# Enhancement to RNG Device: Supporting Multiple RNG Modes

Moad ELMARDI

January 24, 2025

## 1 Introduction

This enhancement modifies the Random Number Generator (RNG) device to support multiple modes of random number generation. The key goal was to allow users to choose between two random number generation methods: a default ‘rand()‘ function and a time-based random number generator derived from system time.

## 2 Enhancement Description

### 2.1 Multiple RNG Modes

The original RNG device used only the C standard library’s ‘rand()‘ function to generate random numbers. With this enhancement, two modes are now available:

- **Mode 0 (Default)**: Uses the ‘rand()‘ function for random numbers.
- **Mode 1 (Time-Based)**: Uses the current system time (‘time(NULL)‘) as an entropy source.

The mode can be selected by writing to a specific memory-mapped I/O (MMIO) register.

### 2.2 Implementation Details

To support multiple modes, we added a **mode** field to the **my_rng** structure. The ‘mmio_read‘ function was updated to check the mode and generate random numbers accordingly:

```
switch (dev->mode) {
    case RNG_MODE_TIME_BASED:
            return time(NULL);
            break;
    case RNG_MODE_DEFAULT:
            return rand();
            break;
```

Additionally, we updated the ‘mmio_write‘ function to set the mode by writing to a specific register. An ‘ioctl‘ interface was implemented in the driver to allow user-space applications to change modes.

## 3 Testing the Enhancement

### 3.1 User-Space Application

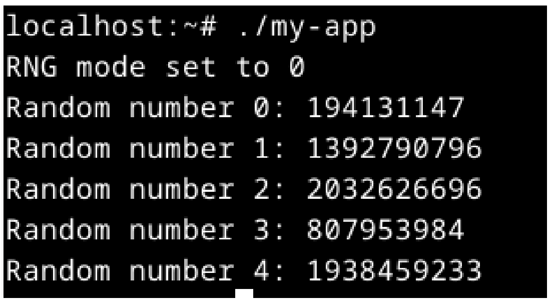
A simple application **test-app** was developed to allow the user to set the mode and retrieve random numbers:

```
ioctl(fd, MY_RNG_IOCTL_SET_MODE, &mode);
ioctl(fd, MY_RNG_IOCTL_RAND, &random_number);
```

The user can test both modes by switching between them and observing the random numbers generated.

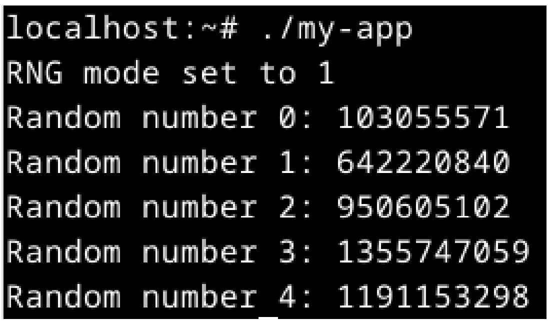
### 3.2 Outcome

Mode 0 generates random numbers based on ‘rand()‘, while Mode 1 generates random numbers based on system time:



```
localhost:~# ./my-app
RNG mode set to 0
Random number 0: 194131147
Random number 1: 1392790796
Random number 2: 2032626696
Random number 3: 807953984
Random number 4: 1938459233
```

Figure 1: Mode 0



```
localhost:~# ./my-app
RNG mode set to 1
Random number 0: 103055571
Random number 1: 642220840
Random number 2: 950605102
Random number 3: 1355747059
Random number 4: 1191153298
```

Figure 2: Mode 1

## 4 Conclusion

This enhancement improves the flexibility of the RNG device by enabling two modes of random number generation. The modification was successfully implemented, and testing confirmed the ability to switch modes and retrieve random numbers as expected.