

ECE550 Post-silicon Validation: Project 2 Part 3 Report

-Abhishek Musku, Sai Teja Gali, Sandeep Goud Abbagouni, Ramakrishna Gopavarapu

Multiprocessor Instructions and CPU Management: Implemented atomic XADD and XCHG instructions with randomized LOCK prefixes for multicore synchronization, along with memory fence instructions (MFENCE, SFENCE, LFENCE) for controlling memory ordering. Additionally, implemented CPU binding functionality using sched_setaffinity() to pin each forked process to specific CPU cores, enabling effective validation of cache coherency and inter-core communication during instruction execution.

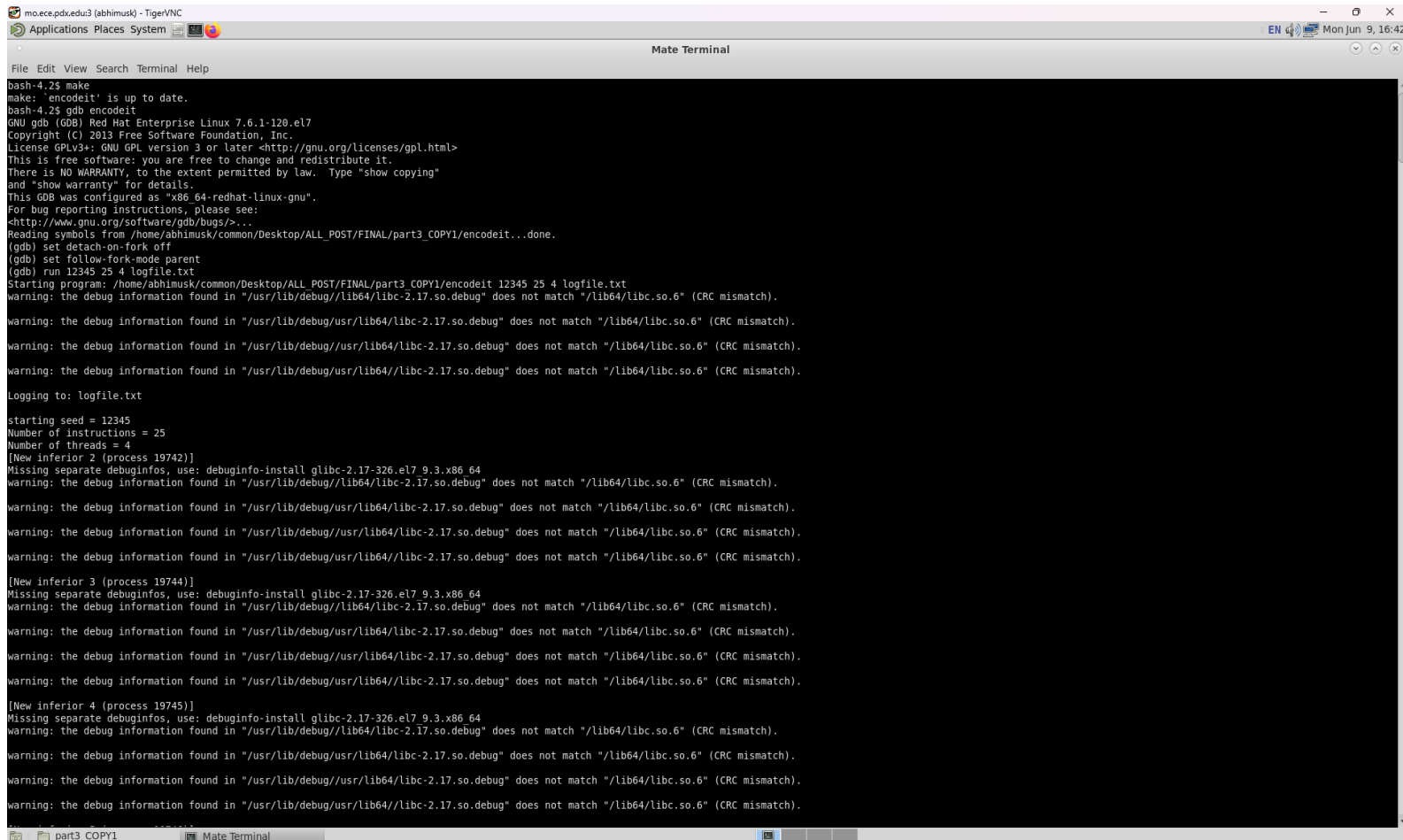
Command Line Arguments: The program accepts four arguments for test configuration: seed value, number of instructions, number of threads, and optional log filename. The code validates thread count against MAX_THREADS (4) and handles log file creation with error checking.

Usage Format:

`./encodeit <seed> <num_instructions> <num_threads> [logfile]`

Step 1: Compiled the code with make, launched it in GDB debugger, and executed it with arguments 12345 25 4 logfile.txt to generate 25 instructions across 4 threads with seed 12345.

run 12345 25 4 logfile.txt



```
mo.ece.pdx.edu3 (abhimusk) - TigerVNC
Applications Places System
Mate Terminal
File Edit View Search Terminal Help
bash-4.2$ make
make: 'encodeit' is up to date.
bash-4.2$ gdb encodeit
GNU gdb (GDB) Red Hat Enterprise Linux 7.6.1-120.el7
Copyright (C) 2013 Free Software Foundation, Inc.
License GPLv3+: GNU GPL version 3 or later <http://gnu.org/licenses/gpl.html>
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law. Type "show copying"
and "show warranty" for details.
This GDB was configured as "x86_64-redhat-linux-gnu".
For bug reporting instructions, please see:
<http://www.gnu.org/software/gdb/bugs/>...
Reading symbols from /home/abhimusk/common/Desktop/ALL_POST/FINAL/part3_COPY1/encodeit...done.
(gdb) set detach-on-fork off
(gdb) set follow-fork-mode parent
(gdb) run 12345 25 4 logfile.txt
Starting program: /home/abhimusk/common/Desktop/ALL_POST/FINAL/part3_COPY1/encodeit 12345 25 4 logfile.txt
warning: the debug information found in "/usr/lib/debug//lib64/libc-2.17.so.debug" does not match "/lib64/libc.so.6" (CRC mismatch).

warning: the debug information found in "/usr/lib/debug/usr/lib64/libc-2.17.so.debug" does not match "/lib64/libc.so.6" (CRC mismatch).
warning: the debug information found in "/usr/lib/debug/usr/lib64/libc-2.17.so.debug" does not match "/lib64/libc.so.6" (CRC mismatch).
warning: the debug information found in "/usr/lib/debug/usr/lib64/libc-2.17.so.debug" does not match "/lib64/libc.so.6" (CRC mismatch).
Logging to: logfile.txt

starting seed = 12345
Number of instructions = 25
Number of threads = 4
[New inferior 2 (process 19742)]
Missing separate debuginfos, use: debuginfo-install glibc-2.17-326.el7 9.3.x86_64
warning: the debug information found in "/usr/lib/debug//lib64/libc-2.17.so.debug" does not match "/lib64/libc.so.6" (CRC mismatch).

warning: the debug information found in "/usr/lib/debug/usr/lib64/libc-2.17.so.debug" does not match "/lib64/libc.so.6" (CRC mismatch).
warning: the debug information found in "/usr/lib/debug/usr/lib64/libc-2.17.so.debug" does not match "/lib64/libc.so.6" (CRC mismatch).
warning: the debug information found in "/usr/lib/debug/usr/lib64/libc-2.17.so.debug" does not match "/lib64/libc.so.6" (CRC mismatch).
[New inferior 3 (process 19744)]
Missing separate debuginfos, use: debuginfo-install glibc-2.17-326.el7 9.3.x86_64
warning: the debug information found in "/usr/lib/debug//lib64/libc-2.17.so.debug" does not match "/lib64/libc.so.6" (CRC mismatch).

warning: the debug information found in "/usr/lib/debug/usr/lib64/libc-2.17.so.debug" does not match "/lib64/libc.so.6" (CRC mismatch).
warning: the debug information found in "/usr/lib/debug/usr/lib64/libc-2.17.so.debug" does not match "/lib64/libc.so.6" (CRC mismatch).
warning: the debug information found in "/usr/lib/debug/usr/lib64/libc-2.17.so.debug" does not match "/lib64/libc.so.6" (CRC mismatch).
[New inferior 4 (process 19745)]
Missing separate debuginfos, use: debuginfo-install glibc-2.17-326.el7 9.3.x86_64
warning: the debug information found in "/usr/lib/debug//lib64/libc-2.17.so.debug" does not match "/lib64/libc.so.6" (CRC mismatch).

warning: the debug information found in "/usr/lib/debug/usr/lib64/libc-2.17.so.debug" does not match "/lib64/libc.so.6" (CRC mismatch).
warning: the debug information found in "/usr/lib/debug/usr/lib64/libc-2.17.so.debug" does not match "/lib64/libc.so.6" (CRC mismatch).
warning: the debug information found in "/usr/lib/debug/usr/lib64/libc-2.17.so.debug" does not match "/lib64/libc.so.6" (CRC mismatch).
```

Step 2: Set breakpoints at build_instructions and executeit functions, continued execution

```
mo.ece.pdx.edu3 (abhimusk) - TigerVNC
Applications Places System
Mate Terminal
File Edit View Search Terminal Help
warning: the debug information found in "/usr/lib/debug/usr/lib64/libc-2.17.so.debug" does not match "/lib64/libc.so.6" (CRC mismatch).
warning: the debug information found in "/usr/lib/debug/usr/lib64//libc-2.17.so.debug" does not match "/lib64/libc.so.6" (CRC mismatch).
^C
Program received signal SIGINT, Interrupt.
0x00007ffff7d060c in waitpid () from /lib64/libc.so.6
Missing separate debuginfos, use: debuginfo-install glibc-2.17-326.el7_9.3.x86_64
(gdb) inferior 2
[Switching to inferior 2 (process 19742)] (/home/abhimusk/common/Desktop/ALL_POST/FINAL/part3_COPY1/encodeit)
[Switching to thread 2 (process 19742)]
#0 0x00007ffff7d0b12 in fork () from /lib64/libc.so.6
(gdb) break build_instructions
Breakpoint 1 at 0x4025a2: build_instructions. (5 locations)
(gdb) break executeit
Breakpoint 2 at 0x4021e8: executeit. (5 locations)
(gdb) continue
Continuing.

Breakpoint 1, build_instructions (next_ptr=0x7ffff7fe8000 "", thread_id=0, logfile=0x606010) at encodeit.c:398
398     int instructions_built = 0;
(gdb) continue
Continuing.
T0: building instructions
T0: MOVING MDPTR: MOV #7FFF7FAC000->R6 (size=8)
T0: Setup: loaded mdptr into RSI
T0: Generating: XCHG R7,R2 (size=1)
ISZ 1 reg-to-reg XCHG: R7 <=> R2
T0: Instruction 2 complete, next_ptr: 0x7ffff7fe8019
T0: Generating: XCHG [RSI+53],R0 (size=4)
ISZ 4 mem-to-reg XCHG: [R6+53] <=> R0
    + 8-bit displacement: 0x35
T0: Instruction 3 complete, next_ptr: 0x7ffff7fe801c
T0: Generating: MOV [RSI+35]->R10 (size=4)
ISZ 4 mem-to-reg: stored at 0x7ffff7fe801d: 0x560b, disp=35
    + 8-bit displacement: 0x23
T0: Instruction 4 complete, next_ptr: 0x7ffff7fe8020
T0: Generating: LOCK XADD [RSI+0],R7 (size=4)
LOCK ISZ 4 mem-to-reg XADD: [R6+0] += R7 (and exchange)
T0: Instruction 5 complete, next_ptr: 0x7ffff7fe8024
T0: Generating: XCHG R2,R0 (size=4)
ISZ 4 reg-to-reg XCHG: R2 <=> R0
T0: Instruction 6 complete, next_ptr: 0x7ffff7fe8027
T0: Generating: SFENCE (store memory barrier)
Generated SFENCE (store memory barrier)
T0: Instruction 7 complete, next_ptr: 0x7ffff7fe802a
T0: Generating: MOV R10->[RSI+11] (size=8)
ISZ 8 reg-to-mem: stored at 0x7ffff7fe802b: 0x5609, disp=11
    + 8-bit displacement: 0x0b
T0: Instruction 8 complete, next_ptr: 0x7ffff7fe802e
T0: Generating: MFENCE (full memory barrier)
Generated MFENCE (full memory barrier)
T0: Instruction 9 complete, next_ptr: 0x7ffff7fe8031
T0: Generating: LFENCE (load memory barrier)
Generated LFENCE (load memory barrier)
T0: Instruction 10 complete, next_ptr: 0x7ffff7fe8034
T0: Generating: LOCK XADD [RSI+71],R2 (size=4)
LOCK ISZ 4 mem-to-reg XADD: [R6+71] += R2 (and exchange)
    + 8-bit displacement: 0x47
T0: Instruction 11 complete, next_ptr: 0x7ffff7fe8039
```

Step 3: Hit the executeit breakpoint and used GDB's x/i command to examine and disassemble the generated machine code for thread 0

```
moece.pdx.edu:3 (abhimusk) - Tiger/VNC
Applications Places System
Mate Terminal
File Edit View Search Terminal Help
T0: Instruction 25 complete, next ptr: 0x7ffff7fe8070
T0: Generating: MOV [RSI+16]->R3 (size=4)
ISZ 4 mem-to-reg: stored at 0x7ffff7fe8070: 0x5e8b, disp=16
+ 8-bit displacement: 0x10
T0: Instruction 26 complete, next ptr: 0x7ffff7fe8073
T0: next ptr is now 0x7ffff7fe8073
T0: Generated 26 total instructions

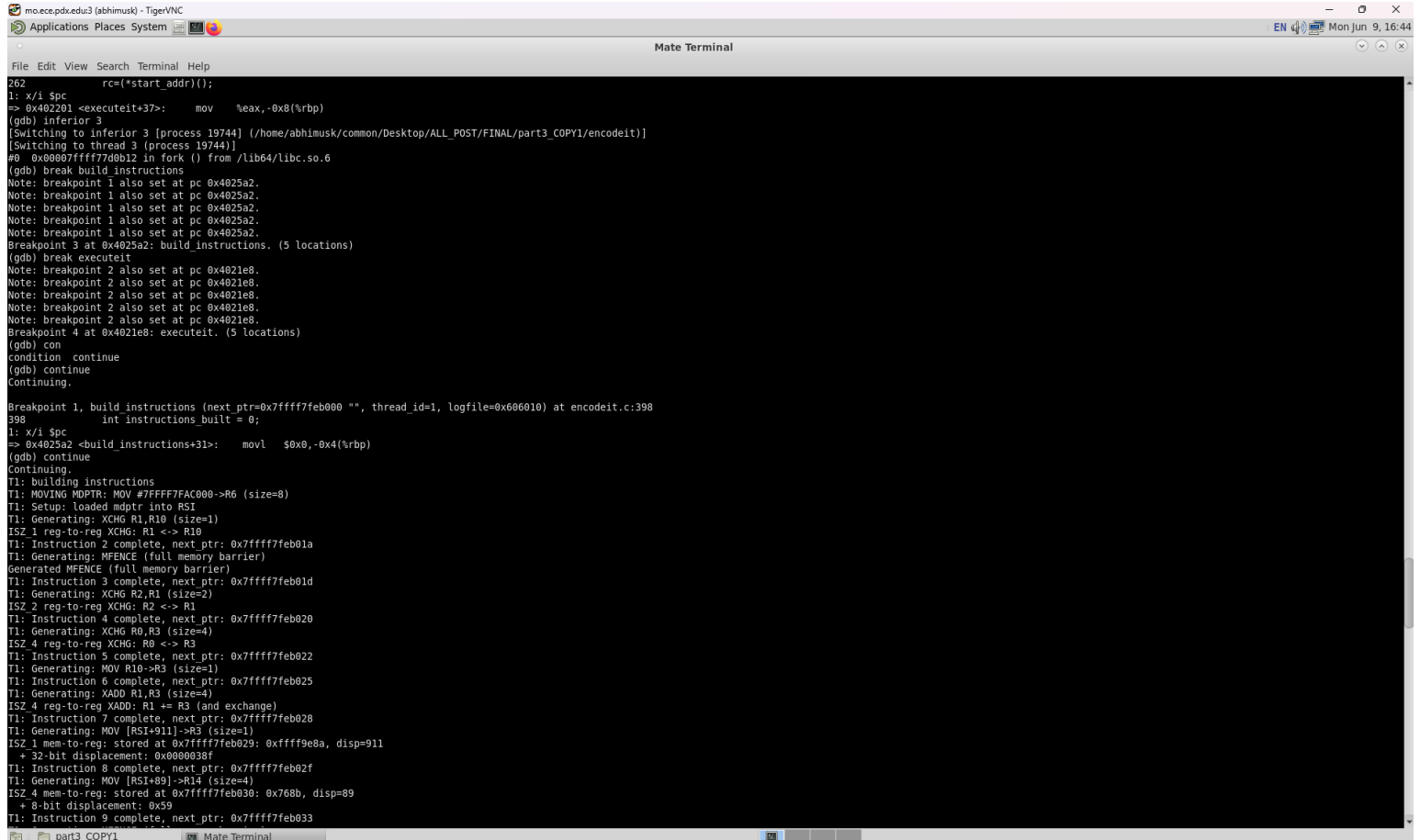
Breakpoint 2, executeit (start_addr=0x7ffff7fe8000) at encodeit.c:258
258 volatile int i,rc=0;
(gdb) x/40i mptr threads[0]
0x7ffff7fe8000: enterq $0x800,$0x0
0x7ffff7fe8004: push %rbx
0x7ffff7fe8005: push %r12
0x7ffff7fe8007: push %r13
0x7ffff7fe8009: push %r14
0x7ffff7fe800b: push %r15
0x7ffff7fe800d: movabs $0x7ffff7fac000,%rsi
0x7ffff7fe8017: xchg %dl,%bh
0x7ffff7fe8019: xchg %eax,0x35(%rsi)
0x7ffff7fe801c: mov 0x23(%rsi),%r10d
0x7ffff7fe8020: lock xadd %edi,(%rsi)
0x7ffff7fe8024: xchg %r9d,%edx
0x7ffff7fe8027: sfence
0x7ffff7fe802a: mov %r10,0xb(%rsi)
0x7ffff7fe802e: mfence
0x7ffff7fe8031: lfence
0x7ffff7fe8034: lock xadd %edx,0x47(%rsi)
0x7ffff7fe8039: sfence
0x7ffff7fe803c: mov (%rsi),%dx
0x7ffff7fe803f: lock xchg %r14b,(%rsi)
0x7ffff7fe8043: lfence
0x7ffff7fe8046: mov %r15b,0x34(%rsi)
0x7ffff7fe804a: mov (%rsi),%r9b
0x7ffff7fe804d: lock xadd %r10d,(%rsi)
0x7ffff7fe8052: xadd %r14d,%r9d
0x7ffff7fe8056: mov %eax,%r11d
0x7ffff7fe8059: lock xchg %dl,0x13c(%rsi)
0x7ffff7fe8060: xchg %r10d,%r9d
0x7ffff7fe8063: mfence
0x7ffff7fe8066: mov %rcx,%r15
0x7ffff7fe8069: xchg %r9b,0x44a(%rsi)
0x7ffff7fe8070: mov 0x10(%rsi),%ebx
0x7ffff7fe8073: pop %r15
0x7ffff7fe8075: pop %r14
0x7ffff7fe8077: pop %r13
0x7ffff7fe8079: pop %r12
0x7ffff7fe807b: pop %rbx
0x7ffff7fe807c: leaveq
0x7ffff7fe807d: retq
0x7ffff7fe807e: add %al,(%rax)
(gdb) display/i $pc
1: x/i $pc
=> 0x402108 <executeit+12>: movl $0x0,-0x8(%rbp)
(gdb) stepi
260 i=0;
1: x/i $pc
=> 0x4021ef <executeit+19>: movl $0x0,-0x4(%rbp)
(gdb)
262 rc=(*start_addr)();
```

Step 4: Executed the generated instructions step-by-step using GDB's stepi command, observing the actual execution of the instruction sequence including function prologue (ENTER, PUSH operations), data movement (MOV), atomic operations (XCHG, LOCK XADD), and verifying that each generated instruction executed correctly in the processor.

```
mo.ece.pdx.edu3 (abhimusk) - TigerVNC
Applications Places System
Mate Terminal
File Edit View Search Terminal Help
(gdb) stepi
260      i=0;
1: x/i $pc
=> 0x4021ef <executeit+19>:    movl    $0x0,-0x4(%rbp)
(gdb)
262      rc=(*start_addr)();
1: x/i $pc
=> 0x4021f6 <executeit+26>:    mov     -0x18(%rbp),%rdx
(gdb) stepi
0x00000000004021fa 262      rc=(*start_addr)();
1: x/i $pc
=> 0x4021fa <executeit+30>:    mov     $0x0,%eax
(gdb) stepi
0x00000000004021ff 262      rc=(*start_addr)();
1: x/i $pc
=> 0x4021ff <executeit+35>:    callq  %rdx
(gdb) stepi
0x00007ffff7fe8000 in ?? ()
1: x/i $pc
=> 0x7ffff7fe8000:    enterq $0x800,$0x0
(gdb) stepi
0x00007ffff7fe8004 in ?? ()
1: x/i $pc
=> 0x7ffff7fe8004:    push   %rbx
(gdb) stepi
0x00007ffff7fe8005 in ?? ()
1: x/i $pc
=> 0x7ffff7fe8005:    push   %r12
(gdb) stepi
0x00007ffff7fe8007 in ?? ()
1: x/i $pc
=> 0x7ffff7fe8007:    push   %r13
(gdb) stepi
0x00007ffff7fe8009 in ?? ()
1: x/i $pc
=> 0x7ffff7fe8009:    push   %r14
(gdb) stepi
0x00007ffff7fe800b in ?? ()
1: x/i $pc
=> 0x7ffff7fe800b:    push   %r15
(gdb) stepi
0x00007ffff7fe800d in ?? ()
1: x/i $pc
=> 0x7ffff7fe800d:    movabs $0x7ffff7fac000,%rsi
(gdb) stepi
0x00007ffff7fe8017 in ?? ()
1: x/i $pc
=> 0x7ffff7fe8017:    xchg   %dl,%bh
(gdb) stepi
0x00007ffff7fe8019 in ?? ()
1: x/i $pc
=> 0x7ffff7fe8019:    xchg   %eax,0x35(%rsi)
(gdb) stepi
0x00007ffff7fe801c in ?? ()
1: x/i $pc
=> 0x7ffff7fe801c:    mov     0x23(%rsi),%r10d
(gdb) stepi
0x00007ffff7fe8020 in ?? ()
1: x/i $pc
=> 0x7ffff7fe8020:    lock xadd %edi,(%rsi)
```

```
mo.ece.pdx.edu:3 (abhimusk) - TigerVNC
Applications Places System
Mate Terminal
File Edit View Search Terminal Help
(gdb) stepi
0x00007ffff7fe0059 in ?? ()
1: x/i $pc
=> 0x7ffff7fe0059:    lock xchg %d1,0x13c(%rsi)
(gdb) stepi
0x00007ffff7fe0060 in ?? ()
1: x/i $pc
=> 0x7ffff7fe0060:    xchg    %r10d,%r9d
(gdb) stepi
0x00007ffff7fe0063 in ?? ()
1: x/i $pc
=> 0x7ffff7fe0063:    mfence
(gdb) stepi
0x00007ffff7fe0066 in ?? ()
1: x/i $pc
=> 0x7ffff7fe0066:    mov     %rcx,%r15
(gdb) stepi
0x00007ffff7fe0069 in ?? ()
1: x/i $pc
=> 0x7ffff7fe0069:    xchg    %r9b,0x44a(%rsi)
(gdb) stepi
0x00007ffff7fe0070 in ?? ()
1: x/i $pc
=> 0x7ffff7fe0070:    mov     0x10(%rsi),%ebx
(gdb) stepi
0x00007ffff7fe0073 in ?? ()
1: x/i $pc
=> 0x7ffff7fe0073:    pop     %r15
(gdb) stepi
0x00007ffff7fe0075 in ?? ()
1: x/i $pc
=> 0x7ffff7fe0075:    pop     %r14
(gdb) stepi
0x00007ffff7fe0077 in ?? ()
1: x/i $pc
=> 0x7ffff7fe0077:    pop     %r13
(gdb) stepi
0x00007ffff7fe0079 in ?? ()
1: x/i $pc
=> 0x7ffff7fe0079:    pop     %r12
(gdb) stepi
0x00007ffff7fe007b in ?? ()
1: x/i $pc
=> 0x7ffff7fe007b:    pop     %rbx
(gdb) stepi
0x00007ffff7fe007c in ?? ()
1: x/i $pc
=> 0x7ffff7fe007c:    leaveq
(gdb) stepi
0x00007ffff7fe007d in ?? ()
1: x/i $pc
=> 0x7ffff7fe007d:    retq
(gdb) stepi
0x0000000000402201 in executeit (start_addr=0x7ffff7fe0000) at encodeit.c:262
262      rc=(*start_addr)();
1: x/i $pc
=> 0x402201 <executeit+37>:    mov     %eax,-0x8(%rbp)
(gdb) inferior 3
[Switching to inferior 3 (process 19744)] (/home/abhimusk/common/Desktop/ALL_POST/FINAL/part3_COPY1/encodeit)
[Switching to thread 3 (process 19744)]
part3_COPY1  Mate Terminal
```

Step 5: Switched to thread 1 execution, where GDB shows breakpoints set for all threads and continues generating instructions for thread 1, demonstrating successful multiprocessor instruction generation with each thread producing its own independent sequence of MOV, XCHG, MFENCE, XADD operations and memory operations with different register assignments and displacements.



```
mo.ece.pdx.edu3 (abhimusk) - TigerVNC
Applications Places System
Mate Terminal

File Edit View Search Terminal Help

262 rc=(*start_addr)();
1: x/i $pc
=> 0x402201 <executeit+37>: mov    %eax, -0x8(%rbp)
(gdb) inferior 3
[Switching to inferior 3 (process 19744)] (/home/abhimusk/common/Desktop/ALL_POST/FINAL/part3_COPY1/encodeit)]
[Switching to thread 3 (process 19744)]
#0 0x0007ffff7d0b12 in fork () from /lib64/libc.so.6
(gdb) break build_instructions
Note: breakpoint 1 also set at pc 0x4025a2.
Note: breakpoint 1 also set at pc 0x4025a2.
Note: breakpoint 1 also set at pc 0x4025a2.
Note: breakpoint 1 also set at pc 0x4025a2.
Note: breakpoint 1 also set at pc 0x4025a2.
Breakpoint 3 at 0x4025a2: build_instructions. (5 locations)
(gdb) break executeit
Note: breakpoint 2 also set at pc 0x4021e8.
Note: breakpoint 2 also set at pc 0x4021e8.
Note: breakpoint 2 also set at pc 0x4021e8.
Note: breakpoint 2 also set at pc 0x4021e8.
Note: breakpoint 2 also set at pc 0x4021e8.
Breakpoint 4 at 0x4021e8: executeit. (5 locations)
(gdb) con
condition continue
(gdb) continue
Continuing.

Breakpoint 1, build_instructions (next_ptr=0x7ffff7feb000 "", thread_id=1, logfile=0x000010) at encodeit.c:398
398     int instructions_built = 0;
1: x/i $pc
=> 0x4025a2 <build_instructions+31>: movl   $0x0, -0x4(%rbp)
(gdb) continue
Continuing.

T1: building instructions
T1: MOVING MDPTR: MOV #7FFFF7AC000->R6 (size=8)
T1: Setup: loaded mdptr into R51
T1: Generating: XCHG R1,R10 (size=1)
ISZ 1 reg-to-reg XCHG: R1 <=> R10
T1: Instruction 2 complete, next_ptr: 0x7ffff7feb01a
T1: Generating: MFENCE (full memory barrier)
Generated MFENCE (full memory barrier)
T1: Instruction 3 complete, next_ptr: 0x7ffff7feb01d
T1: Generating: XCHG R2,R1 (size=2)
ISZ 2 reg-to-reg XCHG: R2 <=> R1
T1: Instruction 4 complete, next_ptr: 0x7ffff7feb020
T1: Generating: XCHG R0,R3 (size=4)
ISZ 4 reg-to-reg XCHG: R0 <=> R3
T1: Instruction 5 complete, next_ptr: 0x7ffff7feb022
T1: Generating: MOV R10->R3 (size=1)
T1: Instruction 6 complete, next_ptr: 0x7ffff7feb025
T1: Generating: XADD R1,R3 (size=4)
ISZ 4 reg-to-reg XADD: R1 += R3 (and exchange)
T1: Instruction 7 complete, next_ptr: 0x7ffff7feb028
T1: Generating: MOV [RSI+911]->R3 (size=1)
ISZ 1 mem-to-reg: stored at 0x7ffff7feb029: 0xffff9e8a, disp=911
+ 32-bit displacement: 0x0000038f
T1: Instruction 8 complete, next_ptr: 0x7ffff7feb02f
T1: Generating: MOV [RSI+89]->R14 (size=4)
ISZ 4 mem-to-reg: stored at 0x7ffff7feb030: 0x768b, disp=89
+ 8-bit displacement: 0x59
T1: Instruction 9 complete, next_ptr: 0x7ffff7feb033
```

Conclusion: Successfully implemented a multiprocessor x86-64 instruction validation framework with atomic operations (XADD/XCHG with LOCK prefixes), memory fence instructions, multithread support via fork(), and CPU binding capabilities. GDB debugging sessions validated correct instruction generation and execution across multiple CPU cores, providing an effective foundation for post-silicon validation of cache coherency and inter-core communication protocols.