

binutils support for Xtensa

- git tree: <https://github.com/jcmvbkbc/binutils-gdb-xtensa>

FDPIC support

- [-] static linking
- [+] PDE
- [+] PIE
- [-] PLT and lazy binding
- [-] TLS

Manual toolchain building script

```
#!/bin/bash -ex

target=${TARGET:-xtensa-linux-uclibcfdpic}
build_base=`pwd`/build
src_base=$(dirname $(readlink -f "$0"))
binutils_src=$HOME/ws/tensilica/binutils-gdb/binutils-gdb
gcc_src=$HOME/ws/tensilica/gcc/gcc
linux_src="$src_base/linux"
uclibc_src="$src_base/uclibc-ng"
uclibc_config_src="$src_base/uclibc-ng-config"

prefix=`pwd`
sysroot="$prefix/$target/sysroot"
linux_headers="$sysroot/usr"

_FLAGS_FOR_HOST=${FLAGS_FOR_HOST:--0g -g}
_FLAGS_FOR_TARGET=${FLAGS_FOR_TARGET:--mauto-litpools -mfdpic -Oz -g}
CROSS_COMPILE=${CROSS_COMPILE:-$prefix/bin/$target-}
TARGET_CFLAGS="$_FLAGS_FOR_TARGET -D_LARGEFILE64_SOURCE -
D_FILE_OFFSET_BITS=64"

if [ "$1" = "-r" ]; then
    reconfigure=1
fi

mkdir -p .build

mkdir .build/binutils && (
    cd .build/binutils
    "$binutils_src/configure" --prefix="$prefix" \
        --target=$target \
        --disable-shared --disable-werror --disable-gdb --disable-
gdbstub \
        CFLAGS=$_FLAGS_FOR_HOST
```

```
    make -j8
    make -j8 install
)

mkdir .build/gcc-initial && (
    cd .build/gcc-initial
    "$gcc_src/configure" --prefix="$prefix" \
        --target=$target \
        --with-sysroot="$sysroot" \
        --enable-languages=c \
        --disable-shared \
        --enable-__cxa_atexit \
        --disable-tls --disable-threads \
        --without-headers --with-newlib \
    CFLAGS_FOR_TARGET="$_FLAGS_FOR_TARGET" \
    CXXFLAGS_FOR_TARGET="$_FLAGS_FOR_TARGET" \
    CFLAGS="$_FLAGS_FOR_HOST" \
    CXXFLAGS="$_FLAGS_FOR_HOST"

    make -j8 all-gcc
    make -j8 all-target-libgcc
    make -j8 install-gcc
    make -j8 install-target-libgcc
)

mkdir .build/linux && (
    cd .build/linux
    make -C "$linux_src" ARCH=xtensa \
        CROSS_COMPILE="$CROSS_COMPILE" O=`pwd` \
        defconfig
    make -C "$linux_src" ARCH=xtensa \
        CROSS_COMPILE="$CROSS_COMPILE" O=`pwd` \
        INSTALL_HDR_PATH="$linux_headers" \
        -j8 headers_install
)

mkdir .build/uclibc && (
    cd .build/uclibc
    cp "$uclibc_config_src/.config" .
    if [ -n "$reconfigure" ]; then
        make -C "$uclibc_src" ARCH=xtensa \
            CROSS_COMPILE="$CROSS_COMPILE" \
            O=`pwd` KERNEL_HEADERS="$linux_headers/include" \
            UCLIBC_EXTRA_CFLAGS="{TARGET_CFLAGS}" \
            menuconfig
        cp .config "$uclibc_config_src"
    fi

    make -C "$uclibc_src" ARCH=xtensa \
        CROSS_COMPILE="$CROSS_COMPILE" \
```

```
    O=`pwd` KERNEL_HEADERS="$linux_headers/include" \  
    UCLIBC_EXTRA_CFLAGS="${TARGET_CFLAGS}" \  
    -j8 "$@"  
make -C "$uclibc_src" ARCH=xtensa \  
    CROSS_COMPILE="$CROSS_COMPILE" \  
    O=`pwd` KERNEL_HEADERS="$linux_headers/include" \  
    UCLIBC_EXTRA_CFLAGS="${TARGET_CFLAGS}" \  
    DESTDIR="$sysroot" \  
    install  
)  
  
mkdir .build/gcc-final && (  
    cd .build/gcc-final  
    "$gcc_src/configure" --prefix="$prefix" \  
        --target=$target \  
        --with-sysroot="$sysroot" \  
        --enable-languages=c,c++ \  
        --disable-shared \  
        --enable-__cxa_atexit \  
        --disable-tls --disable-threads \  
        --with-uclibc \  
    CFLAGS_FOR_TARGET="_FLAGS_FOR_TARGET" \  
    CXXFLAGS_FOR_TARGET="_FLAGS_FOR_TARGET" \  
    CFLAGS="_FLAGS_FOR_HOST" \  
    CXXFLAGS="_FLAGS_FOR_HOST"  
  
    make -j8 all  
    make -j8 install  
)
```

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

Permanent link:
<http://wiki.osll.ru/doku.php/etc:users:jcmvbkbc:binutils-xtensa?rev=1694215289>

Last update: **2023/09/09 02:21**

