

Implementation

I. Steps for running an application on qemu through Zephyr OS

1) Install required packages on ubuntu to run zephyr

```
$ sudo apt-get install git make gcc g++ ncurses-dev gperf ccache\doxygen dfu-util device -tree-compiler python3-ply python3-pip
```

2) Download Zephyr kernel and Zephyr SDK from <http://www.zephyrproject.org/>

3) Run the installation binary

```
$ chmod +x zephyr-sdk-<version>-setup.run  
$ ./zephyr-sdk-<version>-setup.run
```

4) Set the environment variables (required for every new terminal window)

```
$ export ZEPHYR_GCC_VARIANT=zephyr  
$ export ZEPHYR_SDK_INSTALL_DIR=<sdk intallation directory> $  
export ZEPHYR_BASE=<zephyr kernel location>
```

5) Enter the /samples/helloworld directory in zephyr kernel

6) Set the board as qemu_x86 and execute make command \$

```
make BOARD=qemu_x86 qemu
```

The screenshot shows a terminal window on a Linux desktop. The terminal output is as follows:

```
:[sudo] password for tejasree:  
[sudo] password for tejasree:  
[~]$ chmod +x zephyr-sdk-0.9.1-setup.run  
[~]$ sudo ./zephyr-sdk-0.9.1-setup.run  
[sudo] password for tejasree:  
Verifying archive integrity... All good.  
Uncompressing SDK for Zephyr 100%  
Enter target directory for SDK (default: /opt/zephyr-sdk): /opt/zephyr-sdk/  
Installing SDK to /opt/zephyr-sdk  
The directory /opt/zephyr-sdk/sysroots will be removed!  
Do you want to continue (y/n)? y  
[*] Installing x86 tools...  
[*] Installing arm tools...  
[*] Installing arc tools...  
[*] Installing iamcu tools...  
[*] Installing ntos2 tools...  
[*] Installing xtensa tools...  
[*] Installing riscv32 tools...  
[*] Installing additional host tools...  
Success installing SDK. SDK is ready to be used.  
:[~]$ export ZEPHYR_GCC_VARIANT=zephyr  
:[~]$ export ZEPHYR_SDK_INSTALL_DIR=/opt/zephyr-sdk/  
:[~]$ export ZEPHYR_BASE=/home/tejasree/zephyr-zephyr-v1.8.0/  
:[~]$ cd /home/tejasree/zephyr-zephyr-v1.8.0/samples/hello_world/  
:[~]$ su  
root@tejasree-zephyr-zephyr-v1.8.0:/home/tejasree/zephyr-zephyr-v1.8.0/samples/hello_world# make BOARD=qemu_x86 qemu  
This target is deprecated, use make run instead  
make[1]: Entering directory '/home/tejasree/zephyr-zephyr-v1.8.0'  
make[2]: Entering directory '/home/tejasree/zephyr-zephyr-v1.8.0/samples/hello_world/outdir/qemu_x86'  
CHK include/generated/generated_dts_board.conf  
Using /home/tejasree/zephyr-zephyr-v1.8.0 as source for kernel  
GEN ./Makefile  
CHK include/generated/version.h  
CHK include/generated/generated_dts_board.h  
CHK misc/generated/configs.c  
CHK include/generated/offsets.h  
CC lib/libc/minimal/source/stdlib/strtol.o  
CC lib/libc/minimal/source/stdlib/strtoul.o  
CC lib/libc/minimal/source/stdlib/atol.o  
LD lib/libc/minimal/source/stdlib/built-in.o  
CC lib/libc/minimal/source/stdout/fprintf.o  
CC lib/libc/minimal/source/stdout/prf.o  
CC lib/libc/minimal/source/stdout/sprintf.o
```

```

/zephyr-zephyr-v1.8.0/samples/hello_world
CC  drivers/interrupt_controller/ioapic_intr.o
LD  drivers/interrupt_controller/built-in.o
CC  drivers/serial/uart_ns16550.o
LD  drivers/serial/built-in.o
CC  drivers/timer/hpet.o
CC  drivers/timer/sys_clock_init.o
LD  drivers/timer/built-in.o
LD  drivers/built-in.o
CC  kernel/alert.o
CC  kernel/device.o
CC  kernel/errno.o
CC  kernel/idle.o
CC  kernel/init.o
CC  kernel/mailbox.o
CC  kernel/mem_slab.o
CC  kernel/mempool.o
CC  kernel/msg_q.o
CC  kernel/mutex.o
CC  kernel/pipes.o
CC  kernel/queue.o
CC  kernel/sched.o
CC  kernel/sem.o
CC  kernel/stack.o
CC  kernel/sys_clock.o
CC  kernel/system_work_q.o
CC  kernel/thread.o
CC  kernel/thread_abort.o
CC  kernel/timer.o
CC  kernel/work_q.o
AR  kernel/lib.a
CC  src/main.o
LD  src/built-in.o
AR  libzephyr.a
LINK zephyr.lnk
SDT  staticidt.o
BIN  zephyr.bin
To exit from QEMU enter: 'CTRL+a, x'
[QEMU] CPU: qemu32
qemu-system-i386: warning: Unknown firmware file in legacy mode: genroms/multiboot.bin
***** BOOTING ZEPHYR OS v1.8.0 - BUILD: Oct  5 2017 07:09:35 *****
Hello World! x86

```

II. MQTT Server-Client connection using mosquitto on linux machine

1) Steps to install mosquitto (on linux machine)

- Add the mosquitto repository by given below commands

```
$ sudo apt-add-repository ppa:mosquitto-dev/mosquitto-ppa
$ sudo apt-get update
```

- Execute the given below command to install the Mosquitto broker package

```
$ sudo apt-get install mosquitto
```

- Install Mosquitto developer libraries to develop MQTT clients

```
$ sudo apt-get install libmosquitto-devE
```

- Execute the given below command to install Mosquitto client packages

```
$ sudo apt-get install mosquitto-clients
```

- Ensure that Mosquitto broker is running

```
$sudo service mosquitto status
```

- Testing

From terminal 1 -

```
$mosquitto_sub -h localhost -t "mqtt"
```

terminal 2 -

```
$mosquitto_pub -h localhost -t "mqtt" -m "Hello MQTT"
```

The screenshot shows a dual-terminal window on a Linux desktop environment. The left terminal window contains the command:

```
mosquitto_pub -h localhost -t "mqtt" -m "Hello MQTT"
```

The right terminal window contains the command:

```
mosquitto_sub -h localhost -t "mqtt" -v
```

Both terminals show the output of the command being run.

2) Basic connection of client and server through mosquitto

The screenshot shows a dual-terminal window on a Linux desktop environment. The left terminal window contains the command:

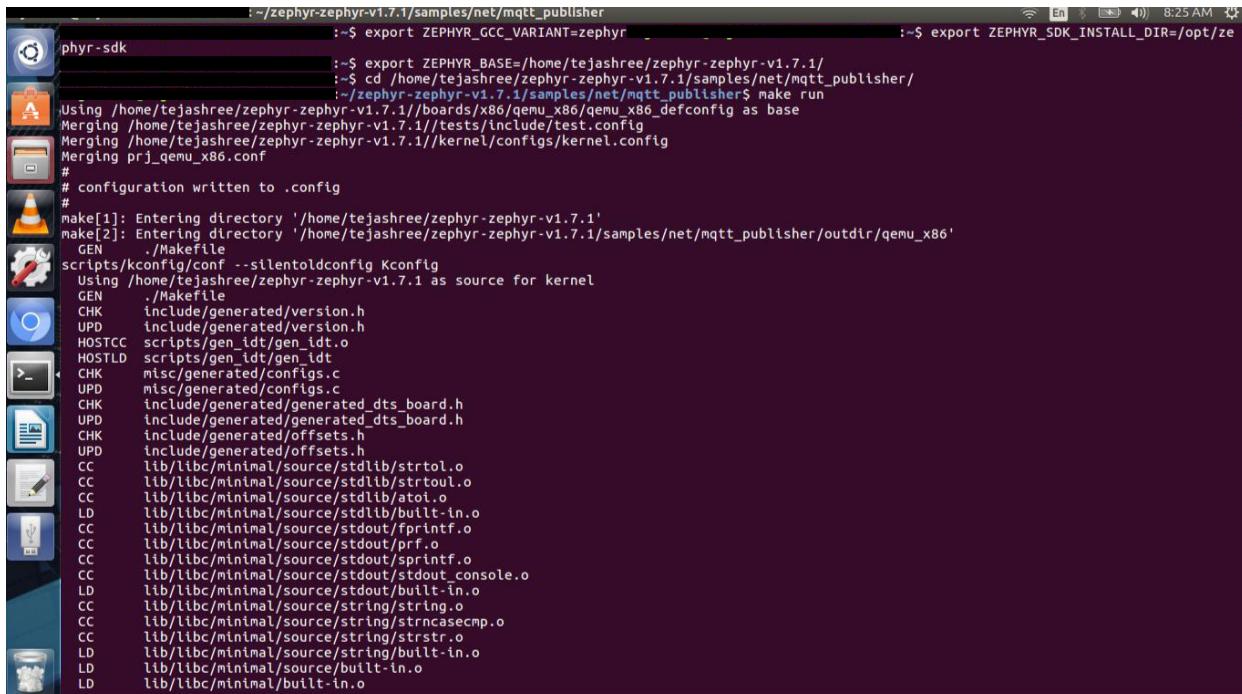
```
mosquitto_sub -h 192.168.0.109 -p 1883 -t "mqtt" -d -v
```

The right terminal window contains the command:

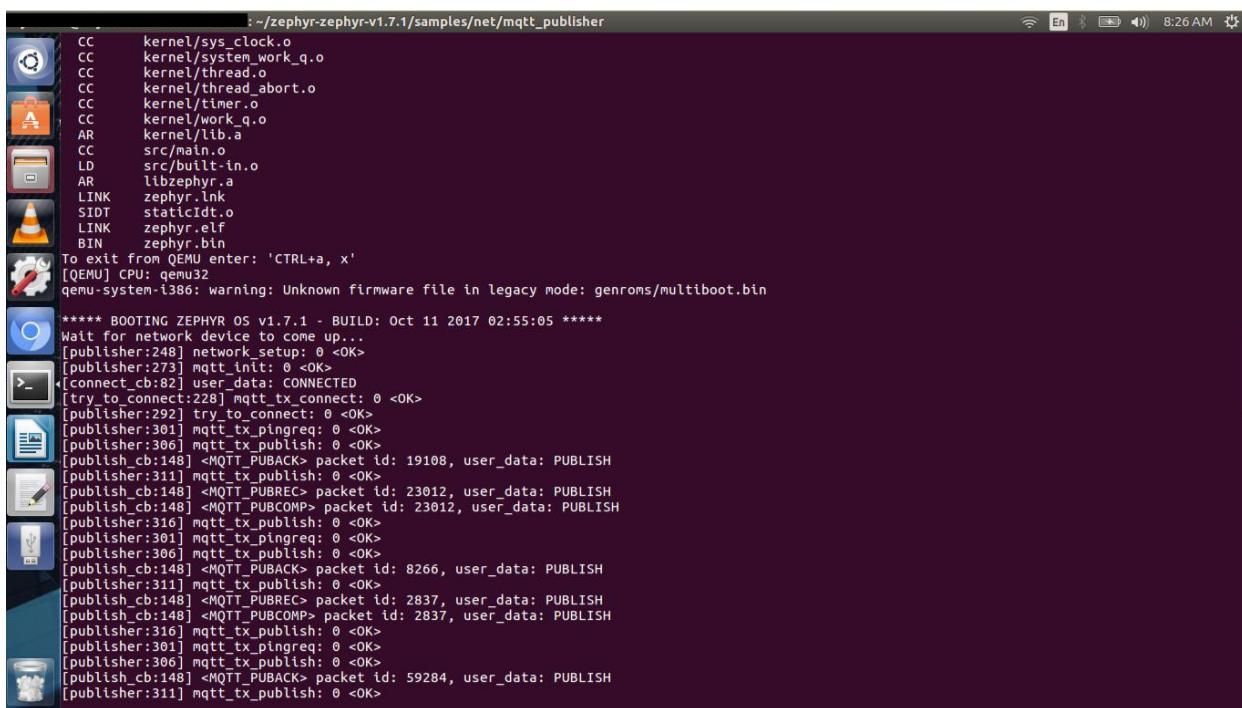
```
mosquitto_pub -h 192.168.0.109 -p 1883 -t "mqtt" -m "MESSAGE FOR MQTT1" -d
```

Both terminals show the detailed log output of the MQTT protocol exchange between the client and the server.

3) Publish subscribe through qemu



```
:=/zephyr-zephyr-v1.7.1/samples/net/mqtt_publisher
:~$ export ZEPHYR_GCC_VARIANT=zephyr
:~$ cd /home/tejashree/zephyr-zephyr-v1.7.1/samples/net/mqtt_publisher/
:~$ ./zephyr-zephyr-v1.7.1/samples/net/mqtt_publisher$ make run
Using /home/tejashree/zephyr-zephyr-v1.7.1//boards/x86/qemu_x86/qemu_x86_defconfig as base
Merging /home/tejashree/zephyr-zephyr-v1.7.1//tests/include/test.config
Merging /home/tejashree/zephyr-zephyr-v1.7.1//kernel/configs/kernel.config
Merging prj_qemu_x86.conf
#
# configuration written to .config
#
make[1]: Entering directory '/home/tejashree/zephyr-zephyr-v1.7.1'
make[2]: Entering directory '/home/tejashree/zephyr-zephyr-v1.7.1/samples/net/mqtt_publisher/outdir/qemu_x86'
GEN      ./Makefile
scripts/kconfig/conf --silentoldconfig Kconfig
Using /home/tejashree/zephyr-zephyr-v1.7.1 as source for kernel
GEN      ./Makefile
CHK      include/generated/version.h
UPD      include/generated/version.h
HOSTCC  scripts/gen_ldt/gen_ldt.o
HOSTLD  scripts/gen_ldt/gen_ldt
CHK      misc/generated/configs.c
UPD      misc/generated/configs.c
CHK      include/generated/generated_dts_board.h
UPD      include/generated/generated_dts_board.h
CHK      include/generated/offsets.h
UPD      include/generated/offsets.h
CC       lib/libc/minimal/source/stdlib strtol.o
CC       lib/libc/minimal/source/stdlib strtol.o
CC       lib/libc/minimal/source/stdlib atol.o
LD       lib/libc/minimal/source/stdlib/built-in.o
CC       lib/libc/minimal/source/stdout/fprintf.o
CC       lib/libc/minimal/source/stdout/prf.o
CC       lib/libc/minimal/source/stdout sprintf.o
CC       lib/libc/minimal/source/stdout/stdout_console.o
LD       lib/libc/minimal/source/stdout/built-in.o
CC       lib/libc/minimal/source/string/string.o
CC       lib/libc/minimal/source/string/strncasecmp.o
CC       lib/libc/minimal/source/string/strstr.o
LD       lib/libc/minimal/source/string/built-in.o
LD       lib/libc/minimal/source/built-in.o
LD       lib/libc/minimal/source/built-in.o
```

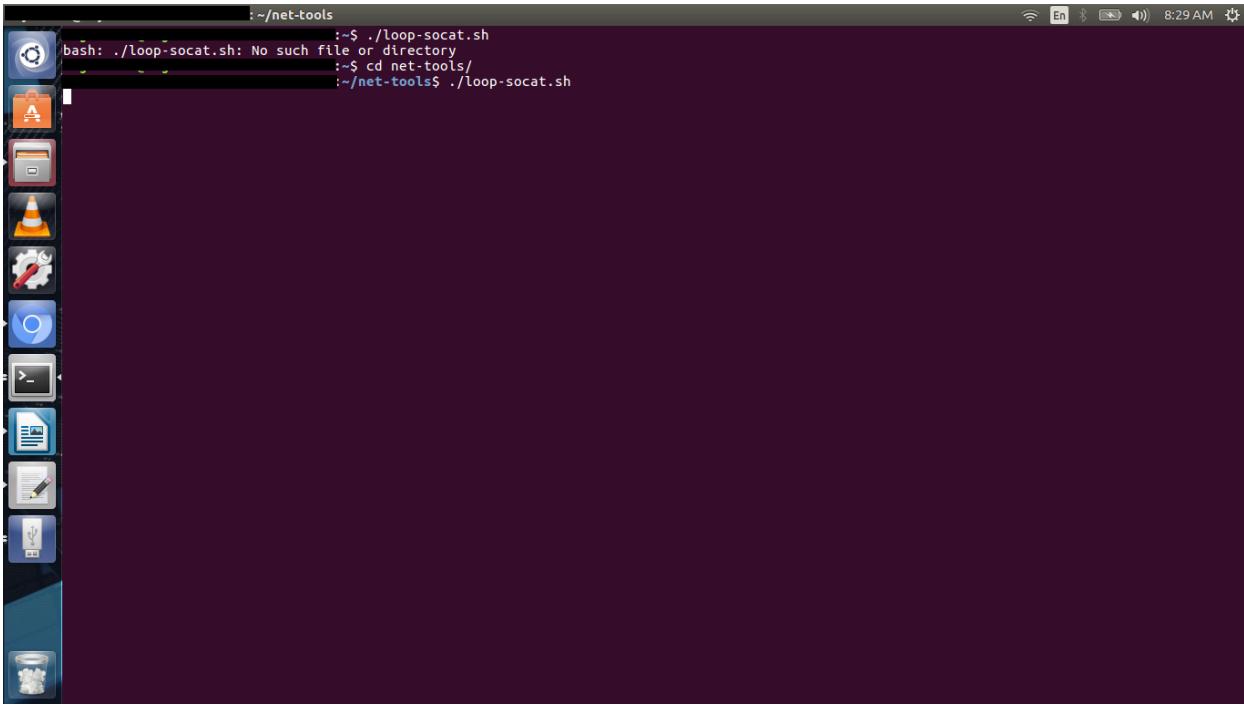


```
:=/zephyr-zephyr-v1.7.1/samples/net/mqtt_publisher
CC  kernel/sys_clock.o
CC  kernel/system_work_q.o
CC  kernel/thread.o
CC  kernel/thread_abort.o
CC  kernel/timer.o
CC  kernel/work_q.o
AR  kernel/lib.a
CC  src/main.o
LD  src/built-in.o
AR  libzephyr.a
LINK zephyr.ink
SIDT staticIdt.o
LINK zephyr.elf
BIN  zephyr.bin
To exit from QEMU enter: 'CTRL+a, x'
[QEMU] CPU: qemu32
qemu-system-i386: warning: Unknown firmware file in legacy mode: genroms/multiboot.bin
***** BOOTING ZEPHYR OS v1.7.1 - BUILD: Oct 11 2017 02:55:05 *****
Wait for network device to come up...
[publisher:248] network_setup: 0 <OK>
[publisher:273] mqtt_inIt: 0 <OK>
[connect_cb:82] user_data: CONNECTED
[try_to_connect:228] mqtt_tx_connect: 0 <OK>
[publisher:292] try_to_connect: 0 <OK>
[publisher:301] mqtt_tx_pingreq: 0 <OK>
[publisher:306] mqtt_tx_publish: 0 <OK>
[publisher_cb:148] <MQTT_PUBLISH> packet id: 19108, user_data: PUBLISH
[publisher:311] mqtt_tx_publish: 0 <OK>
[publisher_cb:148] <MQTT_PUBLISH> packet id: 23012, user_data: PUBLISH
[publisher_cb:148] <MQTT_PUBLISH> packet id: 23012, user_data: PUBLISH
[publisher:316] mqtt_tx_publish: 0 <OK>
[publisher:301] mqtt_tx_pingreq: 0 <OK>
[publisher:306] mqtt_tx_publish: 0 <OK>
[publisher_cb:148] <MQTT_PUBLISH> packet id: 8266, user_data: PUBLISH
[publisher:311] mqtt_tx_publish: 0 <OK>
[publisher_cb:148] <MQTT_PUBLISH> packet id: 2837, user_data: PUBLISH
[publisher_cb:148] <MQTT_PUBLISH> packet id: 2837, user_data: PUBLISH
[publisher:316] mqtt_tx_publish: 0 <OK>
[publisher:301] mqtt_tx_pingreq: 0 <OK>
[publisher:306] mqtt_tx_publish: 0 <OK>
[publisher_cb:148] <MQTT_PUBLISH> packet id: 59284, user_data: PUBLISH
[publisher:311] mqtt_tx_publish: 0 <OK>
```

```
CC  kernel/sys_clock.o
CC  kernel/system_work_q.o
CC  kernel/thread.o
CC  kernel/thread_abort.o
CC  kernel/timer.o
CC  kernel/work_q.o
AR  kernel/lib.a
CC  src/main.o
LD  src/built-in.o
AR  libzephyr.a
LINK zephyr.lnk
SDT  staticldt.o
LINK zephyr.elf
BIN  zephyr.bin
To exit from QEMU enter: 'CTRL+a, x'
[QEMU] CPU: qemu32
qemu-system-i386: warning: Unknown firmware file in legacy mode: genroms/multiboot.bin
***** BOOTING ZEPHYR OS v1.7.1 - BUILD: Oct 11 2017 02:55:05 *****
Wait for network device to come up...
[publisher:248] network_setup: 0 <OK>
[publisher:273] mqtt_init: 0 <OK>
[connect_cb:82] user_data: CONNECTED
[try_to_connect:228] mqtt_tx_connect: 0 <OK>
[publisher:292] try_to_connect: 0 <OK>
[publisher:301] mqtt_tx_pingreq: 0 <OK>
[publisher:306] mqtt_tx_publish: 0 <OK>
[publisher:311] mqtt_tx_publish: 0 <OK>
[publisher:316] <MQTT_PUBLISH> packet id: 19108, user_data: PUBLISH
[publisher:311] mqtt_tx_publish: 0 <OK>
[publisher:316] <MQTT_PUBLISH> packet id: 23012, user_data: PUBLISH
[publisher:316] <MQTT_PUBLISH> packet id: 23012, user_data: PUBLISH
[publisher:316] mqtt_tx_publish: 0 <OK>
[publisher:301] mqtt_tx_pingreq: 0 <OK>
[publisher:306] mqtt_tx_publish: 0 <OK>
[publisher:311] <MQTT_PUBLISH> packet id: 8266, user_data: PUBLISH
[publisher:311] mqtt_tx_publish: 0 <OK>
[publisher:316] <MQTT_PUBLISH> packet id: 2837, user_data: PUBLISH
[publisher:316] <MQTT_PUBLISH> packet id: 2837, user_data: PUBLISH
[publisher:316] mqtt_tx_publish: 0 <OK>
[publisher:301] mqtt_tx_pingreq: 0 <OK>
[publisher:306] mqtt_tx_publish: 0 <OK>
[publisher:311] <MQTT_PUBLISH> packet id: 59284, user_data: PUBLISH
[publisher:311] mqtt_tx_publish: 0 <OK>
```

```
:~/net-tools
:~/net-tools$ cd net-tools/
:~/net-tools$ sudo ./loop-slip-tap.sh[sudo] password for tejasree:
*****SLIP started on '/tmp/slip.dev'
slipfd and inslips reopened
ip neigh flush dev tap0
Cannot find device "tap0"
opened tap device '/dev/tap0'
ifconfig tap0 up
ip -6 route add 2001:db8::/64 dev tap0
ip -6 addr add 2001:db8::2/64 dev tap0
ip route add 192.0.2.0/24 dev tap0
ip addr add 192.0.2.2/24 dev tap0
ifconfig tap0
tap0      Link encap:Ethernet HWaddr 52:81:2f:aa:f5:9a
          inet addr:192.0.2.2 Bcast:0.0.0.0 Mask:255.255.255.0
          inet6 addr: 2001:db8::2/64 Scope:Global
            inet6 addr: fe80::5081:ffff:fea:f59a/64 Scope:Link
              UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
              RX packets:0 errors:0 dropped:0 overruns:0 frame:0
              TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
              collisions:0 txqueuelen:1000
              RX bytes:0 (0.0 B)   TX bytes:0 (0.0 B)

can't open siodev '/tmp/slip.dev'
*****SLIP started on '/tmp/slip.dev'
slipfd and inslips reopened
ip neigh flush dev tap0
can't open siodev '/tmp/slip.dev'
*****SLIP started on '/tmp/slip.dev'
slipfd and inslips reopened
ip neigh flush dev tap0
```



A screenshot of an Ubuntu desktop environment. On the left, there is a vertical dock containing various application icons, including a terminal, file manager, browser, and system settings. The main window is a terminal window titled 'net-tools' with the command line '/net-tools'. The terminal output shows:

```
:~$ ./loop-socat.sh
bash: ./loop-socat.sh: No such file or directory
:~$ cd net-tools/
:~$ ./loop-socat.sh
```