GCC support for Xtensa

- Git tree: https://github.com/jcmvbkbc/gcc-xtensa

Compiler build for full testing

gcc testsuite requires libc, so the easiest way is the following:

- build toolchain with libc, e.g. by the buildroot
- configure and build gcc with sysroot of that toolchain:

```
export TOOLCHAIN=/home/jcmvbkbc/tmp/br/build-toolchain
export TARGET=xtensa-buildroot-linux-uclibc  # use for linux
# export TARGET=xtensa-buildroot-uclinux-uclibc # use for uclinux
PATH=$TOOLCHAIN/host/usr/bin:$PATH ../gcc/configure \
    --prefix=`pwd`/root --target=$TARGET \
    --disable-libssp --disable-libisl --enable-languages=c,c++ \
    --enable-tls --enable-threads --enable-__cxa_atexit \
    --with-sysroot=$TOOLCHAIN/host/usr/$TARGET/sysroot
PATH=$TOOLCHAIN/host/usr/bin:$PATH make -j8 all
```

- run it on your board (board configuration is a file in a directory specified in the ~/.dejagnurc e.g. ~/boards):

```
DEJAGNU_TIMEOUT=1200 PATH=$TOOLCHAIN/host/usr/bin:$PATH \
RUNTESTFLAGS="CFLAGS_FOR_TARGET=--
sysroot=$TOOLCHAIN/host/usr/$TARGET/sysroot --target_board=qemu-xtensa"
\
make -k check
```

  Use check-gcc to only check the compiler. Use check-target to check target libs. Timeouts are weird. See gcc/testsuite/lib/timeout.exp for some timeouts, the rest are TBD.

- run it on linux-user QEMU:

```
PATH=$TOOLCHAIN/host/usr/bin:$PATH \
QEMU_LD_PREFIX=$ROOTFS \
RUNTESTFLAGS="CFLAGS_FOR_TARGET=--
sysroot=$TOOLCHAIN/host/usr/$TARGET/sysroot\ -mauto-litpools" \
make -k check
```

  File system in \$ROOTFS should have sysroot libraries installed, but nothing else. \$TOOLCHAIN/target usually works.

- to run uclinux tests use TARGET=xtensa-buildroot-uclinux-uclibc and provide environment variable FLTFLAGS='-s 0x01000000'. -Wl,-elf2flt='-s 0x01000000' passed in CFLAGS_FOR_TARGET breaks precompiled header tests.
- to run call0 tests provide environment variable QEMU_XTENSA_ABI_CALL0='' in qemu-4.2+

Compiler build for debug

```
../gcc/configure \
    --prefix=`pwd`/root --target=$TARGET \
    --disable-libssp --disable-libisl --enable-languages=c,c++ \
    --enable-debug --enable-valgrind-annotations --disable-docs \
    --enable-checking=all \
    CFLAGS='-O0 -g3' CXXFLAGS='-O0 -g3'
make -j8 all-gcc
make -j8 install-gcc
```

For –enable-valgrind-annotations to work valgrind must be installed.

To use ASAN add the following to configure:

```
../gcc/configure \
    ... \
    --with-stage1-libs="-lstdc++ -ldl" \
    CXXFLAGS=-fsanitize=address \
    LDFLAGS=-fsanitize=address \
    ...
```

and have

```
ASAN_OPTIONS='detect_leaks=0'
```

in the environment when building.

See also

- https://gcc.gnu.org/install/test.html
- https://gcc.gnu.org/wiki/DebuggingGCC
- https://gcc.gnu.org/wiki/HowToPrepareATestcase

From:
http://wiki.osll.ru/ - **Open Source & Linux Lab**

Permanent link:
http://wiki.osll.ru/doku.php/etc:users:jcmvbkbc:gcc-xtensa?rev=1679023344

Last update: **2023/03/17 06:22**