5
Nota: ponto de parada 1 also set at pc 0x103ec.
Ponto de parada 2 at 0x103ec: file divisao_11288405.s, line 5.
(gdb) c
Continuing.

Breakpoint 1, main () at divisao_11288405.s:5
(gdb) s
(gdb) s
(gdb) s
(gdb) s
(gdb) s
alinhar_divisor () at divisao_11288405.s:10
(gdb) s
(gdb) s
(gdb) s
(gdb) s
(gdb) s

- Depois de muitas iterações, dividendo e divisor estão alinhados e podemos partir para a divisão (atente-se para o valor de r2 e r4):

[illegible]

- Efetua a subtração e incrementa quociente (atente-se para r1 e r3):

[illegible]

- Nossa divisão não chegou ao fim, então o branch será executado para voltar a label “divisao” e repetir os últimos passos aqui descritos:

```
Terminal Terminal
~Register group: general~
r0 0x1 1 r1 0x2f3f55 3696405 r2 0x3e8000 4896000 r3 0x2000 8192
r4 0x1000 4000 r5 0x1 1 r6 0x20f14 134932 r7 0x3fff058 1073737816
r8 0x103ec 65540 r9 0x407fff5c 1082130268 r10 0x0 0 r11 0x20f14 134932
r12 0x3ff97000 1073311744 sp 0x407ffde0 0x407ffde0 lr 0x3fe2d958 1071831384 pc 0x10428 0x10428 <continua>
cpsr 0x10 16 fpscr 0x0 0 fpsid 0x410430f0 1890793712 tpexc 0x40000000 1073741824
AFSR0_EL1 0x0 0 AFSR1_EL1 0x0 0 DBGDIDR 0x3515f021 890630177 DBGDSCR 0x0 0
PAR 0x0 0 DBGCBVR 0x0 0 DBGCBR 0x0 0 DBGMCR 0x0 0
DBGMCR 0x0 0 TECCR 0x0 0 MIDR_EL1 0x412fc0f1 1093648625 CTR 0x8444c004 -2075869180
TCNTR 0x0 0 TTBAR_EL1 0x0 0 PMCCNTR 0x0 0 ILBTR 0x0 0
TTBARI_EL1 0x0 0 MIDR 0x412fc0f1 1093648625 TTBCR 0x0 0 HPMIDR_EL1 0x80000000 -2147483648
TTBCR2 0x0 0 REVIDR_EL1 0x0 0 MIDR 0x412fc0f1 1093648625 DIDR 0x0 0
CLIDR 0xa200023 169869347 DFAR 0x0 0 WFIAR 0x0 0 IFAR 0x0 0
JIKR 0x0 0 AIDR 0x0 0 CSSELR 0x0 0 ID_PFR2 0x10 16
VBAR 0x0 0 AMAIR1 0x0 0 AMAIR0 0x0 0 PKEVTYPER0 0x0 0

divisao 11288405.s
17 CMP R1, R2 @ caso R1- R2 seja negativo, devemos pular as operacoes e
18 BNE continua @ mover uma casa
19 SUB R1, R1, R2 @ subtracao do dividendo e divisor
20 ADD R3, R3, R4 @ soma do var no quociente
21 continua:
22 LSRs R2, R2, #1 @ "correcao" do dividendo movendo uma casa para direita
23 LSRs R4, R4, #1 @ "correcao" do var movendo uma casa para direita
> 24 BNE divisao @ a operacao termina quando inc eh zerado
25
26 MOV R5, R1 @ colocamos o resto no R5
27
28 SWI 0 @ fim do codigo
29
30
31
32

remote Thread 1:10280 In: continua L24 PC: 0x10428
(gdb) s
(gdb) s
(gdb) s
(gdb) s
(gdb) s
(gdb) s
(gdb) s
(gdb) s
divisao () at divisao_11288405.s:17
(gdb) s
(gdb) s
(gdb) s
divisao () at divisao_11288405.s:22
(gdb) s
(gdb) s
(gdb) s
```

- Esta é uma das últimas iterações, note o dividendo (r1) que diminui e o quociente (r3) que aumentou. Este resultado já está próximo do esperado.

Terminal

Register group: general

r0	0x1	1	r1	0x195	405	r2	0xfa0	4000	r3	0x2c18	11288
r4	0x103ec	4	r5	0x1	1	r6	0x20f14	134932	r7	0x3ffff058	107373816
r8	0x103ec	0x340	r9	0x407ffff5c	1082130268	r10	0x0	0	r11	0x20f14	134932
r12	0x3ff97000	1073311744	sp	0x407ffde0	0x407ffde0	lr	0x3fc2d958	1071831384	pc	0x10428	0x10428 <cont
cpsr	0x10	16	fpscr	0x0	0	fpsidr	0x410430f0	1090793712	fpexc	0x40000000	1073741024
AFSR0_EL1	0x0	0	AFSR1_EL1	0x0	0	DBGDIDR	0x3515f021	890630177	DBGDSAR	0x0	0
DBGDVR	0x0	0	DBGBCR	0x0	0	DBGWVR	0x0	0	DBGCMR	0x0	0
DAB	0x0	0	DBGDVR	0x0	0	DBGBCR	0x0	0	DBGWVR	0x0	0
DBGCMR	0x0	0	TEECR	0x0	0	MIDR_EL1	0x412fc0f1	1093648625	CTR	0x8444c004	-2075869180
TCHTR	0x0	0	TBRO_EL1	0x0	0	PMCCNTR	0x0	0	TLBR	0x0	0
TBTR1_EL1	0x0	0	MIDR	0x412fc0f1	1093648625	TIDCR	0x0	0	NPIOR_EL1	0x80000000	-2147483648
TIDCR	0x0	0	REVIDR_EL1	0x0	0	MIDR	0x412fc0f1	1093648625	JIDR	0x0	0
CLIDR	0xa200023	169869347	DFAR	0x0	0	WFAR	0x0	0	IFAR	0x0	0
JMCR	0x0	0	AFAR	0x0	0	CSELRL	0x0	0	ID_PFR2	0x10	16
VBAR	0x0	0	AMAIR1	0x0	0	AMAIR0	0x0	0	PHEVTPER0	0x0	0

Terminal

```

17      CMP R1, R2      @ caso R1: R2 seja negativo, devemos pular as operacoes e
18      BMI continua    @ mover uma casa
19      SUB R1, R1, R2   @ subtração do dividendo e divisor
20      ADD R3, R3, R1   @ soma do var no quociente
21      continua:
22      LSRS R2, R2, #1  @ "correcao" do dividendo movendo uma casa para direita
23      LSRS R3, R3, #1  @ "correcao" do var movendo uma casa para direita
24      BNE divisao      @ a operacao termina quando lnc eh zerado
25
26      MOV R5, R1        @ colocamos o resto no R5
27
28      SWI 0             @ fim do codigo
29
30
31
32

```

```

remote thread 1.102048 In: continua
divisao () at divisao_11288405.s:17
(gdb) s
(gdb) s
(gdb) s
(gdb) s
(gdb) s
continua () at divisao_11288405.s:22
(gdb) s
(gdb) s
(gdb) s
divisao () at divisao_11288405.s:17
(gdb) s
(gdb) s
(gdb) s
(gdb) s
(gdb) s
continua () at divisao_11288405.s:22
(gdb) s
(gdb) s
(gdb) s
(gdb) s

```

L24

PC: 0x10428

- Por fim, o resto foi movido para r5 e temos o seguinte resultado:

Quociente: 11288

Resto: 405

- Note que esses resultados estão em r3 e r5:

```

Register group: general
r0 0x1 1 r1 0x195 405 r2 0x1f4 500 r3 0x2c18 11288
r4 0x0 0 r5 0x195 405 r6 0x20f14 134932 r7 0x3ffff058 1073737816
r8 0x103ec 66540 r9 0x407ffff5c 1082130268 r10 0x0 0 r11 0x20f14 134932
r12 0x3ff97000 1073311744 sp 0x407ffde0 0x407ffde0 lr 0x3fe2d958 1071831384
cpsr 0x60000010 1610612752 fpscr 0x0 0 fpstid 0x410430f0 1090793712
AFSR0_EL1 0x0 0 AFSR1_EL1 0x0 0 DBGCIDR 0x3515f021 890630177
DBGCBVR 0x0 0 DBGCBVR 0x0 0 DBGCBVR 0x0 0 DBGCBVR 0x0 0
PAR 0x0 0 DBGCBVR 0x0 0 MIDR_EL1 0x412fc0f1 1093648625
DBGMCR 0x0 0 TEECR 0x0 0 PMCCNTR 0x0 0
TCNTR 0x0 0 TTBR0_EL1 0x0 0 TTBR 0x0 0
TTBR1_EL1 0x0 0 MIDR 0x412fc0f1 1093648625 TTBR 0x0 0
TTBR2 0x0 0 REVIDR_EL1 0x0 0 MIDR 0x412fc0f1 1093648625
CLIDR 0xa200023 169869347 DFAR 0x0 0 WPAR 0x0 0
JMCRR 0x0 0 AIDR 0x0 0 CSSELR 0x0 0 ID_PFR2 0x10 16
VBAR 0x0 0 AMAIR1 0x0 0 AMAIR0 0x0 0 PHEVTYPER0 0x0 0

divisao 11288405.s
1 .text
2 .global main
3
4 main:
5     LDR R1, = 11288405 @ dividend
6     LDR R2, = 1000 @ divisor
7     MOV R3, #0 @ quociente
8     MOV R4, #1 @ variável que realiza a soma do quociente
9     alinhar_divisor: @ alinhamos o divisor para a "mesma" casa do divisor como fazemos com divisão de decimais normalmente, onde subtraímos o divisor dos números mais significativos
10    CMP R1, R2, LSL #1 @ comparamos o dividendo com o divisor (<< 1)
11    BLT divisao @ se dividendo < divisor, passamos para divisao
12    LSL R2, R2, #1 @ anteriormente foi apenas uma verificação, aqui realizamos de fato o shift left do divisor
13    LSL R1, R1, #1 @ valor somado ao quociente seria equivalente ao número de shifts
14    B alinhar_divisor
15
16 divisao:

remote Thread 1.102040 In: continua
(gdb) s
continua () at divisao_11288405.s:22
(gdb) s
(gdb) s
(gdb) s
divisao () at divisao_11288405.s:17
(gdb) s
(gdb) s
continua () at divisao_11288405.s:22
(gdb) s
(gdb) s
(gdb) s
(gdb) s
(gdb) s
Program received signal SIGILL, Illegal instruction.
continua () at divisao_11288405.s:5
(gdb)

```