

1. create makefile-external file, include the following header

```
1.  # machine based sysroot directory
2.  MACHINE ?= granite2v8
3.  # MACHINE ?= granite2
4.
5.  SYSROOT ?= /home/walterzh/work/2016-05-LSP/64-bit/tmp/sysroots
6.  # Don't think about using a 32bit linux to build on any more.
7.  ARM_ROOT ?= $(SYSROOT)/x86_64-linux/usr/bin
8.
9.  ifeq ($(MACHINE), granite2v8)
10.  CCPREFIX ?= aarch64-poky-linux/aarch64-poky-linux-
11.  LIBPATH := $(SYSROOT)/$(MACHINE)/usr/lib64
12.  else
13.  CCPREFIX ?= arm-poky-linux-gnueabi/arm-poky-linux-gnueabi-
14.  LIBPATH := $(SYSROOT)/$(MACHINE)/usr/lib
15.  endif
16.  # if your building 32bit code for the R4 on a 64bit then:
17.  # CCPREFIX := arm-pokymllib32r4-linux-gnueabi/arm-pokymllib32r4-linux-gnueabi
18.
19.  CROSS_COMPILE ?= $(ARM_ROOT)/$(CCPREFIX)
20.
21.  AS      := $(CROSS_COMPILE)gcc --sysroot=$(SYSROOT)/$(MACHINE)
22.  LD      := $(CROSS_COMPILE)ld --sysroot=$(SYSROOT)/$(MACHINE)
23.  CC      := $(CROSS_COMPILE)gcc --sysroot=$(SYSROOT)/$(MACHINE)
24.  AS      := $(CROSS_COMPILE)gcc --sysroot=$(SYSROOT)/$(MACHINE)
25.  LD      := $(CROSS_COMPILE)ld --sysroot=$(SYSROOT)/$(MACHINE)
```

1. 根据不同project添加不同要build的source

For example

- jbig-codec-app-test project

```
1.  SOURCES = jbig_test.c
2.
3.  CFLAGS += -Wall -g -O0 -DASIC_granite2 -I.
4.
5.  OBJECTS=$(SOURCES:.c=.o)
6.
7.  EXE=jbigtest
8.  all : $(EXE)
9.
10. jbigtest : $(OBJECTS)
11.     $(AR) -r libjbigtest.a $(OBJECTS)
12.     $(CC) $(CFLAGS) $(LDFLAGS) $(OBJECTS) -lpthread -lrt -lm -ldmaalloc -ljbig -o $@
```

- jbig-codec-app project

```

1. SOURCES = jbigUtils.c mrvl_jbig.c mrvl_jbig_config.c
2.
3. CFLAGS += -fPIC -g -O0 -Wall -DASIC_granite2 -DHAVE_UIO_LIB -I. "-I=/usr/include/dmaalloc-app"
4.
5. OBJECTS=$(SOURCES:.c=.o)
6.
7. all: $(OBJECTS)
8.     @echo $(OBJECTS)
9.     @echo $(SOURCES)
10.    $(AR) -r libjbig.a $(OBJECTS)
11.    $(CC) $(CFLAGS) $(LDFLAGS) $(OBJECTS) -lpthread -lrt -shared -Wl,-soname,libjbig.so.1 -o libjbig.so.1.0

```

- dmaalloc-app project

```

1. SOURCES=list.c mlist.c key_table.c
2. SOURCES+=BigBuffer.c dma_buffer.c dma_alloc_app.c uio_lib.c
3. SOURCES+=memAPI.c mlimiter.c mlimiter_config.c
4. SOURCES+=logger.c
5. SOURCES+=debug_hexdump.c map_mem.c
6. SOURCES+=posix_ostools.c
7.
8. all: test_dmaalloc_app.c
9.    $(CC) -g test_dmaalloc_app.c $(SOURCES) -I../ -L$(LIBPATH) -lpthread -lrt -o test_dmaalloc_app
10.
11. solink: test_dmaalloc_app.c
12.    $(CC) $(CFLAGS) $(LDFLAGS) $(SOURCES) -lpthread -lrt -shared -Wl,-soname,libdmaalloc.so.1 -o libdmaalloc.so.1.0
13.    $(CC) -g test_dmaalloc_app.c -L$(LIBPATH) -lpthread -lrt -ldmaalloc -o test_dmaalloc_app

```