

# Name: Ladinu Chandrasinghe

## CS333 Project 1

### typescript

```

1 Script started on Sun Apr  6 14:31:32 2014
2 [?1034hbash-3.2$ asm Hello.s
3 bash-3.2$ ls
4 DISK          Hello.s          Runtime.s          makefile          typescript
5 Echo.s          HelloWorld.c      System.c          proj1.pdf
6 Hello.o          HelloWorld.h      System.h          proj1_files
7 bash-3.2$ ldd Hello.o -o Hello
8 bash-3.2$ ls
9 DISK          Hello.o          HelloWorld.h      System.h          proj1_files
10 Echo.s          Hello.s          Runtime.s          makefile          typescript
11 Hello          HelloWorld.c      System.c          proj1.pdf
12 bash-3.2$ blitz -g Hello
13 Beginning execution...
14 Hello, world!
15
16
17
18 **** A 'debug' instruction was encountered ****
19
20 Done! The next instruction to execute will be:
21 000080: A1FFFFB8          jmp      0xFFFFB8          ! targetAddr = main
22
23
24 Entering machine-level debugger...
25
26 =====
27 =====
28 ===== The BLITZ Machine Emulator =====
29 =====
30 ===== Copyright 2001-2007, Harry H. Porter III =====
31 =====
32 =====
33
34 Enter a command at the prompt. Type 'quit' to exit or 'help' for
35 info about commands.
36 > q
37 Number of Disk Reads      = 0
38 Number of Disk Writes     = 0
39 Instructions Executed     = 1705
40 Time Spent Sleeping       = 0
41 Total Elapsed Time       = 1705
42 bash-3.2$ asm Echo.s
43 bash-3.2$ ls
44 DISK          Hello          HelloWorld.c      System.c          proj1.pdf
45 Echo.o          Hello.o          HelloWorld.h      System.h          proj1_files
46 Echo.s          Hello.s          Runtime.s          makefile          typescript
47 bash-3.2$ ldd Echo.o -o Echo
48 bash-3.2$ blitz Echo
49 =====
50 =====

```

```

51 ===== The BLITZ Machine Emulator =====
52 =====
53 ===== Copyright 2001-2007, Harry H. Porter III =====
54 =====
55 =====
56
57 Enter a command at the prompt. Type 'quit' to exit or 'help' for
58 info about commands.
59 > g
60 Beginning execution...
61 foo
62 foo
63 asd
64 asd
65 echo
66 echo
67 q
68 q
69
70 **** A 'debug' instruction was encountered ****
71
72 Done! The next instruction to execute will be:
73 cont:
74 0000A4: A1FFFFAC jmp 0xFFFFAC ! targetAddr = loop
75 > go
76 Beginning execution...
77
78 a
79 a
80 b
81 b
82 q
83 q
84
85 **** A 'debug' instruction was encountered ****
86
87 Done! The next instruction to execute will be:
88 cont:
89 0000A4: A1FFFFAC jmp 0xFFFFAC ! targetAddr = loop
90 > q
91 Number of Disk Reads = 0
92 Number of Disk Writes = 0
93 Instructions Executed = 242833558
94 Time Spent Sleeping = 0
95 Total Elapsed Time = 242833558
96 bash-3.2$ kpl -unsafe System
97 bash-3.2$ ls
98 DISK Hello HelloWorld.h System.s typescript
99 Echo Hello.o Runtime.s makefile
100 Echo.o Hello.s System.c proj1.pdf
101 Echo.s HelloWorld.c System.h proj1_files
102 bash-3.2$ asm System.s
103 bash-3.2$ ls
104 DISK Hello HelloWorld.h System.o proj1_files
105 Echo Hello.o Runtime.s System.s typescript
106 Echo.o Hello.s System.c makefile
107 Echo.s HelloWorld.c System.h proj1.pdf
108 bash-3.2$ kpl HelloWorld

```

```

109 bash-3.2$ ls
110 DISK          Hello          HelloWorld.h    System.h        proj1.pdf
111 Echo          Hello.o        HelloWorld.s    System.o        proj1_files
112 Echo.o        Hello.s        Runtime.s      System.s        typescript
113 Echo.s        HelloWorld.c    System.c       makefile
114 bash-3.2$ asm HelloWorld.s
115 bash-3.2$ ls
116 DISK          Hello          HelloWorld.h    System.c        makefile
117 Echo          Hello.o        HelloWorld.o    System.h        proj1.pdf
118 Echo.o        Hello.s        HelloWorld.s    System.o        proj1_files
119 Echo.s        HelloWorld.c    Runtime.s      System.s        typescript
120 bash-3.2$ asm Runtime.s
121 bash-3.2$ ls
122 DISK          Hello.o        HelloWorld.s    System.o        typescript
123 Echo          Hello.s        Runtime.o      System.s
124 Echo.o        HelloWorld.c    Runtime.s      makefile
125 Echo.s        HelloWorld.h    System.c       proj1.pdf
126 Hello         HelloWorld.o    System.h        proj1_files
127 bash-3.2$ ldd Runtime.o Sysr[Ktem.o HelloWorld.o -o HelloWorld
128 bash-3.2$ kpl [K[K[K[Kls
129 DISK          Hello.o        HelloWorld.o    System.h        proj1_files
130 Echo          Hello.s        HelloWorld.s    System.o        typescript
131 Echo.o        HelloWorld     Runtime.o      System.s
132 Echo.s        HelloWorld.c    Runtime.s      makefile
133 Hello         HelloWorld.h    System.c       proj1.pdf
134 bash-3.2$ kpl System
135 System.h:30: ***** ERROR at PTR: Using 'ptr to void' is unsafe; you must compile with the
'unsafe' option if you wish to do this
136 System.h:31: ***** ERROR at PTR: Using 'ptr to void' is unsafe; you must compile with the
'unsafe' option if you wish to do this
137 System.h:32: ***** ERROR at PTR: Using 'ptr to void' is unsafe; you must compile with the
'unsafe' option if you wish to do this
138 System.c:64: ***** ERROR at "+": Adding ptrs to ints is an unsafe operation; you must com
pile with the 'unsafe' option if you wish to do this
139 System.c:92: ***** ERROR at PTR: Using 'ptr to void' is unsafe; you must compile with the
'unsafe' option if you wish to do this
140 System.c:100: ***** ERROR at AS_PTR_TO: Using 'asPtrTo' is an unsafe operation; you must
compile with the 'unsafe' option if you wish to do this
141 System.c:113: ***** ERROR at "&": Taking the address of field within a record or obje
ct is an unsafe operation; you must compile with the 'unsafe' option if you wish to do thi
142 System.c:113: ***** ERROR at "&": The expression on the righthand side of this assign
ment does not have the correct type
143 System.c:113:          The type of the expression is: ptr to ptr to DISPATCH_TABLE
144 System.c:92:          The expected type is: ptr to ptr to void
145 System.c:120: ***** ERROR at "+": Adding ptrs to ints is an unsafe operation; you must co
mpile with the 'unsafe' option if you wish to do this
146 System.c:149: ***** ERROR at AS_PTR_TO: Using 'asPtrTo' is an unsafe operation; you must
compile with the 'unsafe' option if you wish to do this
147 System.c:207: ***** ERROR at "&": Taking the address of an element within an array is
an unsafe operation; you must compile with the 'unsafe' option if you wish to do this
148 System.c:208: ***** ERROR at AS_PTR_TO: Using 'asPtrTo' is an unsafe operation; you must
compile with the 'unsafe' option if you wish to do this
149 System.c:208: ***** ERROR at AS_PTR_TO: Using 'asPtrTo' is an unsafe operation; you must
compile with the 'unsafe' option if you wish to do this
150 System.c:212: ***** ERROR at "+": Adding ptrs to ints is an unsafe operation; you must co
mpile with the 'unsafe' option if you wish to do this
151 System.c:246: ***** ERROR at AS_PTR_TO: Using 'asPtrTo' is an unsafe operation; you must
compile with the 'unsafe' option if you wish to do this

```



```

207 > q
208 Number of Disk Reads      = 0
209 Number of Disk Writes    = 0
210 Instructions Executed     = 1296
211 Time Spent Sleeping       = 0
212     Total Elapsed Time    = 1296
213 bash-3.2$ cat HelloWorld.c
214 code Hello
215
216 -- This is the "hello world" program, for use in Project 1.
217
218 function main ()
219     print ("Hello, world...\n")
220     --foo (10)
221     endFunction
222
223 function foo (x: int)
224     bar (x+1)
225     endFunction
226
227 function bar (a: int)
228     var b: int
229     b = a + 1
230     print ("The value of b is ")
231     printInt (b)
232     nl ()
233     debug
234     bar (b)
235     endFunction
236
237 endCode
238 bash-3.2$ cat HelloWorld.c
239 code Hello
240
241 -- This is the "hello world" program, for use in Project 1.
242
243 function main ()
244     print ("Hello, world...\n")
245     foo (10)
246     endFunction
247
248 function foo (x: int)
249     bar (x+1)
250     endFunction
251
252 function bar (a: int)
253     var b: int
254     b = a + 1
255     print ("The value of b is ")
256     printInt (b)
257     nl ()
258     debug
259     bar (b)
260     endFunction
261
262 endCode
263 bash-3.2$ make
264 kpl HelloWorld

```

```

265 asm HelloWorld.s
266 ldd Runtime.o System.o HelloWorld.o -o HelloWorld
267 bash-3.2$ ls
268 DISK          Hello.o          HelloWorld.o    System.h       proj1_files
269 Echo          Hello.s          HelloWorld.s    System.o       typescript
270 Echo.o        HelloWorld      Runtime.o      System.s
271 Echo.s        HelloWorld.c    Runtime.s      makefile
272 Hello         HelloWorld.h    System.c       proj1.pdf
273 bash-3.2$ ls makecat HelloWorld.c
274 [C[C[C[C[C[C[C[C[C[C]Cblitz -g HelloWorld
275 [C[C[C[C[C[C[C[C[C[C]C3l@ldd Runtime.o System.o HelloWorld.o -o[C[C[C[C[C
[C[C[C[C[C[C
276 [C[C[C[C[C[C[C[C[C[C]C3lPblitz -g[C[C[C[C[C[C[C[C[C[C]C
277 Beginning execution...
278 ===== KPL PROGRAM STARTING =====
279
280 Hello, world...
281
282 The value of b is 12
283
284
285
286 **** A 'debug' instruction was encountered ****
287
288 Done! The next instruction to execute will be:
289 0028A4: 8B1EFFF0      load      [r14+0xFFFF0],r1      ! decimal: -16
290
291
292 Entering machine-level debugger...
293
294 =====
295 =====
296 ===== The BLITZ Machine Emulator =====
297 =====
298 ===== Copyright 2001-2007, Harry H. Porter III =====
299 =====
300 =====
301
302 Enter a command at the prompt. Type 'quit' to exit or 'help' for
303 info about commands.
304 > go
305 Beginning execution...
306 The value of b is 13
307
308
309
310 **** A 'debug' instruction was encountered ****
311
312 Done! The next instruction to execute will be:
313 0028A4: 8B1EFFF0      load      [r14+0xFFFF0],r1      ! decimal: -16
314 > go
315 Beginning execution...
316 The value of b is 14
317
318
319
320 **** A 'debug' instruction was encountered ****
321

```

```

322 Done! The next instruction to execute will be:
323 0028A4: 8B1EFFF0      load      [r14+0xFFF0],r1      ! decimal: -16
324 > g
325 Beginning execution...
326 The value of b is 15
327
328
329
330 ***** A 'debug' instruction was encountered *****
331
332 Done! The next instruction to execute will be:
333 0028A4: 8B1EFFF0      load      [r14+0xFFF0],r1      ! decimal: -16
334 > st
335      Function/Method      Frame Addr      Execution at...
336      =====
337      bar      00FFFE74      HelloWorld.c, line 20
338      bar      00FFFE90      HelloWorld.c, line 21
339      bar      00FFFEAC      HelloWorld.c, line 21
340      bar      00FFFE8C      HelloWorld.c, line 21
341      foo      00FFFE70      HelloWorld.c, line 11
342      main     00FFFEF8      HelloWorld.c, line 7
343 Bottom of activation frame stack!
344 > fr
345 ===== Frame number 0 (where StackTop = 0) =====
346 Function Name:      bar
347 Filename:           HelloWorld.c
348 Execution now at:   line 20
349 Frame Addr:         00FFFE74
350 frameSize:         12
351 totalParmSize:      4
352      =====
353      sp--> -20      00FFFE60: 0000000F
354      -16      00FFFE64: 0000000F
355      -12      00FFFE68: 0000906C
356 R.D.ptr: -8      00FFFE6C: 000028D0
357      r13: -4      00FFFE70: 00000015
358      fp: 0      00FFFE74: 00FFFE90
359 RetAddr: 4      00FFFE78: 000028B8
360      =====
361      Args: 8      00FFFE7C: 0000000E
362
363 PARAMETERS AND LOCAL VARIABLES WITHIN THIS FRAME:
364 =====
365      a: int
366      8      00FFFE7C: 0000000E      value = 14
367      _temp_15
368      -12      00FFFE68: 0000906C
369      b: int
370      -16      00FFFE64: 0000000F      value = 15
371 =====
372 > down
373 ===== Frame number 1 (where StackTop = 0) =====
374 Function Name:      bar
375 Filename:           HelloWorld.c
376 Execution now at:   line 21
377 Frame Addr:         00FFFE90
378 frameSize:         12
379 totalParmSize:      4

```

```

380                                     =====
381             -20    00FFFE7C:    0000000E
382             -16    00FFFE80:    0000000E
383             -12    00FFFE84:    0000906C
384 R.D.ptr:   -8     00FFFE88:    000028D0
385             r13:   -4    00FFFE8C:    00000015
386             fp:    0     00FFFE90:    00FFFEAC
387 RetAddr:    4     00FFFE94:    000028B8
388                                     =====
389     Args:     8     00FFFE98:    0000000D
390
391 PARAMETERS AND LOCAL VARIABLES WITHIN THIS FRAME:
392 =====
393     a: int
394             8     00FFFE98:    0000000D    value = 13
395     _temp_15
396            -12    00FFFE84:    0000906C
397     b: int
398            -16    00FFFE80:    0000000E    value = 14
399 =====
400 > down
401 ===== Frame number 2 (where StackTop = 0) =====
402 Function Name:    bar
403 Filename:         HelloWorld.c
404 Execution now at: line 21
405 Frame Addr:       00FFFEAC
406 frameSize:        12
407 totalParmSize:    4
408                                     =====
409             -20    00FFFE98:    0000000D
410             -16    00FFFE9C:    0000000D
411             -12    00FFFEA0:    0000906C
412 R.D.ptr:   -8     00FFFEA4:    000028D0
413             r13:   -4    00FFFEA8:    00000015
414             fp:    0     00FFFEAC:    00FFFEAC
415 RetAddr:    4     00FFFEB0:    000028B8
416                                     =====
417     Args:     8     00FFFEB4:    0000000C
418
419 PARAMETERS AND LOCAL VARIABLES WITHIN THIS FRAME:
420 =====
421     a: int
422             8     00FFFEB4:    0000000C    value = 12
423     _temp_15
424            -12    00FFFEA0:    0000906C
425     b: int
426            -16    00FFFE9C:    0000000D    value = 13
427 =====
428 > down
429 ===== Frame number 3 (where StackTop = 0) =====
430 Function Name:    bar
431 Filename:         HelloWorld.c
432 Execution now at: line 21
433 Frame Addr:       00FFFEAC
434 frameSize:        12
435 totalParmSize:    4
436                                     =====
437             -20    00FFFEB4:    0000000C

```



```

438          -16  00FFFE8:  0000000C
439          -12  00FFFEBC:  0000906C
440 R.D.ptr:  -8  00FFFE0:  000028D0
441      r13:  -4  00FFFE4:  0000000B
442      fp:    0  00FFFE8:  00FFFE0
443 RetAddr:   4  00FFFECC:  000027B4
444                                     =====
445      Args:   8  00FFFE0:  0000000B
446
447 PARAMETERS AND LOCAL VARIABLES WITHIN THIS FRAME:
448 =====
449      a: int
450          8  00FFFE0:  0000000B      value = 11
451      _temp_15
452          -12  00FFFEBC:  0000906C
453      b: int
454          -16  00FFFE8:  0000000C      value = 12
455 =====
456 > up
457 ===== Frame number 2 (where StackTop = 0) =====
458 Function Name:      bar
459 Filename:           HelloWorld.c
460 Execution now at:   line 21
461 Frame Addr:         00FFFEAC
462 frameSize:          12
463 totalParmSize:      4
464                                     =====
465          -20  00FFFE98:  0000000D
466          -16  00FFFE9C:  0000000D
467          -12  00FFFEA0:  0000906C
468 R.D.ptr:  -8  00FFFEA4:  000028D0
469      r13:  -4  00FFFEA8:  00000015
470      fp:    0  00FFFEAC:  00FFFE0
471 RetAddr:   4  00FFFE0:  000028B8
472                                     =====
473      Args:   8  00FFFE4:  0000000C
474
475 PARAMETERS AND LOCAL VARIABLES WITHIN THIS FRAME:
476 =====
477      a: int
478          8  00FFFE4:  0000000C      value = 12
479      _temp_15
480          -12  00FFFEA0:  0000906C
481      b: int
482          -16  00FFFE9C:  0000000D      value = 13
483 =====
484 > up
485 ===== Frame number 1 (where StackTop = 0) =====
486 Function Name:      bar
487 Filename:           HelloWorld.c
488 Execution now at:   line 21
489 Frame Addr:         00FFFE90
490 frameSize:          12
491 totalParmSize:      4
492                                     =====
493          -20  00FFFE7C:  0000000E
494          -16  00FFFE80:  0000000E
495          -12  00FFFE84:  0000906C

```

```

496 R.D.ptr:  -8   00FFFE88:  000028D0
497   r13:    -4   00FFFE8C:  00000015
498   fp:      0   00FFFE90:  00FFFEAC
499 RetAddr:   4   00FFFE94:  000028B8
500                                     =====
501   Args:     8   00FFFE98:  0000000D
502
503 PARAMETERS AND LOCAL VARIABLES WITHIN THIS FRAME:
504 =====
505   a: int
506           8   00FFFE98:  0000000D   value = 13
507   _temp_15
508          -12   00FFFE84:  0000906C
509   b: int
510          -16   00FFFE80:  0000000E   value = 14
511 =====
512 > help
513 =====
514 This program accepts commands typed into the terminal.  Each command
515 should be typed without any arguments; the commands will prompt for
516 arguments when needed.  Case is not significant.  Some abbreviations
517 are allowed, as shown.  Typing control-C will halt execution.
518
519 The available commands are:
520
521 quit      - Terminate this program
522 q
523 help      - Produce this display
524 h
525 info      - Display the current state of the machine
526 i
527 dumpMem   - Display the contents of memory
528 dm
529 setmem     - Used to alter memory contents
530 fmem      - Display floating point values from memory
531 go         - Begin or resume BLITZ instruction execution
532 g
533 step      - Single step; execute one machine-level instruction
534 s
535 t          - Single step; execute one KPL statement
536 u          - Execute continuously until next KPL call, send, or return statement
537 stepn     - Execute N machine-level instructions
538 r          - Display all the integer registers
539 r1         - Change the value of register r1
540 ...
541 r15        - Change the value of register r15
542 float      - Display all the floating-point registers
543 f
544 f0         - Change the value of floating-point register f0
545 ...
546 f15        - Change the value of floating-point register f15
547 dis        - Disassemble several instructions
548 d          - Disassemble several instructions from the current location
549 hex        - Convert a user-entered hex number into decimal and ascii
550 dec        - Convert a user-entered decimal number into hex and ascii
551 ascii      - Convert a user-entered ascii char into hex and decimal
552 setI       - Set the I bit in the Status Register
553 setS       - Set the S bit in the Status Register

```

```

554 setP      - Set the P bit in the Status Register
555 setZ      - Set the Z bit in the Status Register
556 setV      - Set the V bit in the Status Register
557 setN      - Set the N bit in the Status Register
558 clearI    - Clear the I bit in the Status Register
559 clearS    - Clear the S bit in the Status Register
560 clearP    - Clear the P bit in the Status Register
561 clearZ    - Clear the Z bit in the Status Register
562 clearV    - Clear the V bit in the Status Register
563 clearN    - Clear the N bit in the Status Register
564 setPC      - Set the Program Counter (PC)
565 setPTBR   - Set the Page Table Base Register (PTBR)
566 setPTLR   - Set the Page Table Length Register (PTLR)
567 pt        - Display the Page Table
568 trans     - Perform page table translation on a single address
569 cancel    - Cancel all pending interrupts
570 labels    - Display the label table
571 find      - Find a label by name
572 find2     - Find a label by value
573 add       - Add a new label, inserting it into the indexes
574 reset     - Reset the machine state and re-read the a.out file
575 io        - Display the state of the I/O devices
576 read      - Read a word from memory-mapped I/O region
577 write     - Write a word to memory-mapped I/O region
578 raw       - Switch serial input to raw mode
579 cooked    - Switch serial input to cooked mode
580 input     - Enter input characters for future serial I/O input
581 format    - Create and format a BLITZ disk file
582 sim       - Display the current simulation constants
583 stack     - Display the KPL calling stack
584 st        -
585 frame     - Display the current activation frame
586 fr        -
587 up        - Move up in the activation frame stack
588 down     - Move down in the activation frame stack
589
590 =====
591 > step
592 Done! The next instruction to execute will be:
593 0028A8: 8F1F0000      store    r1,[r15+0x0000]      ! decimal: 0 (PowerOnReset)
594 > s
595 Done! The next instruction to execute will be:
596 0028AC: 87D00015      or       r0,0x0015,r13        ! decimal: 21, ascii: ".."
597 > s
598 Done! The next instruction to execute will be:
599 0028B0: 87A04341      or       r0,0x4341,r10        ! decimal: 17217, ascii: "CA"
600 > t
601 About to execute FUNCTION CALL                      in bar (HelloWorld.c, line 21)  time =
602 > t
603 About to execute FUNCTION ENTRY                     in bar (HelloWorld.c, line 14)  time =
604 > t
605 About to execute ASSIGN statement                   in bar (HelloWorld.c, line 16)  time =
606 > i
607 =====
608 Memory size = 0x01000000      ( decimal: 16777216      )

```

```

609 Page size    = 0x00002000      ( decimal: 8192      )
610 .text Segment
611     addr      = 0x00000000      ( decimal: 0      )
612     size      = 0x00004000      ( decimal: 16384   )
613 .data Segment
614     addr      = 0x00004000      ( decimal: 16384   )
615     size      = 0x00006000      ( decimal: 24576   )
616 .bss Segment
617     addr      = 0x0000A000      ( decimal: 40960   )
618     size      = 0x00000000      ( decimal: 0      )
619 ===== USER REGISTERS =====
620     r0  = 0x00000000      ( decimal: 0      )
621     r1  = 0x00000000      ( decimal: 0      )
622     r2  = 0x00000000      ( decimal: 0      )
623     r3  = 0x00000000      ( decimal: 0      )
624     r4  = 0x00000000      ( decimal: 0      )
625     r5  = 0x00000000      ( decimal: 0      )
626     r6  = 0x00000000      ( decimal: 0      )
627     r7  = 0x00000000      ( decimal: 0      )
628     r8  = 0x00000000      ( decimal: 0      )
629     r9  = 0x00000000      ( decimal: 0      )
630     r10 = 0x00000000      ( decimal: 0      )
631     r11 = 0x00000000      ( decimal: 0      )
632     r12 = 0x00000000      ( decimal: 0      )
633     r13 = 0x00000000      ( decimal: 0      )
634     r14 = 0x00000000      ( decimal: 0      )
635     r15 = 0x00000000      ( decimal: 0      )
636 ===== SYSTEM REGISTERS =====
637     r0  = 0x00000000      ( decimal: 0      )
638     r1  = 0x00000000      ( decimal: 0      )
639     r2  = 0x0000000A      ( decimal: 10      )
640     r3  = 0x00000012      ( decimal: 18      )
641     r4  = 0x8CC97375      ( decimal: -1932954763 )
642     r5  = 0x00000000      ( decimal: 0      )
643     r6  = 0x00000000      ( decimal: 0      )
644     r7  = 0x00000000      ( decimal: 0      )
645     r8  = 0x00000000      ( decimal: 0      )
646     r9  = 0x00000000      ( decimal: 0      )
647     r10 = 0x00004153      ( decimal: 16723    )
648     r11 = 0x00000000      ( decimal: 0      )
649     r12 = 0x00000000      ( decimal: 0      )
650     r13 = 0x00000010      ( decimal: 16      )
651     r14 = 0x00FFFE58      ( decimal: 16776792 )
652     r15 = 0x00FFFE44      ( decimal: 16776772 )
653 ===== FLOATING-POINT REGISTERS =====
654     f0  = 0x00000000 00000000 ( value = 0      )
655     f1  = 0x00000000 00000000 ( value = 0      )
656     f2  = 0x00000000 00000000 ( value = 0      )
657     f3  = 0x00000000 00000000 ( value = 0      )
658     f4  = 0x00000000 00000000 ( value = 0      )
659     f5  = 0x00000000 00000000 ( value = 0      )
660     f6  = 0x00000000 00000000 ( value = 0      )
661     f7  = 0x00000000 00000000 ( value = 0      )
662     f8  = 0x00000000 00000000 ( value = 0      )
663     f9  = 0x00000000 00000000 ( value = 0      )
664     f10 = 0x00000000 00000000 ( value = 0      )
665     f11 = 0x00000000 00000000 ( value = 0      )
666     f12 = 0x00000000 00000000 ( value = 0      )

```

HardwareFault )

```

667 f13 = 0x00000000 00000000 ( value = 0 )
668 f14 = 0x00000000 00000000 ( value = 0 )
669 f15 = 0x00000000 00000000 ( value = 0 )
670 =====
671 PC = 0x00002844 ( decimal: 10308 )
672 PTBR = 0x00000000 ( decimal: 0 )
673 PTLR = 0x00000000 ( decimal: 0 )
674 -----
675 SR = 0x00000010 = 0000 0000 0000 0000 0000 0000 0001 0000 --IS PZVN
676 I = 0 Interrupts Disabled
677 S = 1 System Mode
678 P = 0 Paging Disabled
679 Z = 0 Not Zero
680 V = 0 No Overflow
681 N = 0 Not Negative
682 =====
683 Pending Interrupts = 0x00000002
684 TIMER_INTERRUPT
685 System Trap Number = 0x00000000
686 Page Invalid Offending Address = 0x00000000
687 Page Readonly Offending Address = 0x00000000
688 Time of next timer event = 5005
689 Time of next disk event = 2147483647
690 Time of next serial in event = 30039
691 Time of next serial out event = 2147483647
692 Current Time = 889
693 Time of next event = 5005
694 Time Spent Sleeping = 0
695 Instructions Executed = 889
696 Number of Disk Reads = 0
697 Number of Disk Writes = 0
698 =====
699 The next instruction to execute will be:
700 002844: 8B1E0008 load [r14+0x0008],r1 ! decimal: 8 (DiskInterrupt)
701 About to execute ASSIGN statement in bar (HelloWorld.c, line 16) time =
889
702 > reset
703 Resetting all CPU registers and re-reading file "HelloWorld"...
704 > g
705 Beginning execution...
706 ===== KPL PROGRAM STARTING =====
707
708 Hello, world...
709
710 The value of b is 12
711
712
713
714 **** A 'debug' instruction was encountered ****
715
716 Done! The next instruction to execute will be:
717 0028A4: 8B1EFFF0 load [r14+0xFFFF0],r1 ! decimal: -16
718 > stack
719 Function/Method Frame Addr Execution at...
720 =====
721 bar 00FFFE08 HelloWorld.c, line 20
722 foo 00FFFE00 HelloWorld.c, line 11
723 main 00FFFEF8 HelloWorld.c, line 7

```

```

724 Bottom of activation frame stack!
725 > fr
726 ===== Frame number 0 (where StackTop = 0) =====
727 Function Name:      bar
728 Filename:           HelloWorld.c
729 Execution now at:   line 20
730 Frame Addr:         00FFFE08
731 frameSize:         12
732 totalParmSize:      4
733
734      sp--> -20    00FFFE04:  0000000C
735           -16    00FFFE08:  0000000C
736           -12    00FFFE0C:  0000906C
737 R.D.ptr:  -8     00FFFE00:  000028D0
738      r13:  -4     00FFFE04:  0000000B
739      fp:    0     00FFFE08:  00FFFE00
740 RetAddr:   4     00FFFE0C:  000027B4
741
742      Args:    8     00FFFE00:  0000000B
743
744 PARAMETERS AND LOCAL VARIABLES WITHIN THIS FRAME:
745 =====
746  a: int
747      8     00FFFE00:  0000000B      value = 11
748  _temp_15
749      -12    00FFFE0C:  0000906C
750  b: int
751      -16    00FFFE08:  0000000C      value = 12
752 =====
753 > down
754 ===== Frame number 1 (where StackTop = 0) =====
755 Function Name:      foo
756 Filename:           HelloWorld.c
757 Execution now at:   line 11
758 Frame Addr:         00FFFE00
759 frameSize:         8
760 totalParmSize:      4
761
762      -16    00FFFE00:  0000000B
763      -12    00FFFE04:  0000000B
764 R.D.ptr:  -8     00FFFE08:  000027CC
765      r13:  -4     00FFFE0C:  00000007
766      fp:    0     00FFFE00:  00FFFE08
767 RetAddr:   4     00FFFE04:  00002710
768
769      Args:    8     00FFFE08:  0000000A
770
771 PARAMETERS AND LOCAL VARIABLES WITHIN THIS FRAME:
772 =====
773  x: int
774      8     00FFFE08:  0000000A      value = 10
775  _temp_11
776      -12    00FFFE04:  0000000B
777 =====
778 > up
779 ===== Frame number 0 (where StackTop = 0) =====
780 Function Name:      bar
781 Filename:           HelloWorld.c

```

```

782 Execution now at: line 20
783 Frame Addr:      00FFFE08
784 frameSize:      12
785 totalParmSize:   4
786
787      sp--> -20    00FFFE04: 0000000C
788           -16    00FFFE08: 0000000C
789           -12    00FFFE0C: 0000906C
790 R.D.ptr:  -8     00FFFE00: 000028D0
791      r13:  -4     00FFFE04: 0000000B
792      fp:    0     00FFFE08: 00FFFE00
793 RetAddr:   4     00FFFE0C: 000027B4
794
795      Args:   8     00FFFE00: 0000000B
796
797 PARAMETERS AND LOCAL VARIABLES WITHIN THIS FRAME:
798 =====
799      a: int
800           8     00FFFE00: 0000000B      value = 11
801      _temp_15
802          -12    00FFFE0C: 0000906C
803      b: int
804          -16    00FFFE08: 0000000C      value = 12
805 =====
806 > q
807 Number of Disk Reads      = 0
808 Number of Disk Writes    = 0
809 Instructions Executed     = 609
810 Time Spent Sleeping      = 0
811      Total Elapsed Time   = 609
812 bash-3.2$ format
813 bash: format: command not found
814 bash-3.2$ formatblitz -g HelloWorld
815 Beginning execution...
816 ===== KPL PROGRAM STARTING =====
817
818 Hello, world...
819
820 The value of b is 12
821
822
823
824 **** A 'debug' instruction was encountered ****
825
826 Done! The next instruction to execute will be:
827 0028A4: 8B1EFFF0      load      [r14+0xFFF0],r1      ! decimal: -16
828
829
830 Entering machine-level debugger...
831
832 =====
833 =====
834 ===== The BLITZ Machine Emulator =====
835 =====
836 ===== Copyright 2001-2007, Harry H. Porter III =====
837 =====
838 =====
839

```

```

840 Enter a command at the prompt. Type 'quit' to exit or 'help' for
841 info about commands.
842 > format
843 =====
844 This command is used to create or modify a file to be used by the BLITZ emulator
845 for the disk. By default, this file will be called "DISK". The filename may
846 be specified on the emulator command line with the "-d filename" option. This
847 command will create the file if it does not exist. It will set the file to the
848 desired size and initialize all newly allocated space.
849
850 The size of the disk file is an integral number of tracks. Each track will
851 contain 16 sectors. The size of each sector is the same as the page size.
852 Thus, the sector size is 8192 bytes. The actual file size will be the number
853 tracks times the number of sectors per track times the sector size, plus an
854 additional 4 bytes, which will contain a "magic number". The magic number
855 is 0x424C5A64 (decimal: 1112300132, ASCII: "BLZd") and is used to identify this
856 file as a BLITZ disk file.
857
858 Initialization consists of writing the magic number in the first 4 bytes of the
859 file and adjusting the file length. Any data previously stored in the file will
860 be preserved and any additional sectors created will be initialized.
861 =====
862
863 The name of the disk file is "DISK".
864 The file "DISK" existed previously.
865     Old File Length = 131076 bytes
866     Old Sector Count = 16
867     Old Track Count = 1
868 Enter the number of tracks (e.g., 1000; type 0 to abort):
869 j q
870 Aborting; file not changed!
871 > q
872 Number of Disk Reads      = 0
873 Number of Disk Writes    = 0
874 Instructions Executed     = 609
875 Time Spent Sleeping      = 0
876     Total Elapsed Time   = 609
877 bash-3.2$ exit
878
879 Script done on Sun Apr  6 14:47:33 2014

```