

Packed SIMD Compiler Build

There has been P-ext compiler support developed by a couple of labs - namely PLCT Lab and NTHU's Programming Language Lab (PLLAB). PLCT has developed the most recent support, but they have them only downstream since P-ext is not yet rectified and may undergo refactoring by the task group.

Requirements

In order to build the compiler, the repositories that we require are:

- RISC-V GNU Compiler Toolchain: <https://github.com/riscv-collab/riscv-gnu-toolchain>
- gcc: <https://github.com/plctlab/riscv-gcc/tree/riscv-gcc-p-ext>
- binutils: <https://github.com/plctlab/riscv-binutils-gdb/tree/riscv-binutils-p-ext>

Now, follow the steps for a successful build to compile the instructions.

Procedure

Clone the toolchain -

```
git clone https://github.com/riscv-collab/riscv-gnu-toolchain
cd riscv-gnu-toolchain
```

Create a .gitmodules.pext file(modified .gitmodules for pext) with the following content:

```
[submodule "binutils"]
  path = binutils
  url = https://github.com/plctlab/riscv-binutils-gdb
  branch = riscv-binutils-p-ext
[submodule "gcc"]
  path = gcc
  url = https://github.com/plctlab/riscv-gcc
  branch = riscv-gcc-p-ext
[submodule "glibc"]
  path = glibc
```

```

url = https://sourceware.org/git/glibc.git
[submodule "dejagnu"]
  path = dejagnu
  url = https://git.savannah.gnu.org/git/dejagnu.git
  branch = dejagnu-1.6.3
[submodule "newlib"]
  path = newlib
  url = https://sourceware.org/git/newlib-cygwin.git
  branch = master
[submodule "gdb"]
  path = gdb
  url = https://sourceware.org/git/binutils-gdb.git
  branch = gdb-12-branch
[submodule "qemu"]
  path = qemu
  url = https://gitlab.com/qemu-project/qemu.git
[submodule "musl"]
  path = musl
  url = https://git.musl-libc.org/git/musl
  branch = master
[submodule "spike"]
  path = spike
  url = https://github.com/riscv-software-src/riscv-isa-sim.git
  branch = master
[submodule "pk"]
  path = pk
  url = https://github.com/riscv-software-src/riscv-pk.git
  branch = master

```

This helps in swapping the submodules of GCC and binutils in riscv-gnu-toolchain with our desired fork and branch. To reflect the changes, run-

```

cp ../.gitmodules.pext .gitmodules
git submodule sync
git submodule update --init --recursive --remote
./configure --prefix=`pwd`/installed-tools --with-arch=rv64imafd_zpn --with-abi=lp64d
sudo make

```

It should build successfully after these commands.

Testing

To test, copy the following assembly code to psimd.S with P-ext's radd instruction.

```

_start:
andi t0, t0, 0 # Clear register t0

```

```

andi t1, t1, 0 # Clear register t1
andi t2, t2, 0 # Clear register t2
andi t3, t3, 0 # Clear register t3
andi t4, t4, 0 # Clear register t3
li t0, 394 # Load register t0
li t1, 19 # Load register t1
#Muldiv
mul t2, t0, t1
#FP single and Double precision
fcvt.s.w ft0, t2
fcvt.d.w ft1, t1
fdiv.d ft2, ft1, ft0
fadd.s ft3, ft0, ft2
fcvt.w.s t4, ft3
#Psimd
radd16 t3, t2, t4

```

Run the following now, to compile and disassemble-

```

riscv64-unknown-linux-gnu-gcc -march=rv64imafd_zpn -c psimd.S
riscv64-unknown-linux-gnu-objdump -d psimd.o

```

Expected output-

```

psimd.o:      file format elf64-littleriscv

Disassembly of section .text:

0000000000000000 <start>:
   0: 0002f293      andi  t0,t0,0
   4: 00037313      andi  t1,t1,0
   8: 0003f393      andi  t2,t2,0
  c: 000e7e13      andi  t3,t3,0
 10: 000efe93      andi  t4,t4,0
 14: 18a00293      li    t0,394
 18: 01300313      li    t1,19
 1c: 026283b3      mul   t2,t0,t1
 20: d003f053      fcvt.s.w  ft0,t2
 24: d20300d3      fcvt.d.w  ft1,t1
 28: 1a00f153      fdiv.d  ft2,ft1,ft0
 2c: 002071d3      fadd.s  ft3,ft0,ft2
 30: c001fed3      fcvt.w.s  t4,ft3
 34: 01d38e77      radd16  t3,t2,t4

```

Should you get the following output, you have successfully built your compiler!

Thanks,

L N Saaswath

ln.saaswath.eee19@iitbhu.ac.in