This is a very simple yet flexible PRNG. It can be configured for 2..16 bits of randomness. Thanks to Zed Yago for providing the proper tap sequences. They are taken from “Graphics Gems”.

Tap Sequences

```
LFSR Term -> Period
-------------------
$03 -> 1..$03
$06 -> 1..$07
$0c -> 1..$0f
$14 -> 1..$1f
$30 -> 1..$3f
$60 -> 1..$7f
$b8 -> 1..$ff
$0110 -> 1..$01ff
$0240 -> 1..$03ff
$0500 -> 1..$07ff
$0ca0 -> 1..$0fff
$1b00 -> 1..$1fff
$3500 -> 1..$3fff
$6000 -> 1..$7fff
$b400 -> 1..$ffff
```

16-bit version - you have to read the random number from rand.

```
prng16:
    lsr rand+1
    ror rand
    bcc skip
    lda rand+1
    ; $60 = 1..32767, see table above
    eor #$60
    sta rand+1
skip:
    rts

rand:
    ; seed mustn't be zero!
    .byte $01,$00
```

8-bit version - returns the random number in A.

```
prng8:
    lsr rand
    lda rand
    bcc skip
    ; $30 = 1..63, see table above
    eor #$30
    sta rand
skip:
    rts
```

```
rand:
    ; seed mustn't be zero!
    .byte $01
```