Non-Deterministic Timers for Hardware Trojan Activation (Or How a Little Randomness Can Go the *Wrong* Way)

Frank Imeson

Electrical and Computer Engineering

UNIVERSITY OF WATERLOO



August 8^{th}, 2016

Collaborators: Saeed Nejati (Waterloo), Siddharth Garg (NYU) and Mahesh V. Tripunitara (Waterloo)

Intro

1. Intro
   - Attack Model
   - Background
2. Deterministic Timers
   - Hardware
   - Single Stage D-Timer
   - Dual Stage D-Timer
3. ND-Timers
   - Single Shot Timer
   - ND-Timer
4. TRNG
   - Clock Jitter
   - Ring Oscillators
   - Sampling
5. Results
   - Hardware
   - 24 Hour Experiment
6. Detection
   - Hardware
   - Simulation
   - Analysis
7. Summary

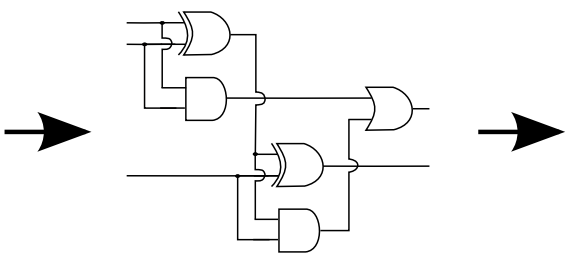
Attack Model

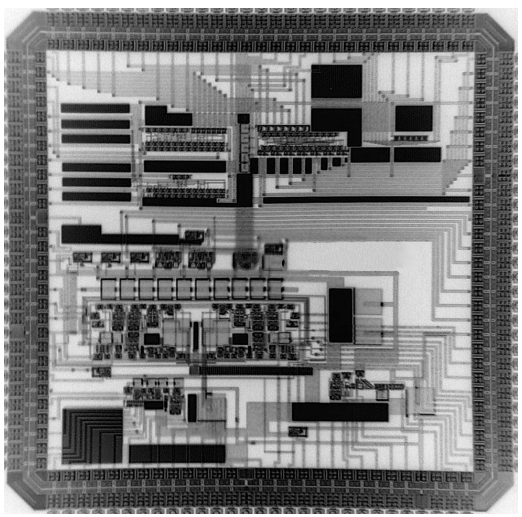
HDL **Netlist** **IC**

```
case(display_state)
 UPDATE : begin
  seg00_reg <= seg00;
  seg01_reg <= seg01;

  // update leds
  if (count00[0]) begin
    state <= UPDATE;
  end

 default : begin
  ons00 <= 0;
  count00 <= 0;
  display_state <= UPDATE;
 end
endcase
```

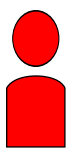




Credit: www.theverge.com/2011/11/16/2565638/mit-neural-connectivity-silicon-synapse

Attack Model

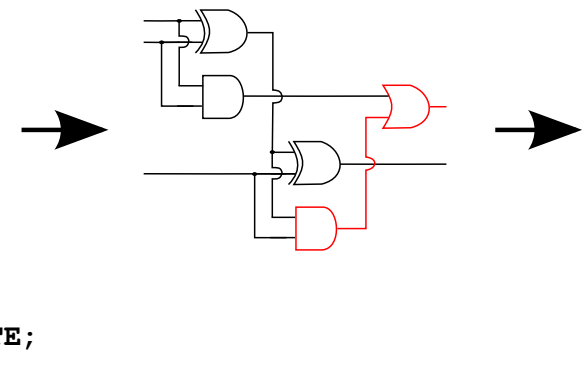
HDL **Netlist** **IC**



```
case(display_state)
 UPDATE : begin
  seg00_reg <= seg00;
  seg01_reg <= seg01;

  // update leds
  if (count00[0]) begin
    state <= UPDATE;
  end

 default : begin
  ons00 <= 1;
  count00 <= 1;
  display_state <= UPDATE;
 end
endcase
```



Credit: www.theverge.com/2011/11/16/2565638/mit-neural-connectivity-silicon-synapse

Background

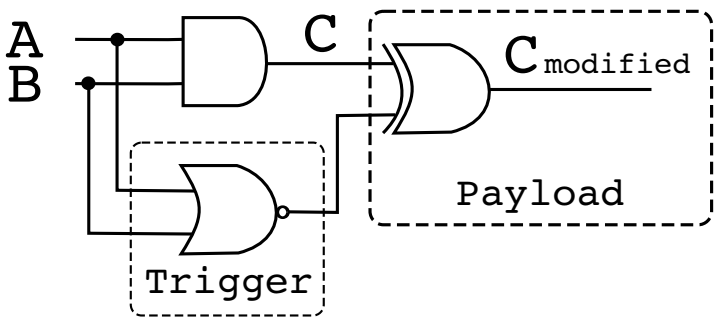
Hardware modifications are permanent!

Trojans:

- Privilege escalation [5]
- Leaking private information [3]
- Sabotage [4]

Activation Types:

- Time Trigger [2]
- Data Trigger [5]



Deterministic Timers

1 Intro
- Attack Model
- Background

2 Deterministic Timers
- Hardware
- Single Stage D-Timer
- Dual Stage D-Timer

3 ND-Timers
- Single Shot Timer
- ND-Timer

4 TRNG
- Clock Jitter
- Ring Oscillators
- Sampling

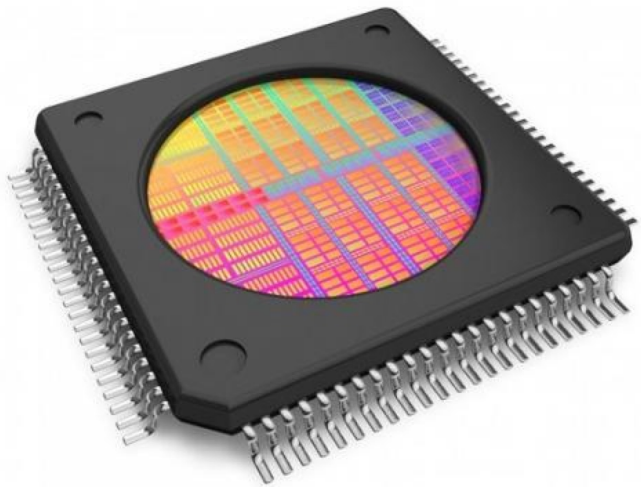
5 Results
- Hardware
- 24 Hour Experiment

6 Detection
- Hardware
- Simulation
- Analysis

7 Summary

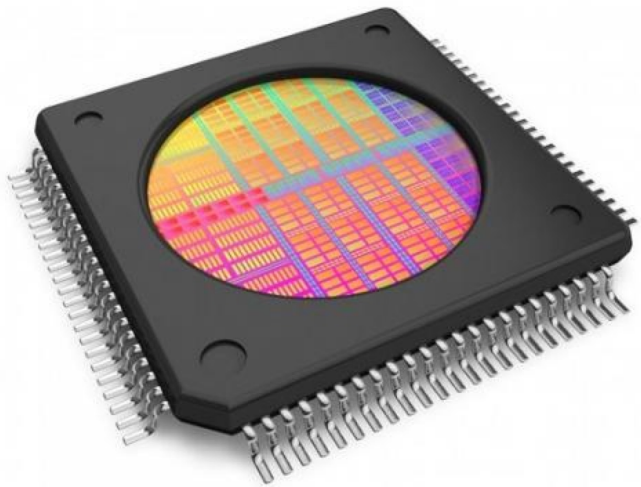
Hardware Constraints

- No access to Wall time (guarded)
- Power on time
- Power is not persistent
- Cannot save state off chip (guarded)
- NV-memory can be used on chip

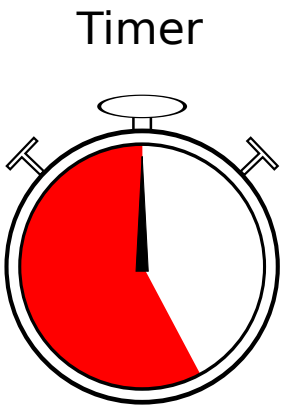


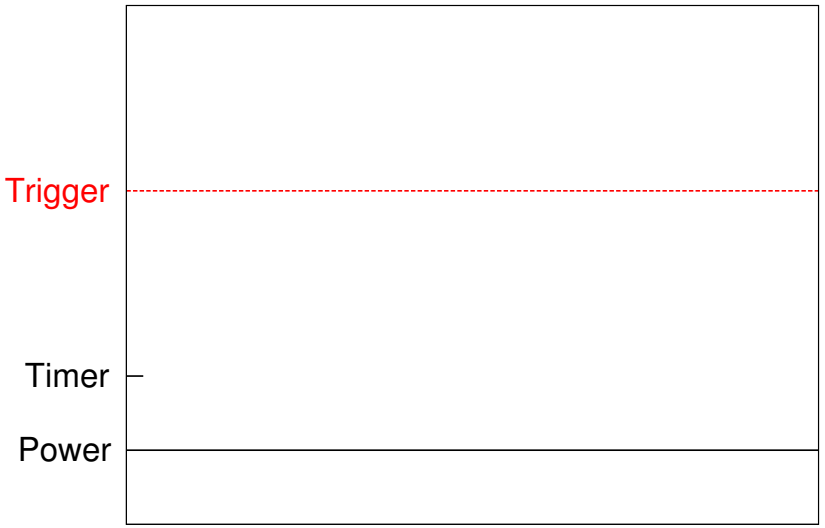
Hardware Constraints

- No access to Wall time (guarded)
- Power on time
- Power is not persistent
- Cannot save state off chip (guarded)
- NV-memory can be used on chip
- NV-memory technology, e.g., Flash, has limited write durability



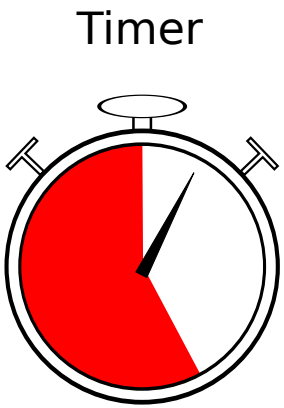
Single Stage D-Timer

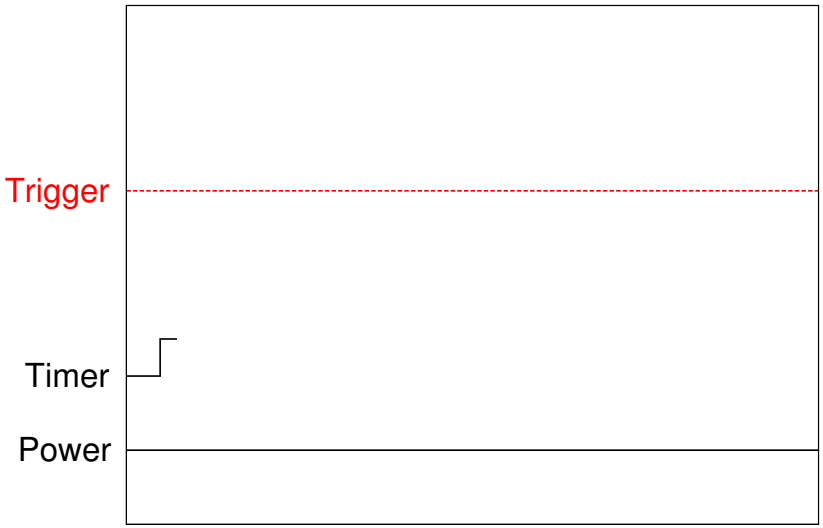




- Deterministic Timer (D-Timer) will trigger at a specified number of clock cycles with probability 1

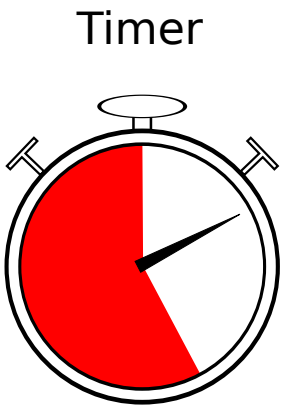
Single Stage D-Timer

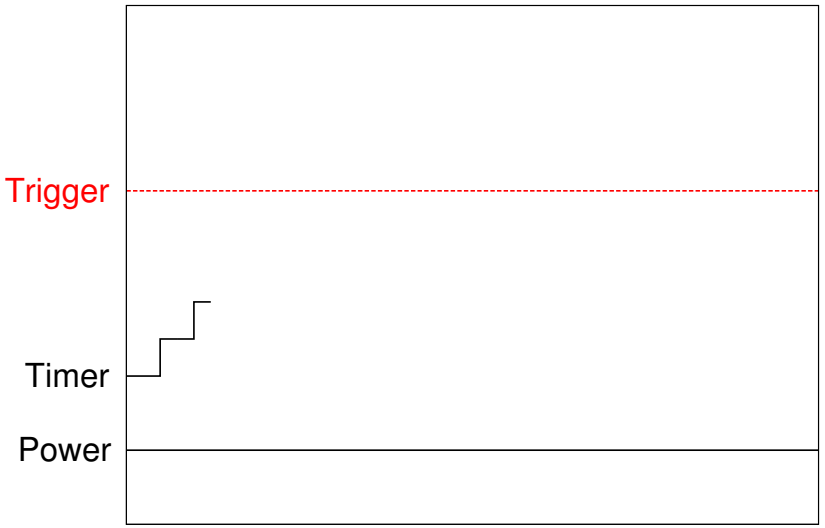




- Deterministic Timer (D-Timer) will trigger at a specified number of clock cycles with probability 1

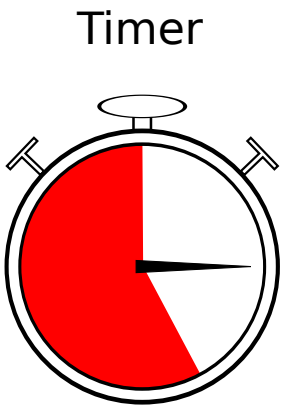
Single Stage D-Timer

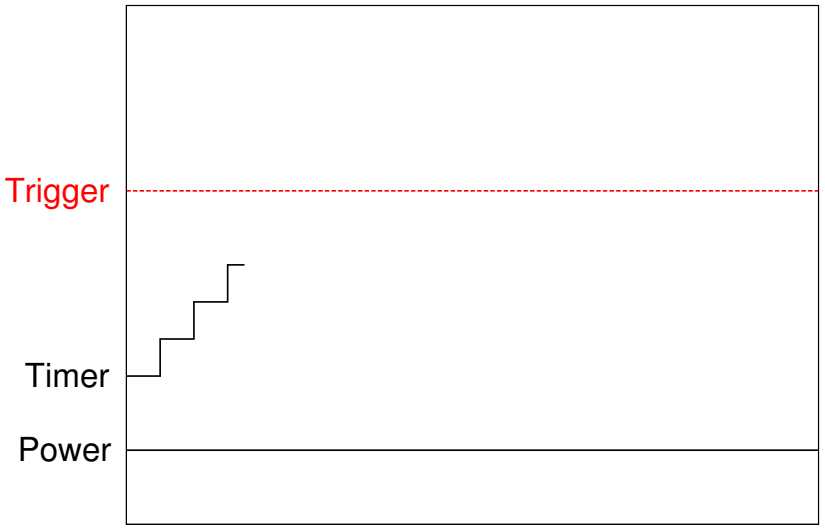




- Deterministic Timer (D-Timer) will trigger at a specified number of clock cycles with probability 1

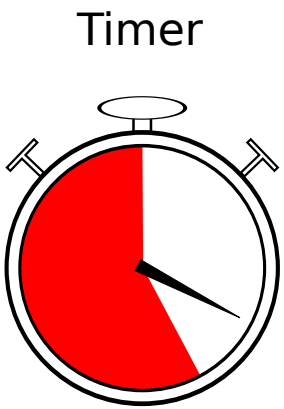
Single Stage D-Timer

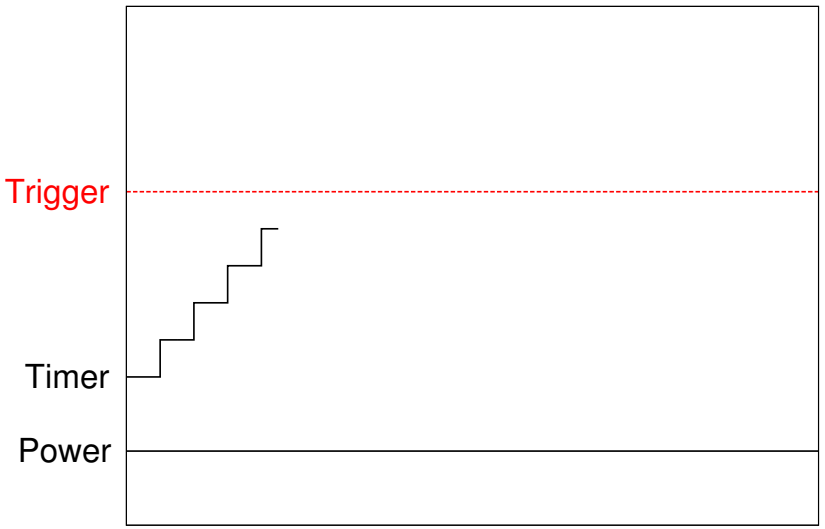




- Deterministic Timer (D-Timer) will trigger at a specified number of clock cycles with probability 1

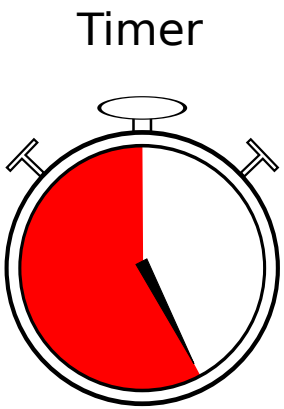
Single Stage D-Timer

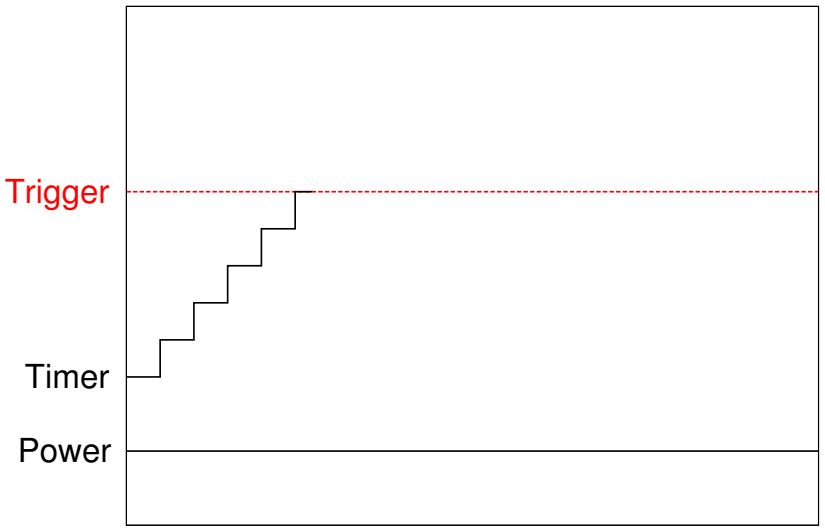




- Deterministic Timer (D-Timer) will trigger at a specified number of clock cycles with probability 1

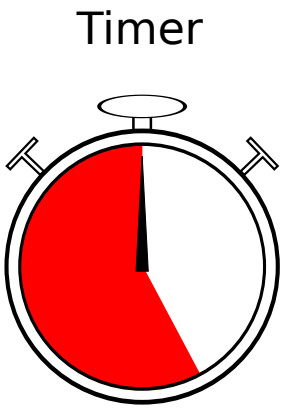
Single Stage D-Timer

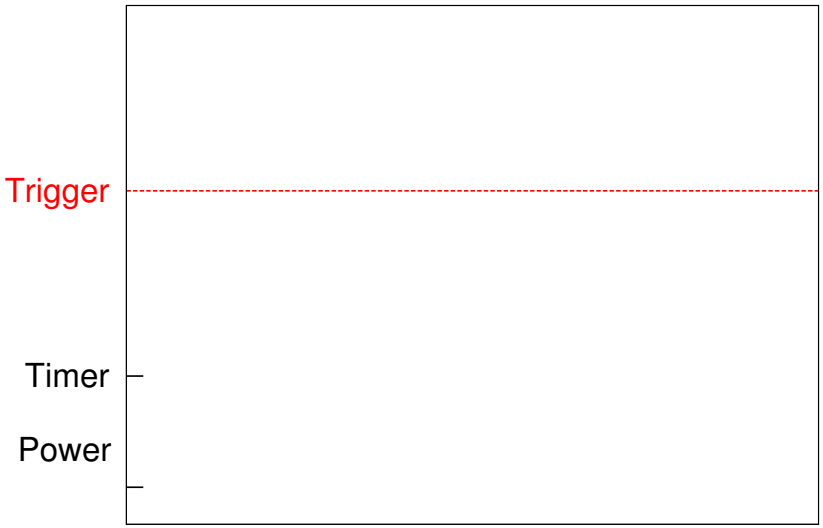




- Deterministic Timer (D-Timer) will trigger at a specified number of clock cycles with probability 1

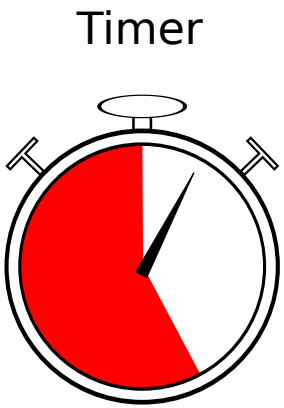
Single Stage D-Timer

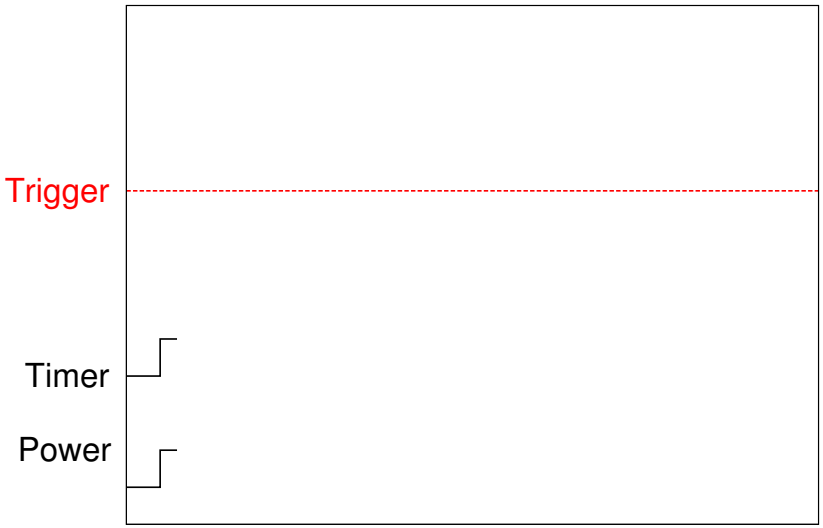




- Deterministic Timer (D-Timer) will trigger at a specified number of clock cycles with probability 1

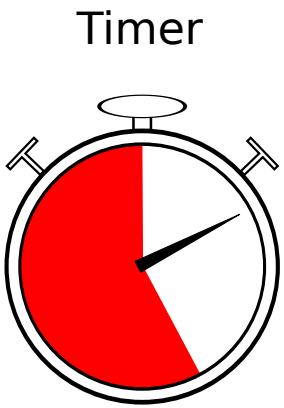
Single Stage D-Timer

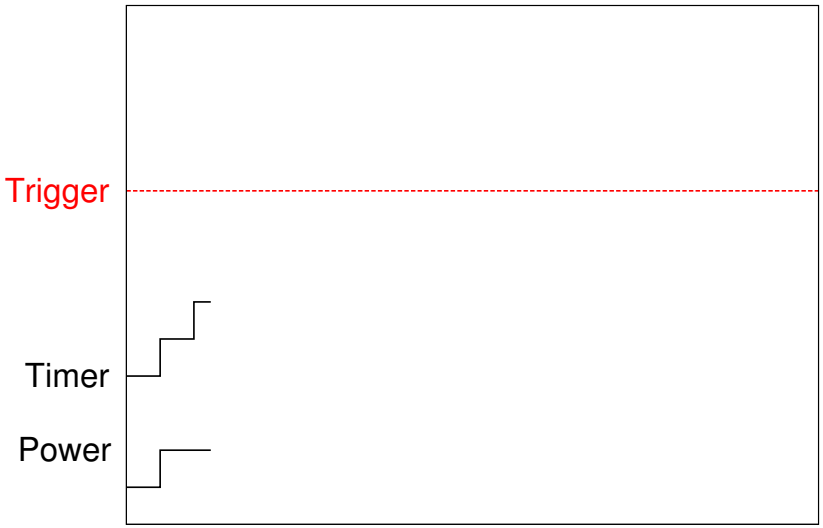




- Deterministic Timer (D-Timer) will trigger at a specified number of clock cycles with probability 1

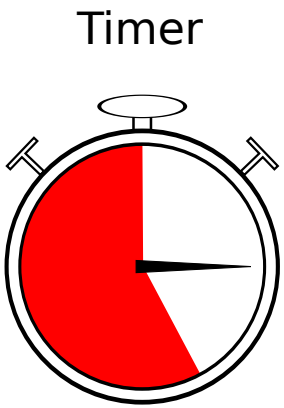
Single Stage D-Timer

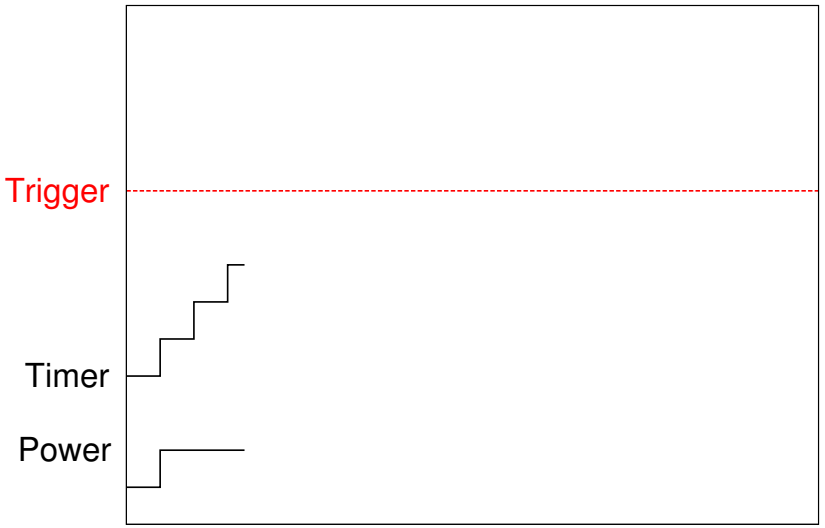




- Deterministic Timer (D-Timer) will trigger at a specified number of clock cycles with probability 1

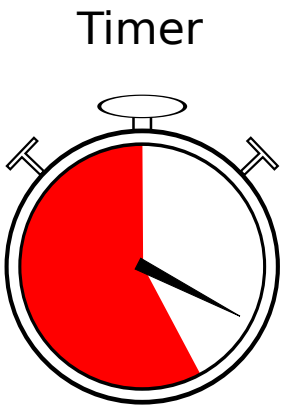
Single Stage D-Timer

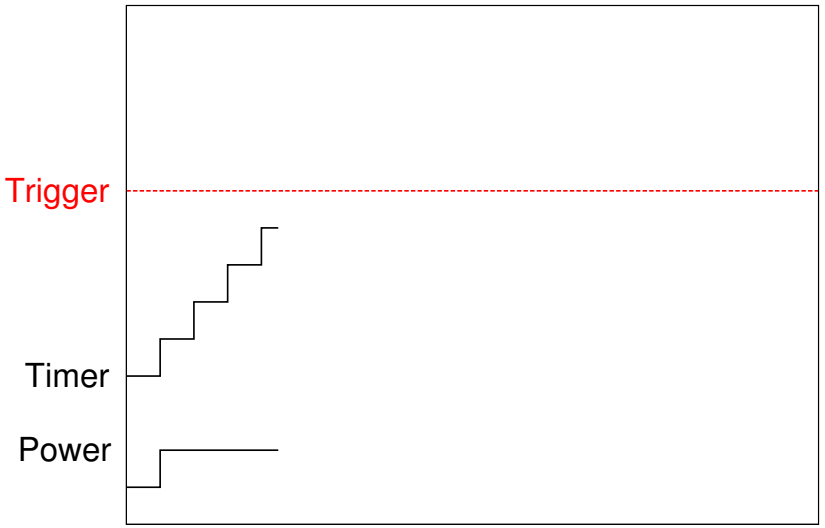




- Deterministic Timer (D-Timer) will trigger at a specified number of clock cycles with probability 1

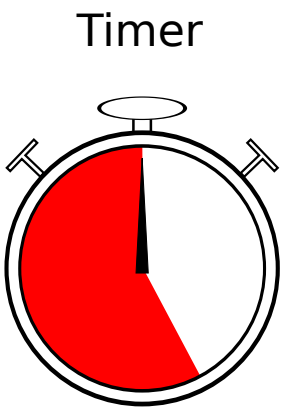
Single Stage D-Timer

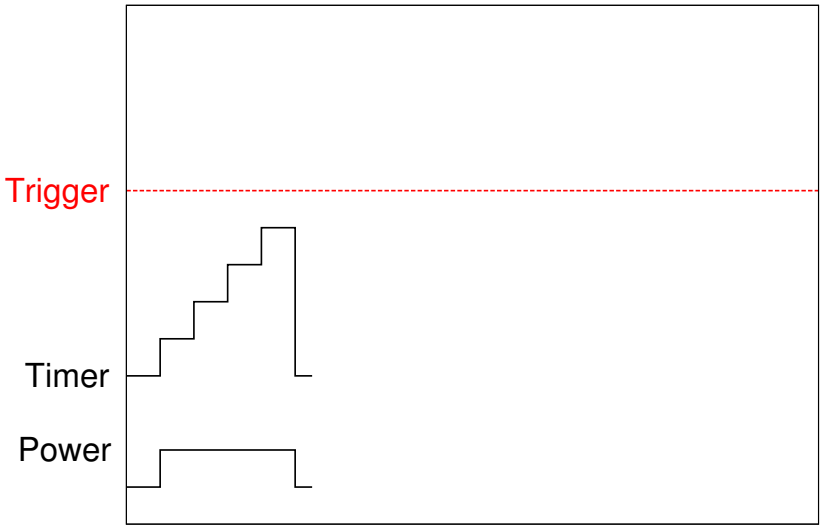




- Deterministic Timer (D-Timer) will trigger at a specified number of clock cycles with probability 1

Single Stage D-Timer

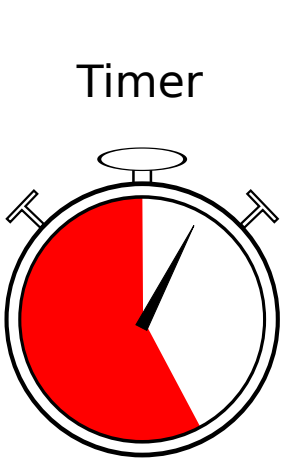


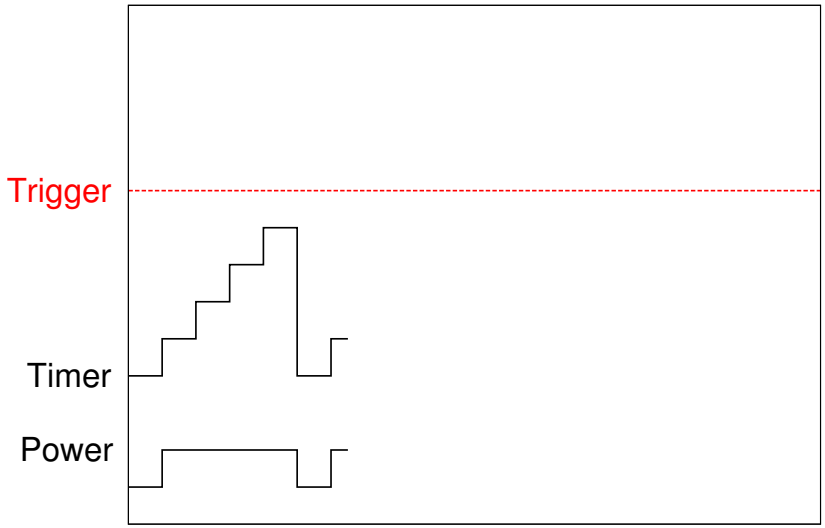


- Deterministic Timer (D-Timer) will trigger at a specified number of clock cycles with probability 1
- Can be defeated with power cycling [7]

[7] Waksman and Sethumadhavan, Silencing Hardware Backdoors
IEEE Symposium on Security and Privacy

Single Stage D-Timer

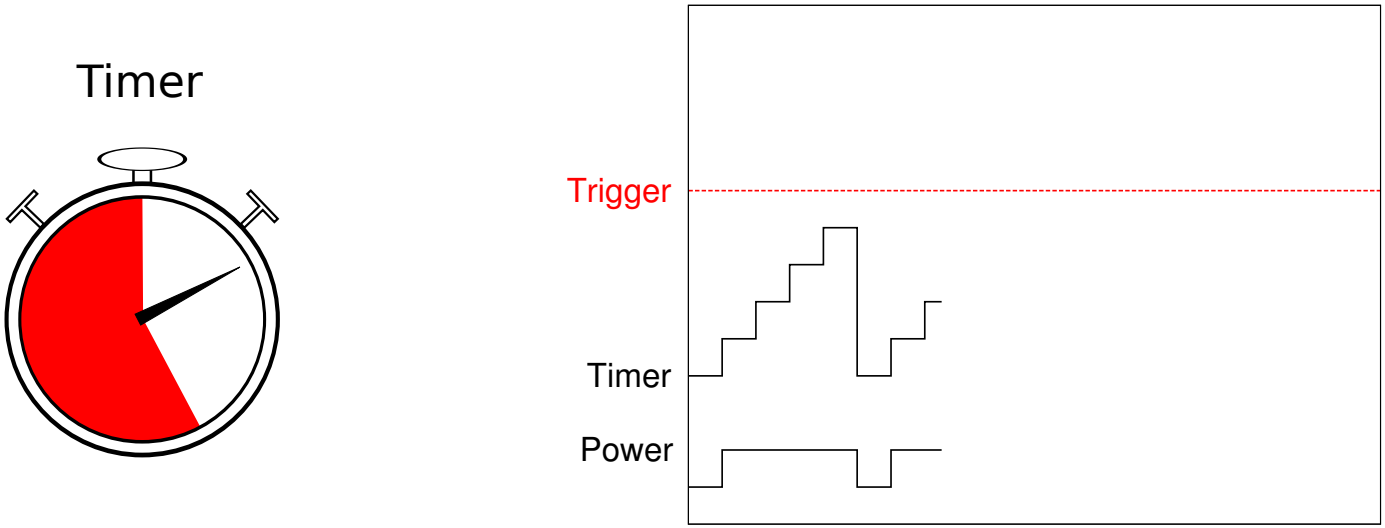




- Deterministic Timer (D-Timer) will trigger at a specified number of clock cycles with probability 1
- Can be defeated with power cycling [7]

[7] Waksman and Sethumadhavan, Silencing Hardware Backdoors
IEEE Symposium on Security and Privacy

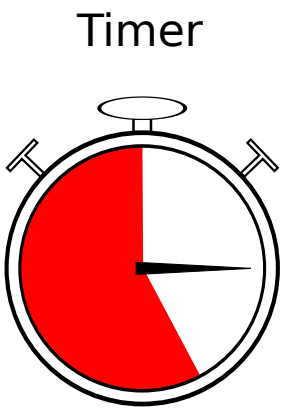
Single Stage D-Timer

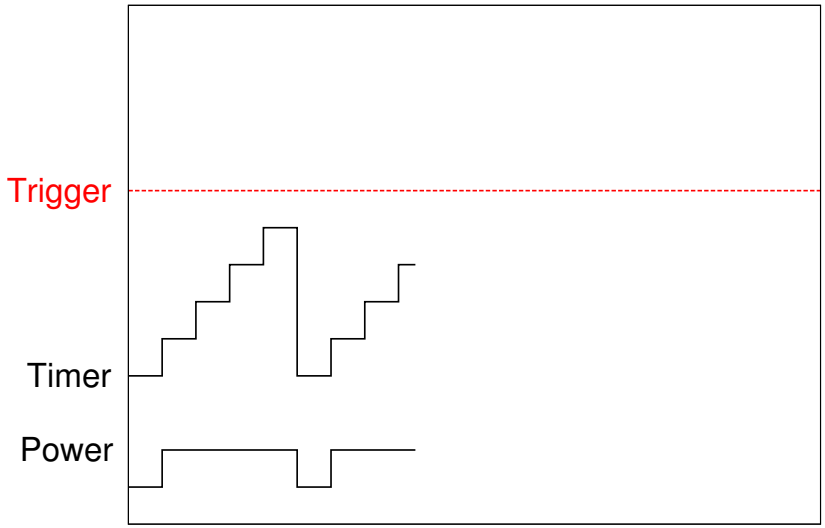


- Deterministic Timer (D-Timer) will trigger at a specified number of clock cycles with probability 1
- Can be defeated with power cycling [7]

[7] Waksman and Sethumadhavan, Silencing Hardware Backdoors
IEEE Symposium on Security and Privacy

Single Stage D-Timer

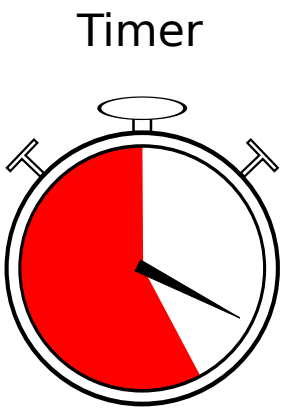


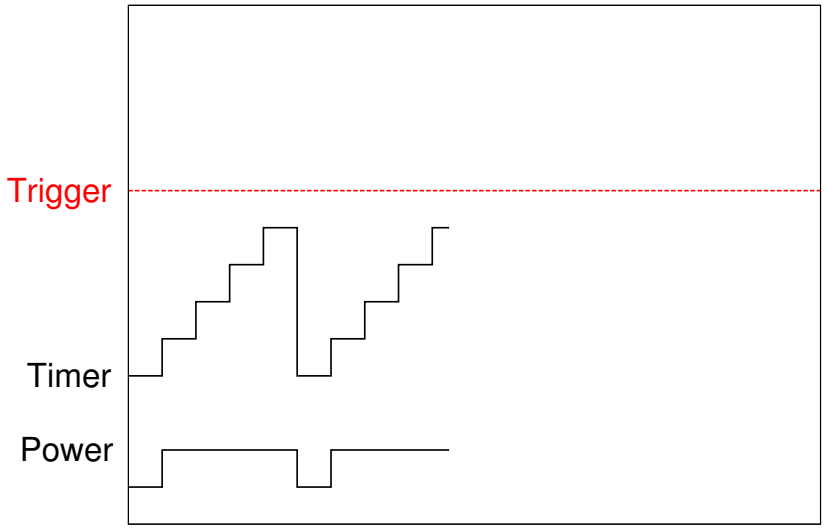


- Deterministic Timer (D-Timer) will trigger at a specified number of clock cycles with probability 1
- Can be defeated with power cycling [7]

[7] Waksman and Sethumadhavan, Silencing Hardware Backdoors
IEEE Symposium on Security and Privacy

Single Stage D-Timer

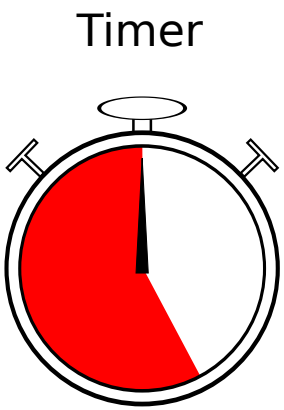


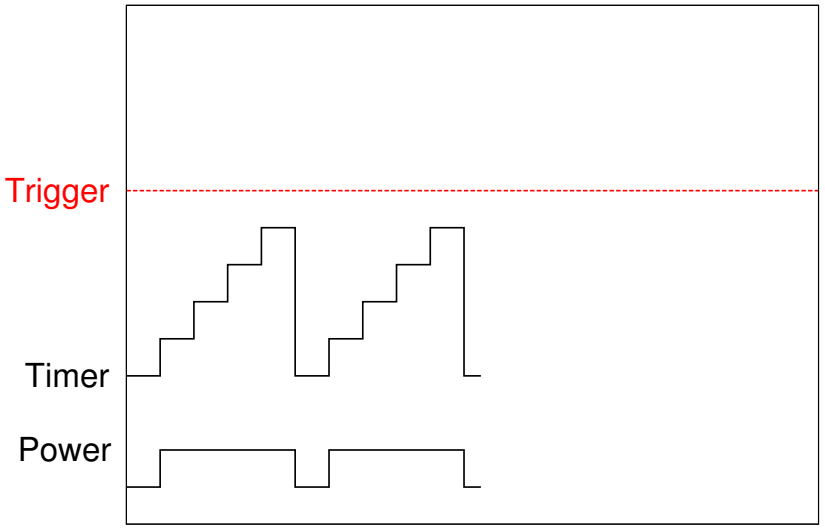


- Deterministic Timer (D-Timer) will trigger at a specified number of clock cycles with probability 1
- Can be defeated with power cycling [7]

[7] Waksman and Sethumadhavan, Silencing Hardware Backdoors
IEEE Symposium on Security and Privacy

Single Stage D-Timer

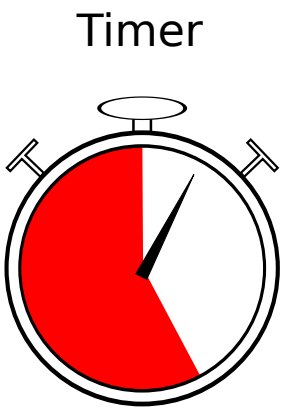


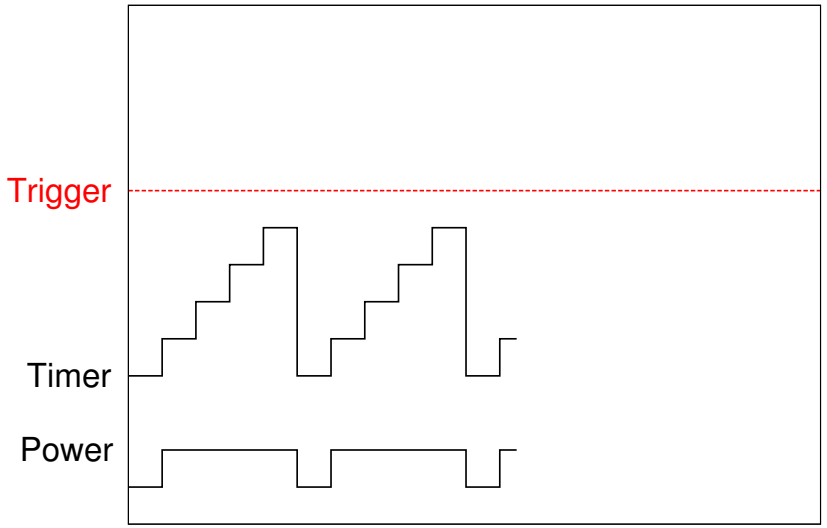


- Deterministic Timer (D-Timer) will trigger at a specified number of clock cycles with probability 1
- Can be defeated with power cycling [7]

[7] Waksman and Sethumadhavan, Silencing Hardware Backdoors
IEEE Symposium on Security and Privacy

Single Stage D-Timer

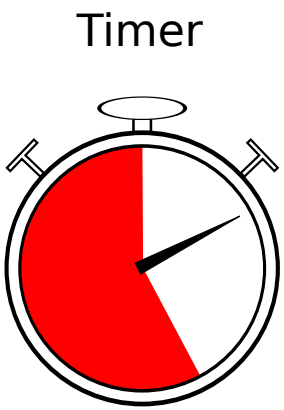


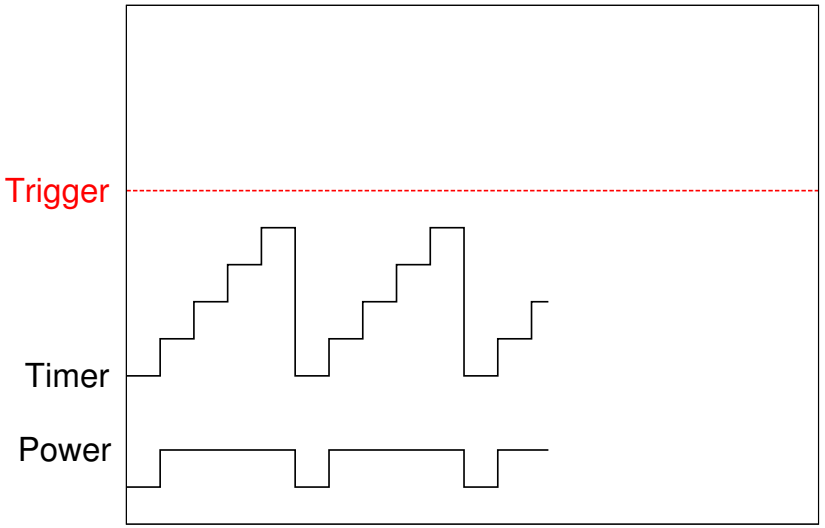


- Deterministic Timer (D-Timer) will trigger at a specified number of clock cycles with probability 1
- Can be defeated with power cycling [7]

[7] Waksman and Sethumadhavan, Silencing Hardware Backdoors
IEEE Symposium on Security and Privacy

Single Stage D-Timer

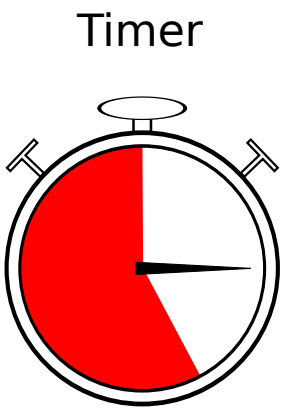


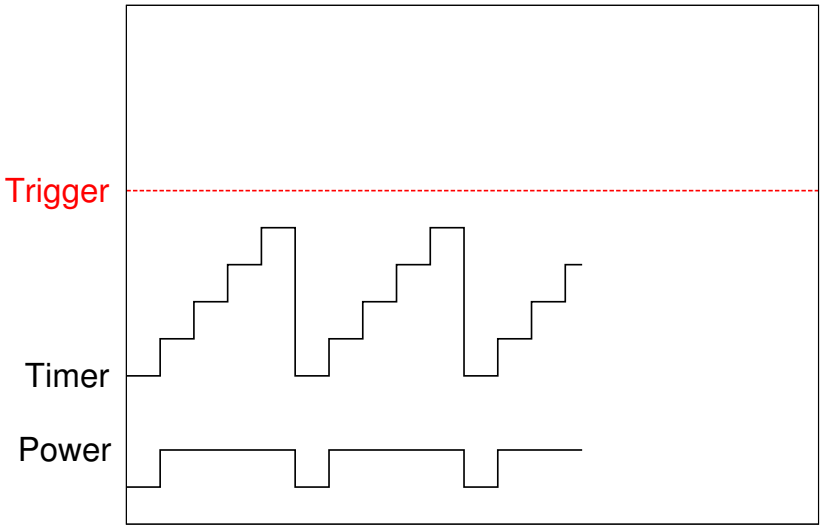


- Deterministic Timer (D-Timer) will trigger at a specified number of clock cycles with probability 1
- Can be defeated with power cycling [7]

[7] Waksman and Sethumadhavan, Silencing Hardware Backdoors
IEEE Symposium on Security and Privacy

Single Stage D-Timer

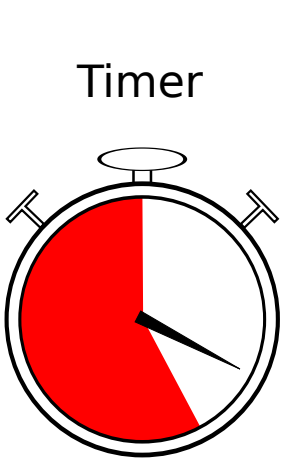


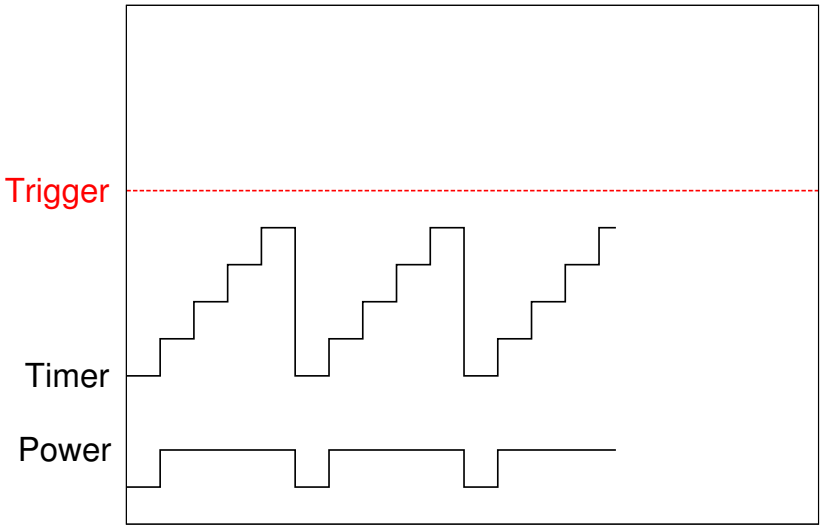


- Deterministic Timer (D-Timer) will trigger at a specified number of clock cycles with probability 1
- Can be defeated with power cycling [7]

[7] Waksman and Sethumadhavan, Silencing Hardware Backdoors
IEEE Symposium on Security and Privacy

Single Stage D-Timer

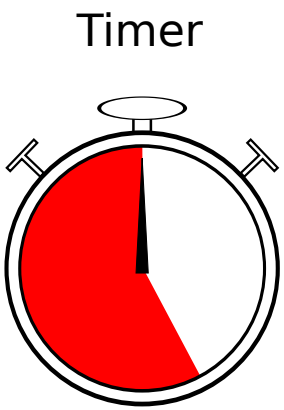


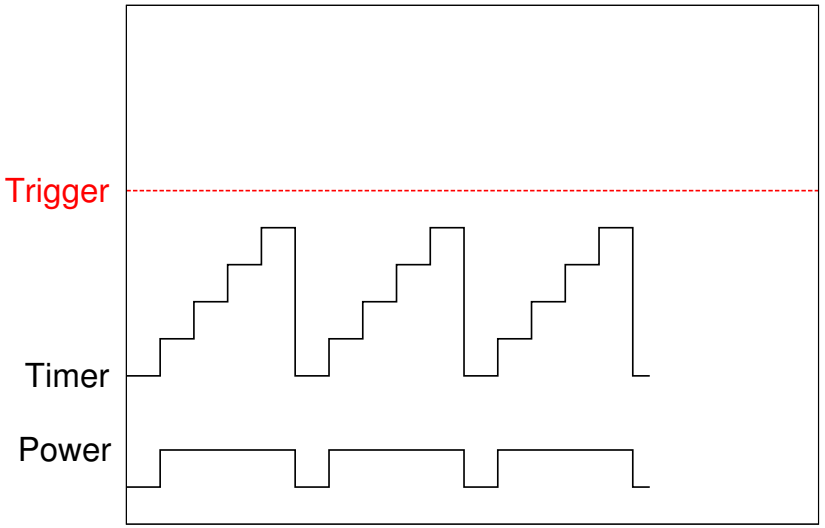


- Deterministic Timer (D-Timer) will trigger at a specified number of clock cycles with probability 1
- Can be defeated with power cycling [7]

[7] Waksman and Sethumadhavan, Silencing Hardware Backdoors
IEEE Symposium on Security and Privacy

Single Stage D-Timer

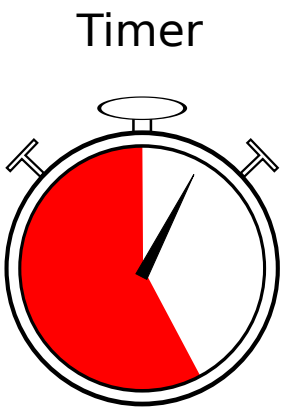


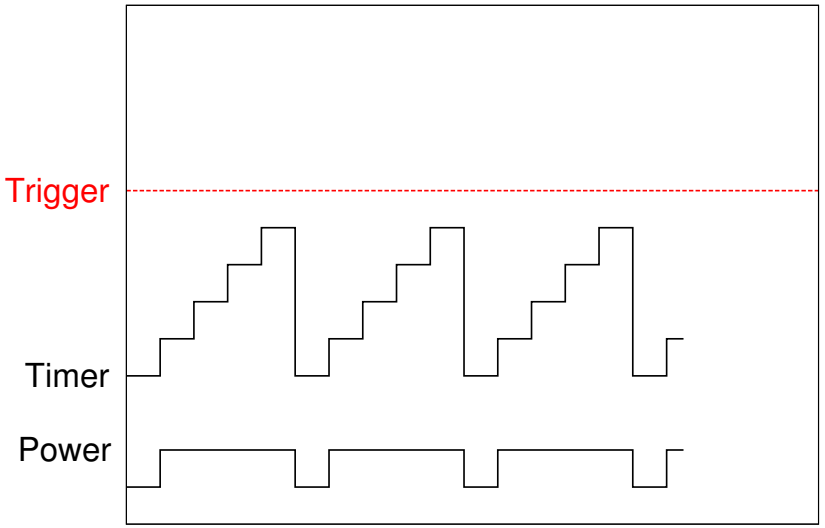


- Deterministic Timer (D-Timer) will trigger at a specified number of clock cycles with probability 1
- Can be defeated with power cycling [7]

[7] Waksman and Sethumadhavan, Silencing Hardware Backdoors
IEEE Symposium on Security and Privacy

Single Stage D-Timer

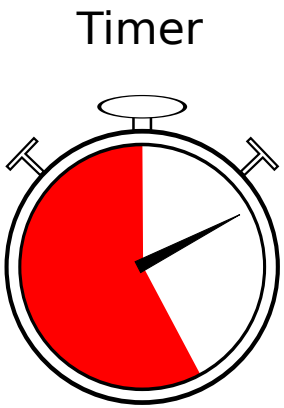


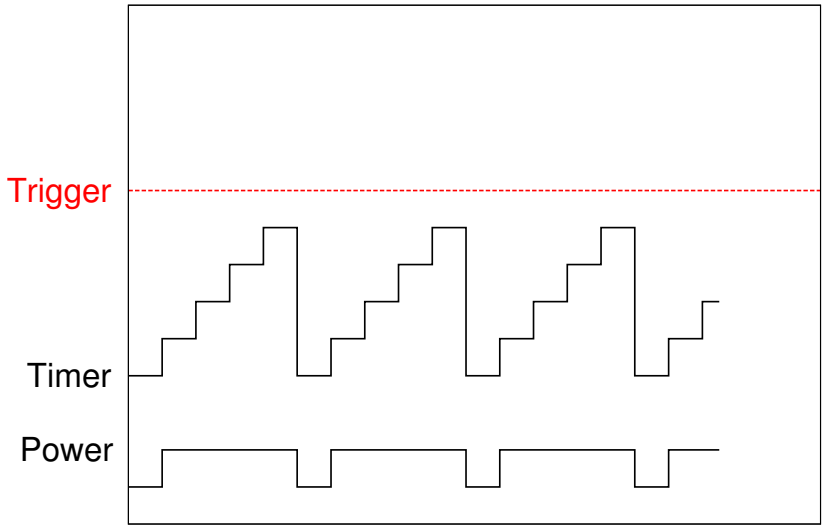


- Deterministic Timer (D-Timer) will trigger at a specified number of clock cycles with probability 1
- Can be defeated with power cycling [7]

[7] Waksman and Sethumadhavan, Silencing Hardware Backdoors
IEEE Symposium on Security and Privacy

Single Stage D-Timer

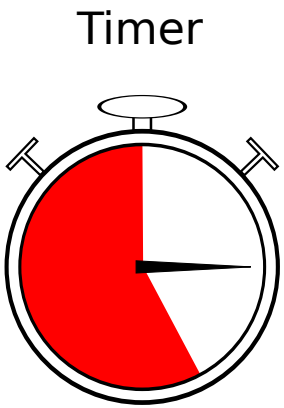


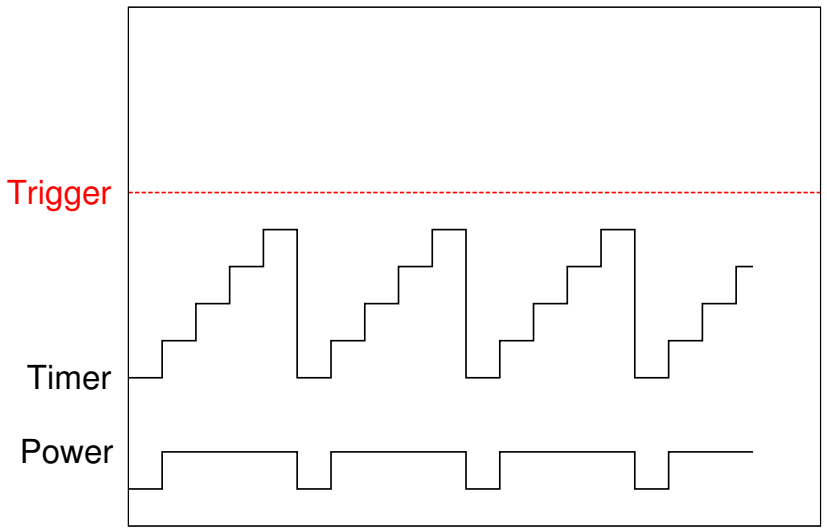


- Deterministic Timer (D-Timer) will trigger at a specified number of clock cycles with probability 1
- Can be defeated with power cycling [7]

[7] Waksman and Sethumadhavan, Silencing Hardware Backdoors
IEEE Symposium on Security and Privacy

Single Stage D-Timer

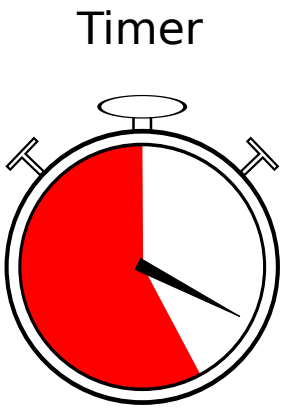


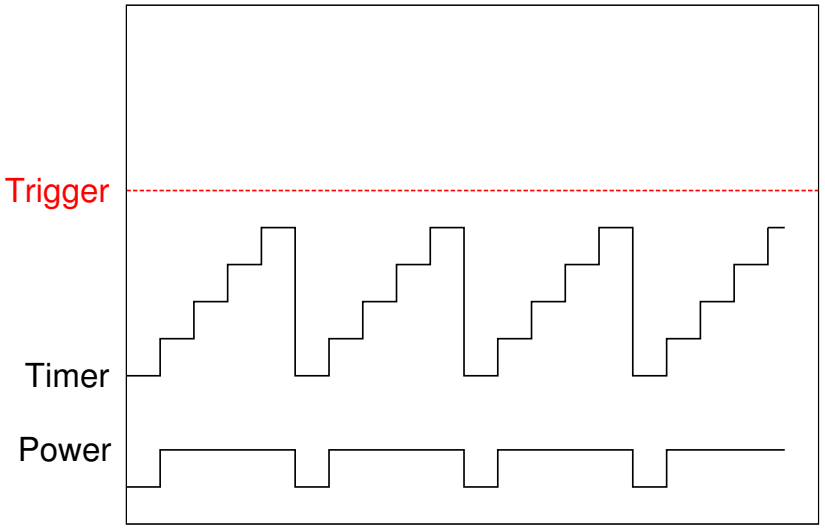


- Deterministic Timer (D-Timer) will trigger at a specified number of clock cycles with probability 1
- Can be defeated with power cycling [7]

[7] Waksman and Sethumadhavan, Silencing Hardware Backdoors
IEEE Symposium on Security and Privacy

Single Stage D-Timer

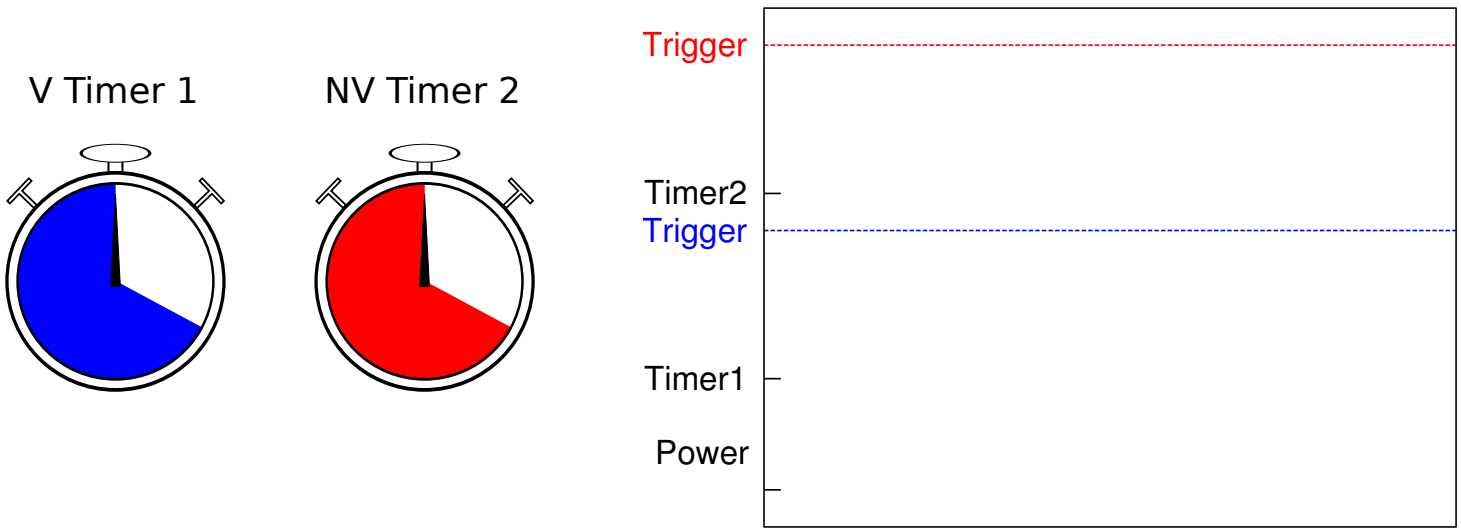




- Deterministic Timer (D-Timer) will trigger at a specified number of clock cycles with probability 1
- Can be defeated with power cycling [7]

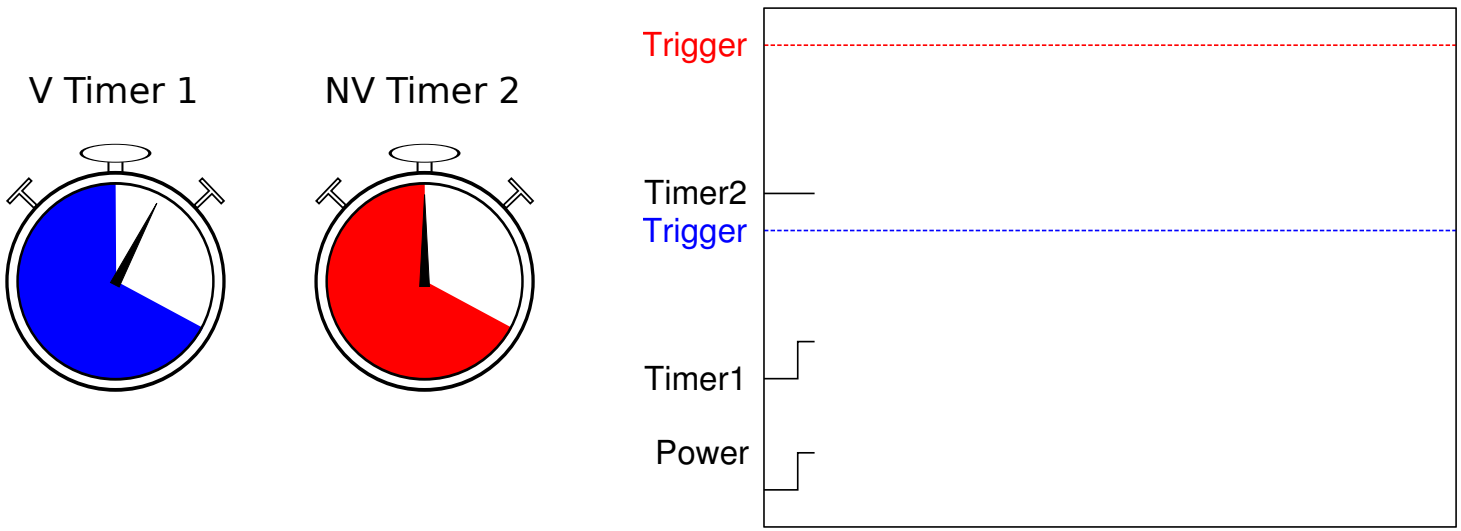
[7] Waksman and Sethumadhavan, Silencing Hardware Backdoors
IEEE Symposium on Security and Privacy

Dual Stage D-Timer



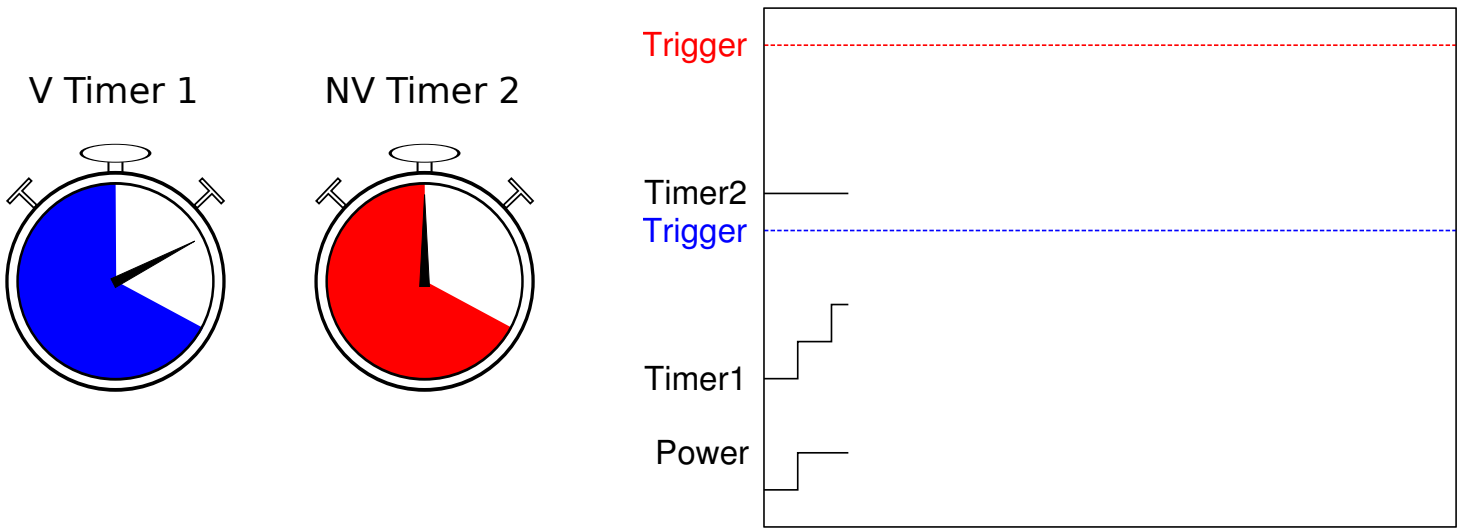
- Dual stage timer stores 2^{nd} timer in NV memory

Dual Stage D-Timer



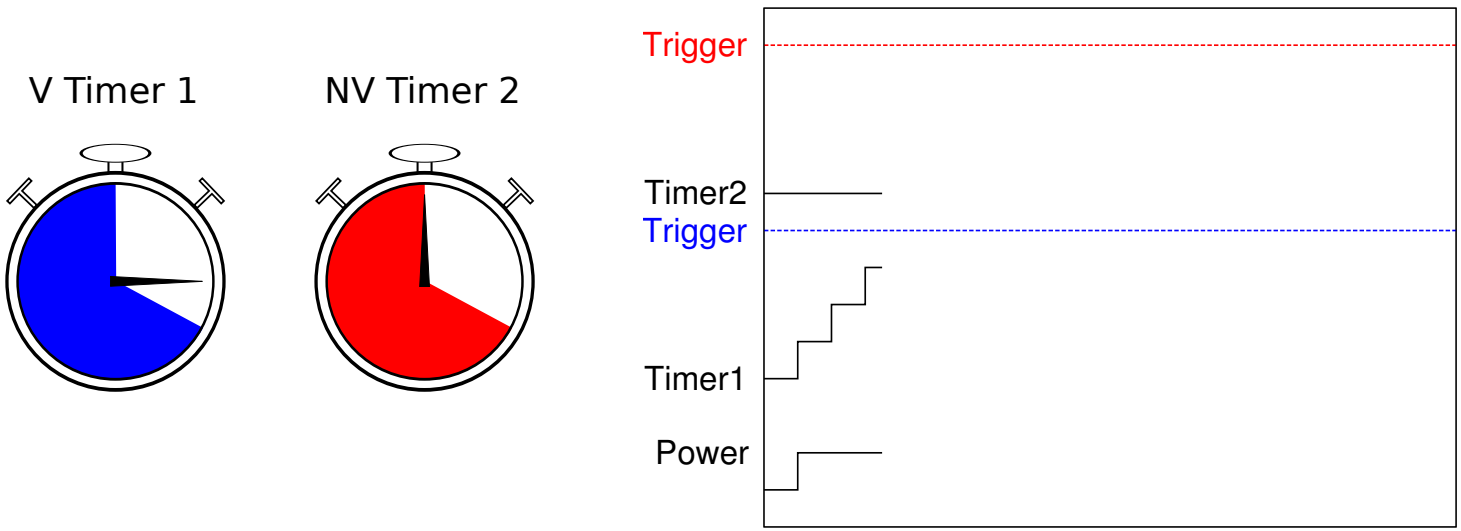
- Dual stage timer stores 2^{nd} timer in NV memory

Dual Stage D-Timer



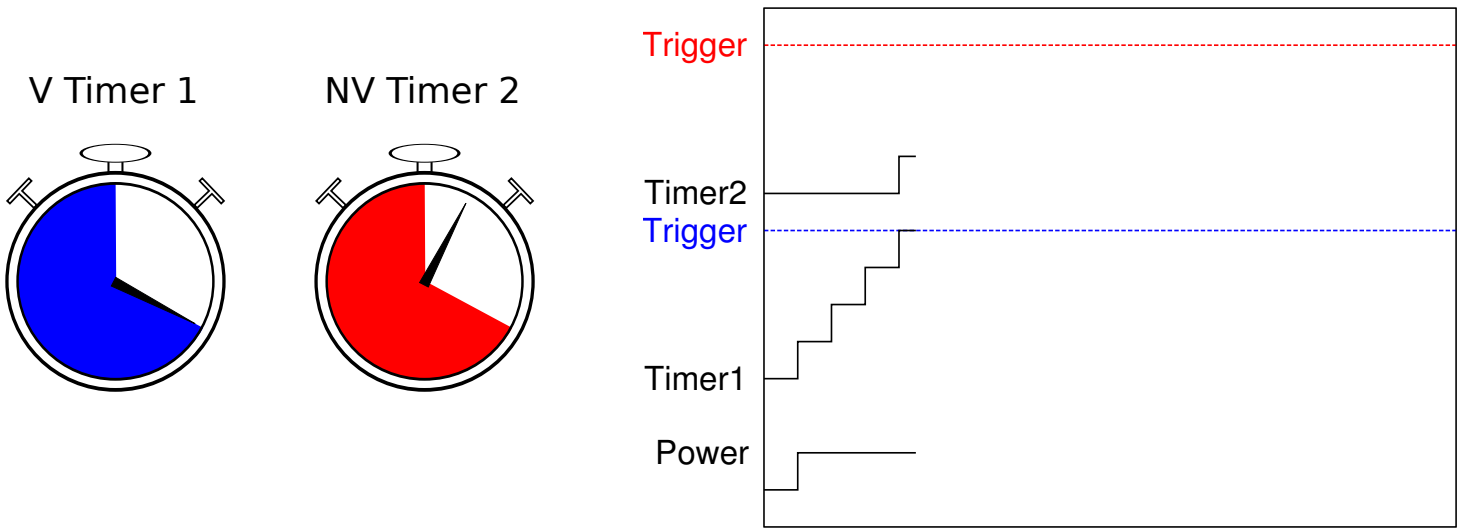
- Dual stage timer stores 2^{nd} timer in NV memory

Dual Stage D-Timer



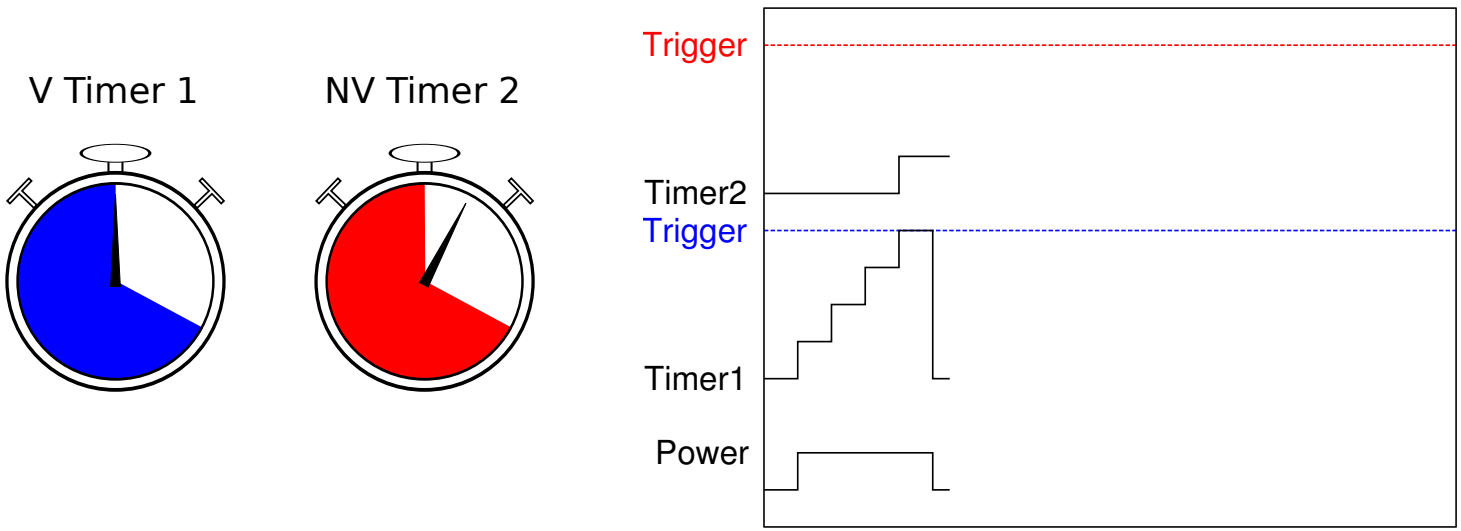
- Dual stage timer stores 2^{nd} timer in NV memory

Dual Stage D-Timer



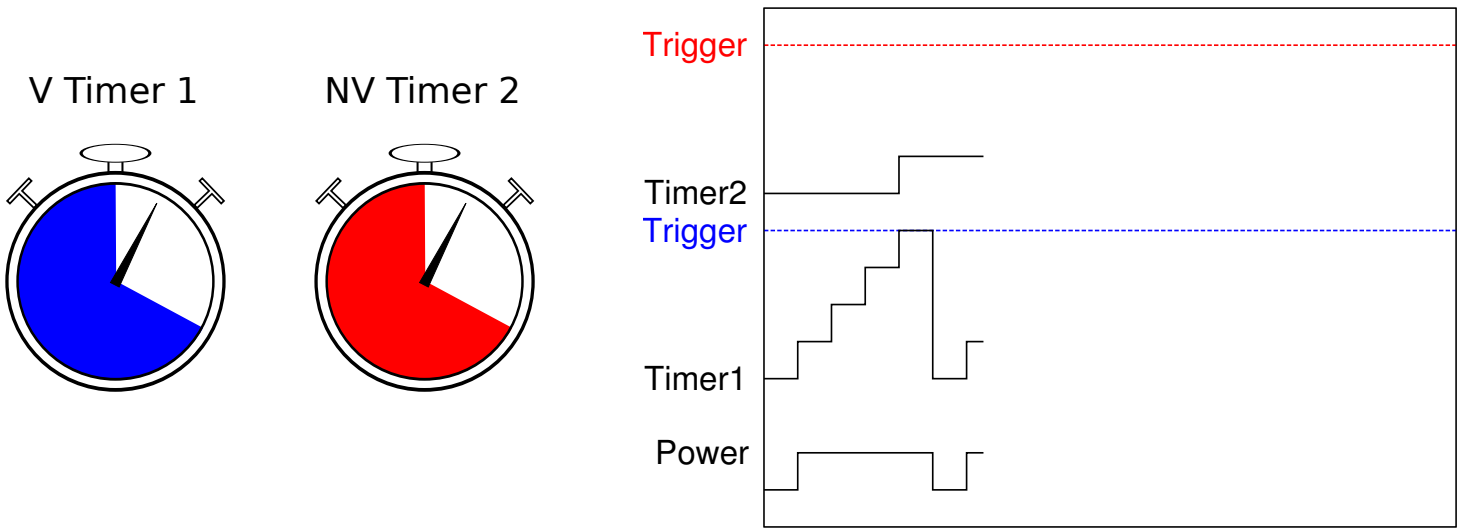
- Dual stage timer stores 2^{nd} timer in NV memory

Dual Stage D-Timer



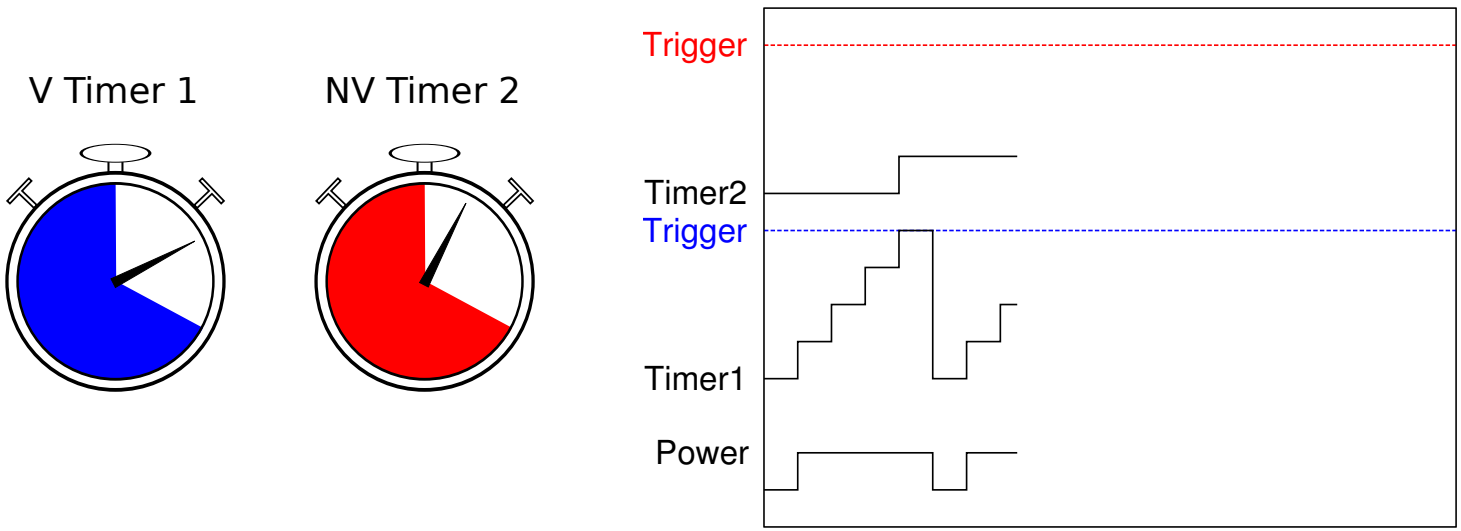
- Dual stage timer stores 2^{nd} timer in NV memory

Dual Stage D-Timer



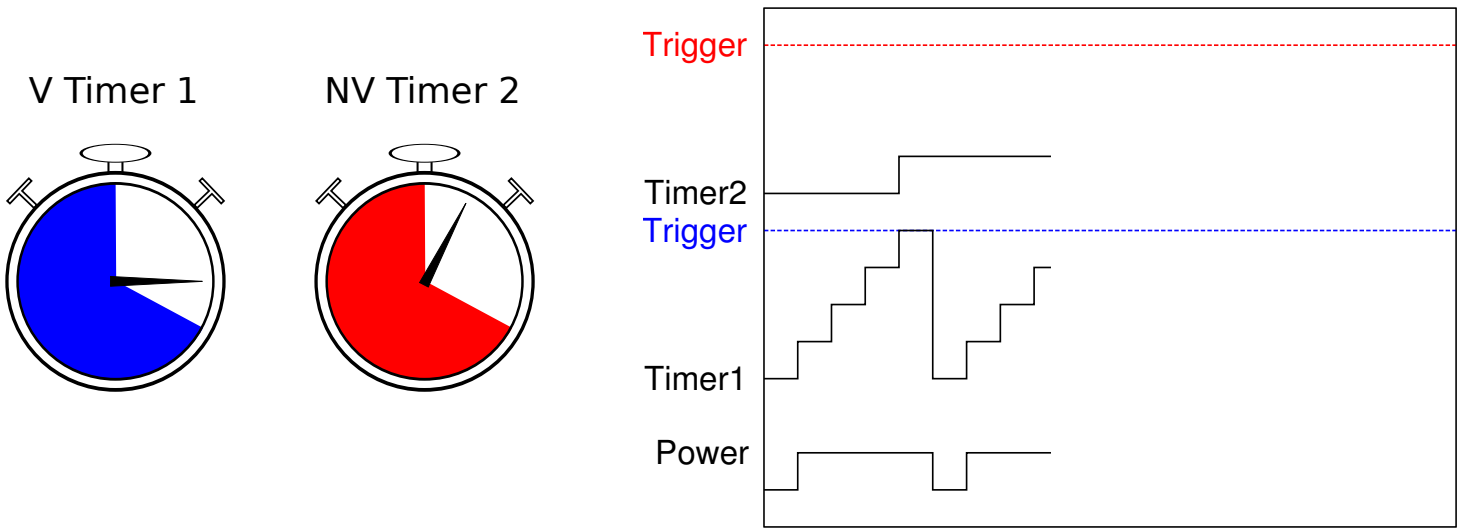
- Dual stage timer stores 2^{nd} timer in NV memory

Dual Stage D-Timer



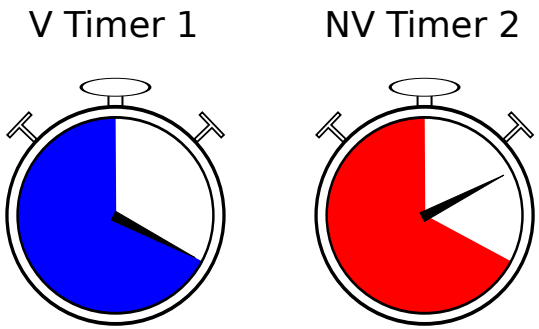
- Dual stage timer stores 2^{nd} timer in NV memory

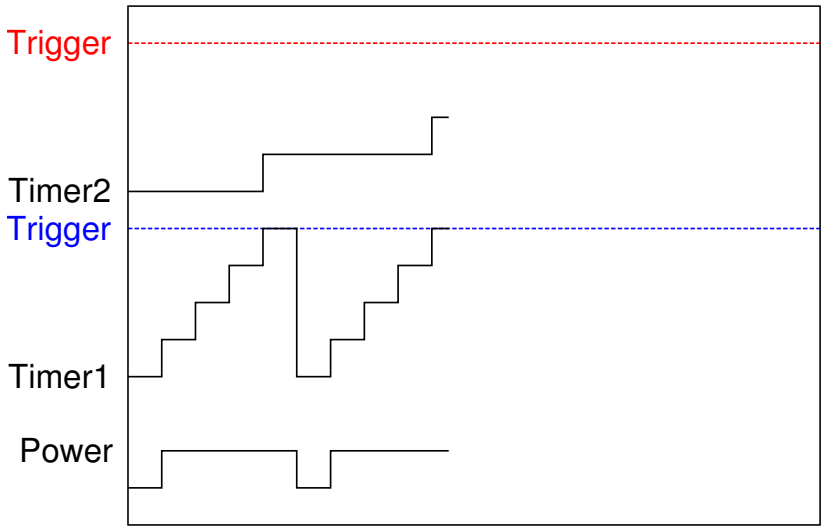
Dual Stage D-Timer



- Dual stage timer stores 2^{nd} timer in NV memory

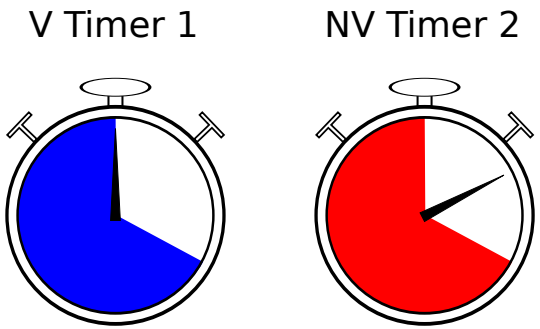
Dual Stage D-Timer

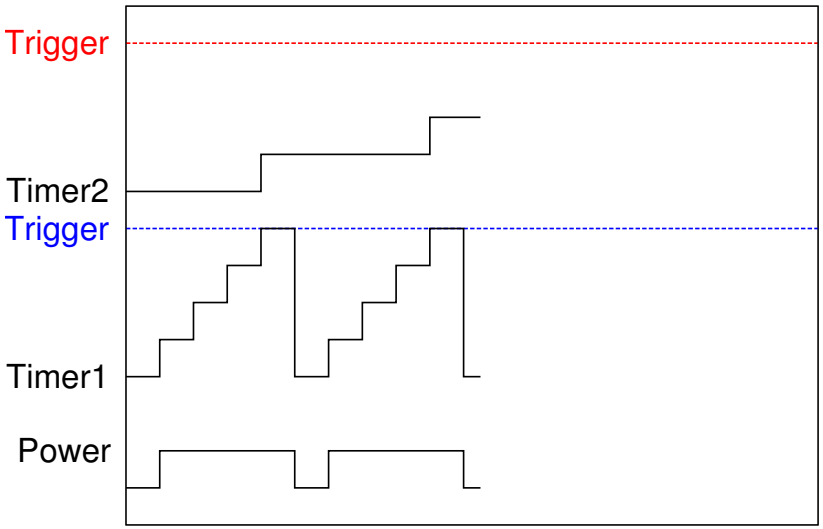




- Dual stage timer stores 2^{nd} timer in NV memory

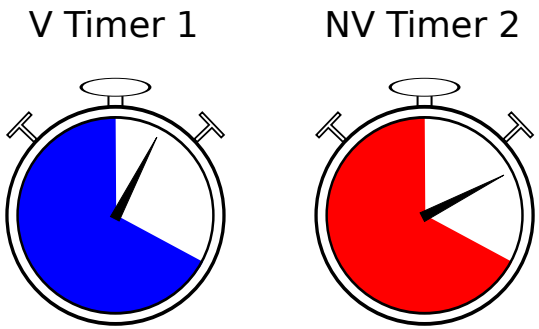
Dual Stage D-Timer

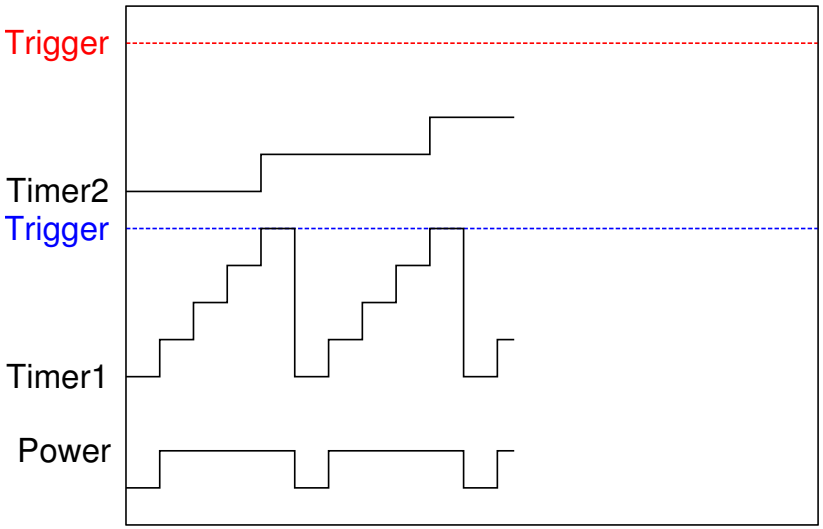




- Dual stage timer stores 2^{nd} timer in NV memory

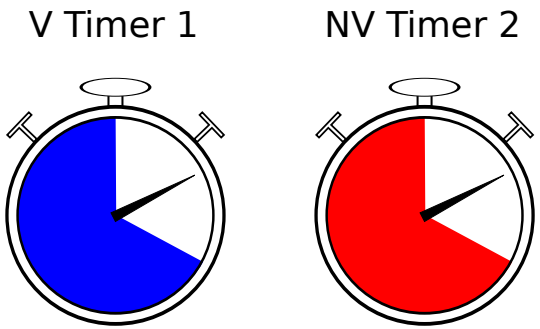
Dual Stage D-Timer

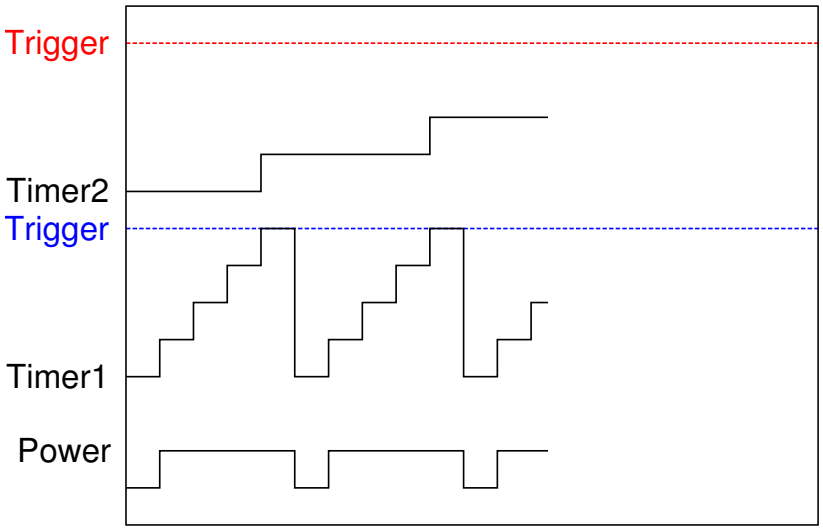




- Dual stage timer stores 2^{nd} timer in NV memory

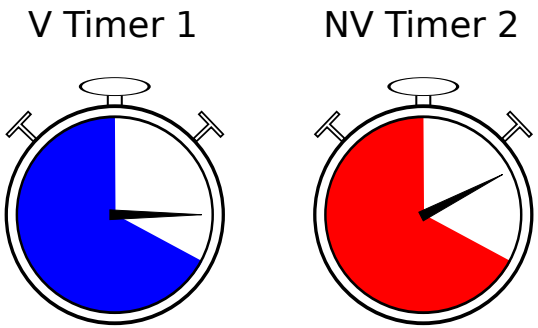
Dual Stage D-Timer

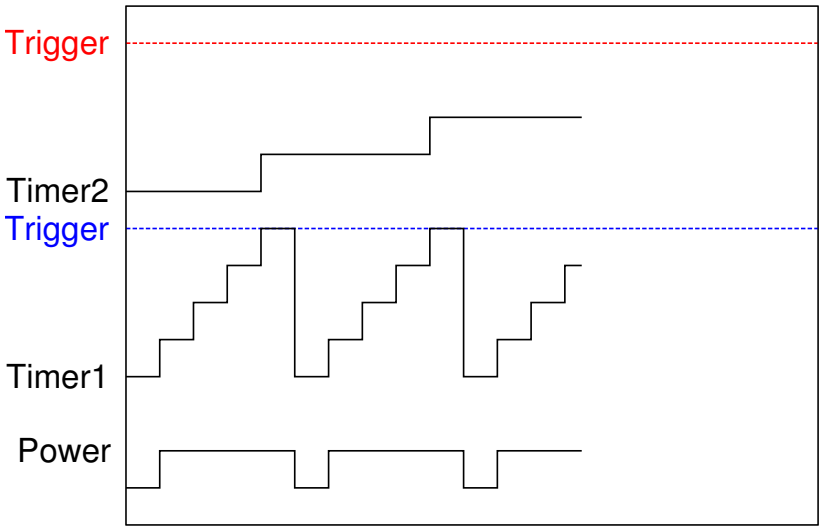




- Dual stage timer stores 2^{nd} timer in NV memory

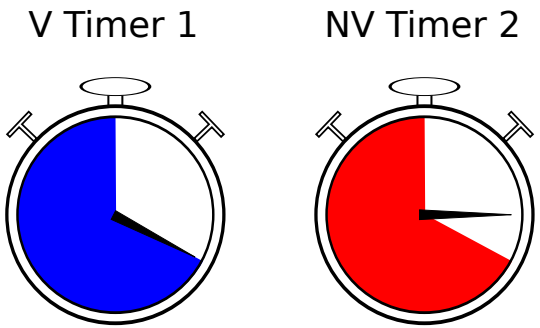
Dual Stage D-Timer

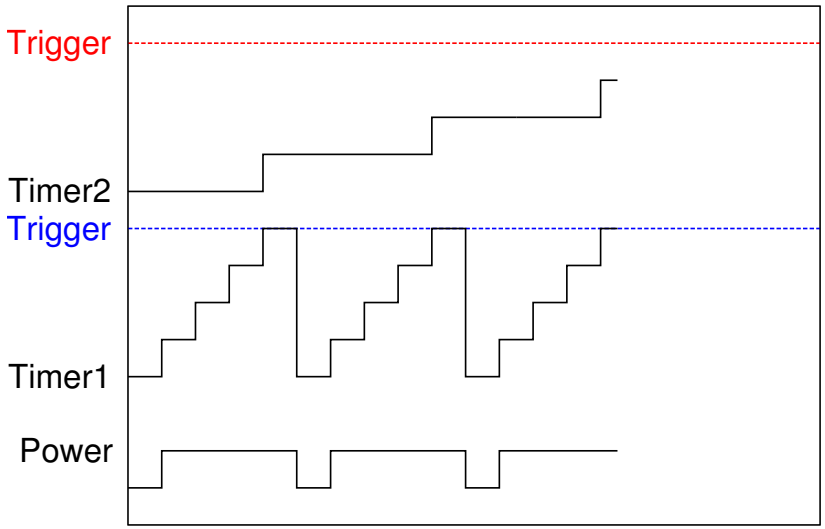




- Dual stage timer stores 2^{nd} timer in NV memory

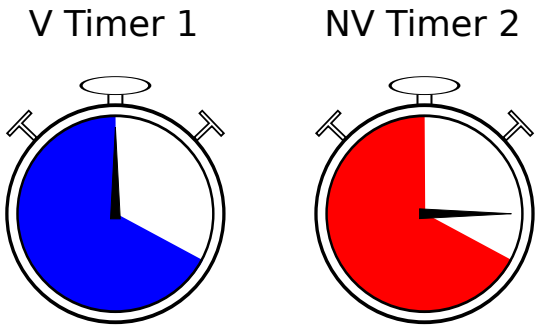
Dual Stage D-Timer

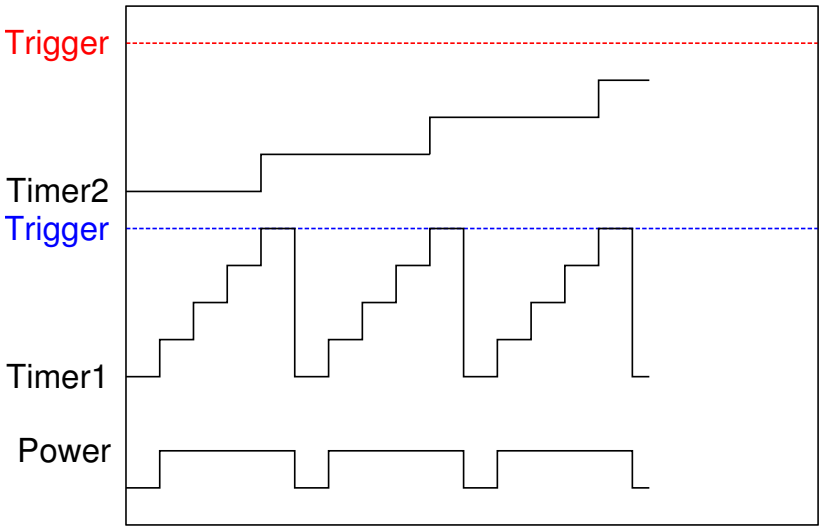




- Dual stage timer stores 2^{nd} timer in NV memory

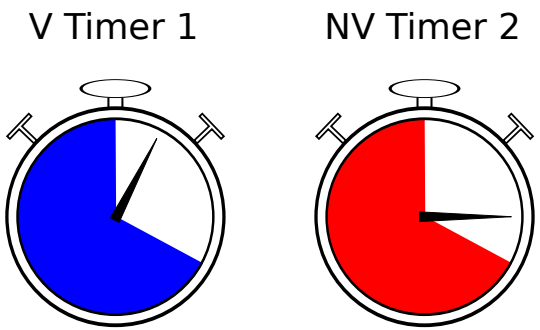
Dual Stage D-Timer

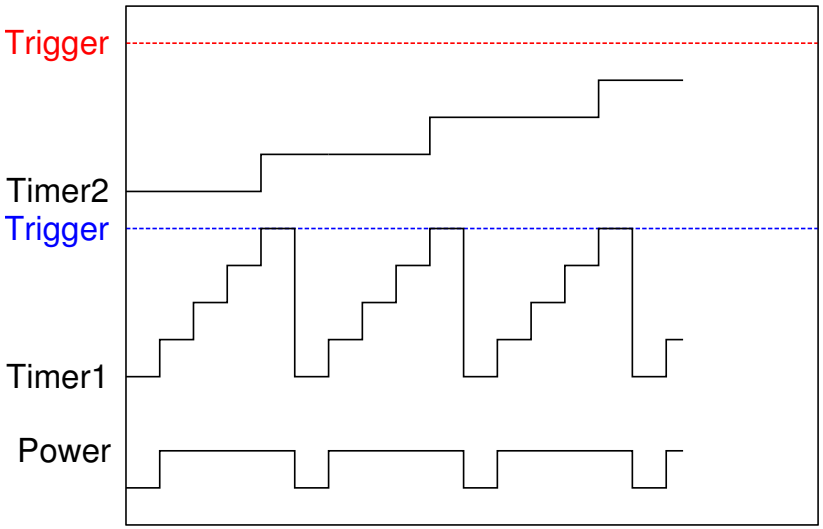




- Dual stage timer stores 2^{nd} timer in NV memory

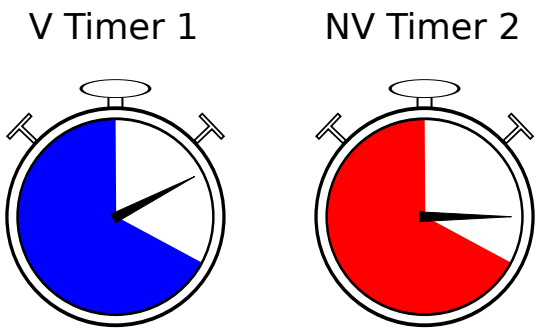
Dual Stage D-Timer

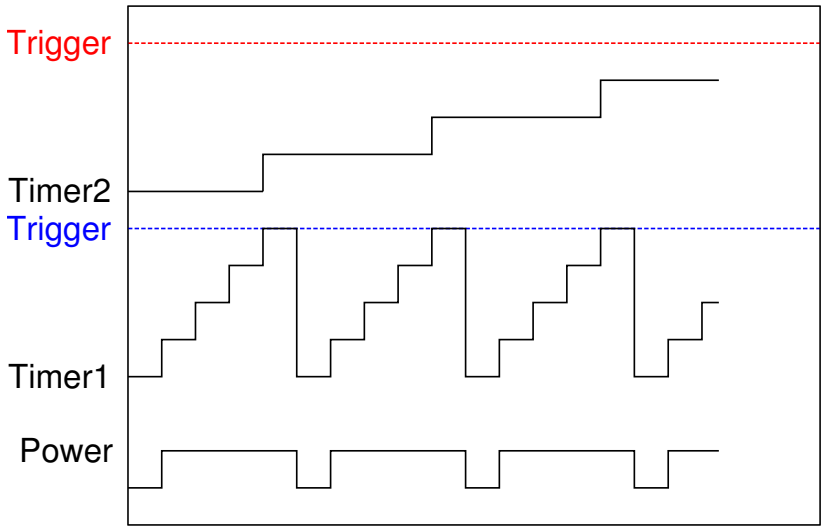




- Dual stage timer stores 2^{nd} timer in NV memory

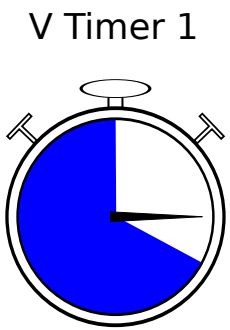
Dual Stage D-Timer

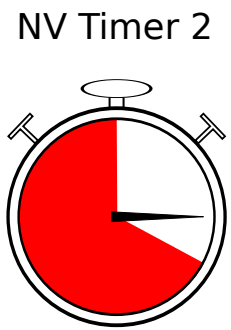


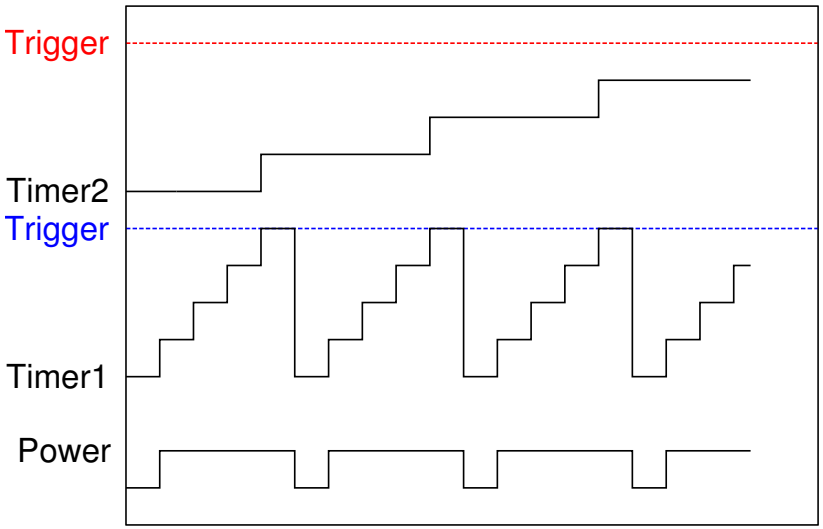


- Dual stage timer stores 2^{nd} timer in NV memory

Dual Stage D-Timer

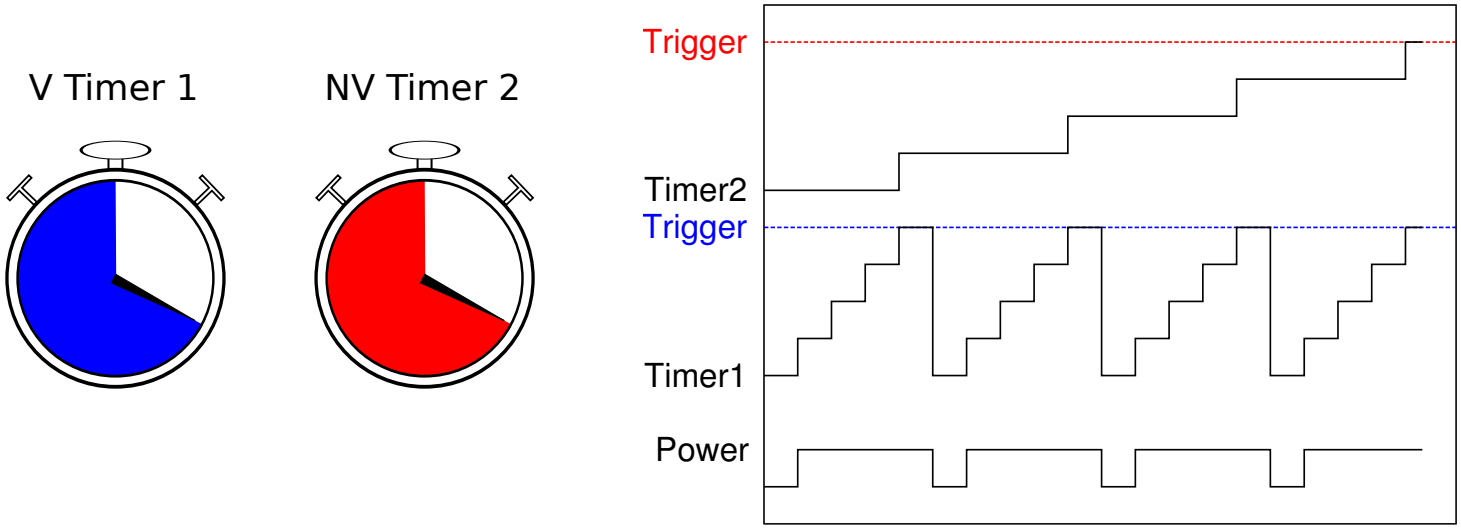






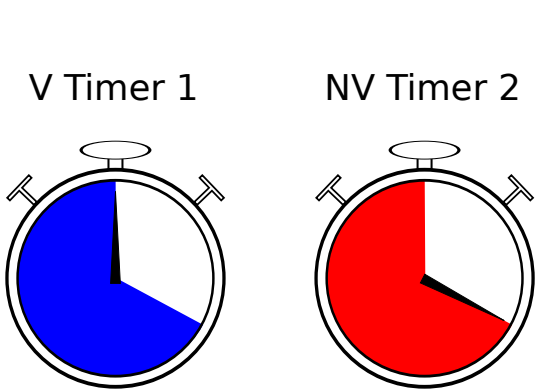
- Dual stage timer stores 2^{nd} timer in NV memory

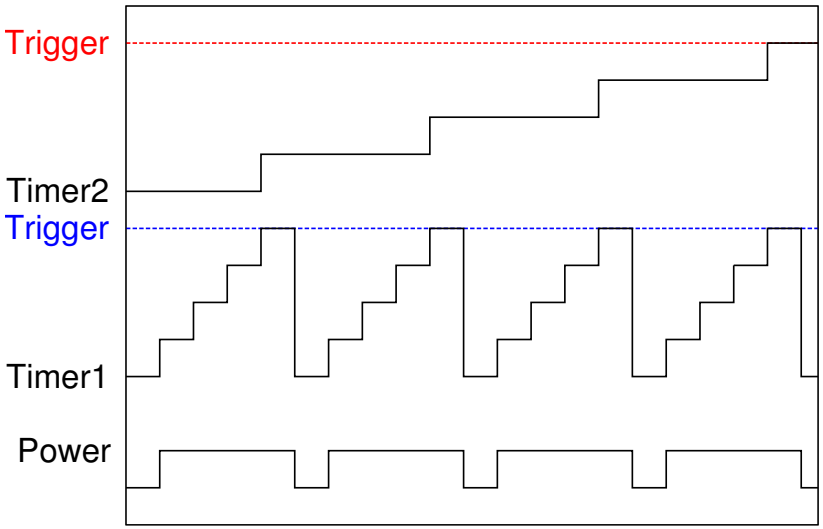
Dual Stage D-Timer



- Dual stage timer stores 2^{nd} timer in NV memory
- This design defeats the power cycling

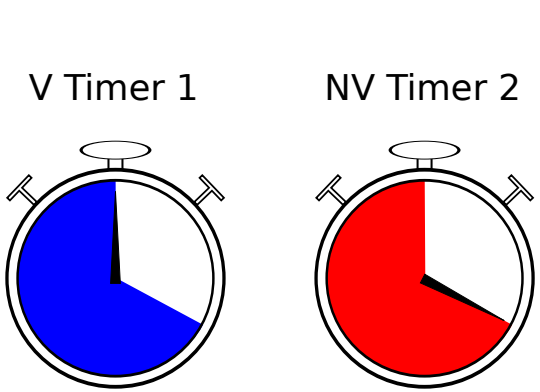
Dual Stage D-Timer

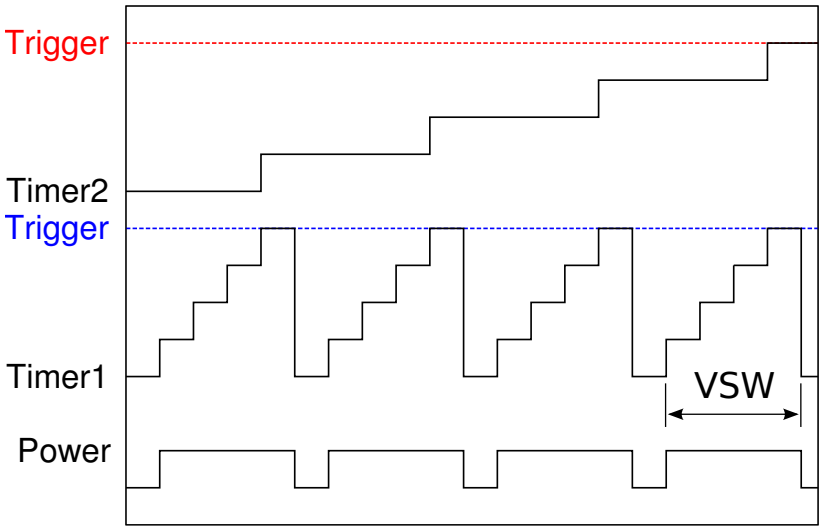




- Dual stage timer stores 2^{nd} timer in NV memory
- This design defeats the power cycling

Dual Stage D-Timer





- Dual stage timer stores 2^{nd} timer in NV memory
- This design defeats the power cycling
- Vulnerable to power cycling with period $\leq$ VSW (Volatile State Window)

Deterministic Timer Drawbacks

Dual Stage D-Timers require frequent writes to NV. Given a write durability w and a timer that counts up to k we need $m \geq \frac{k}{w}$ bits.

A one year timer:

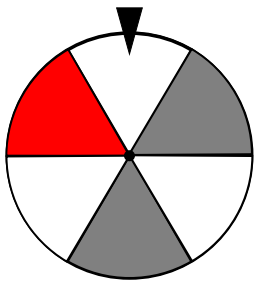
- VSW $= 1$ second
- $m \geq 3153$ NV-bits

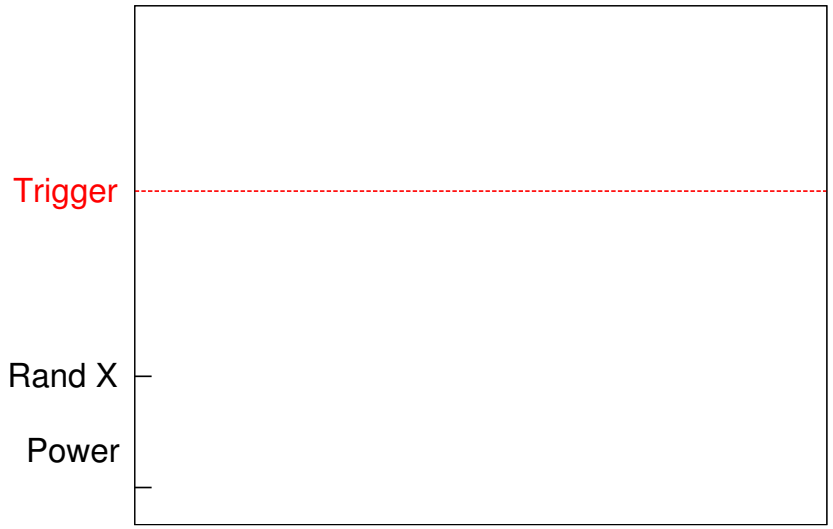
- VSW $= 52$ minutes
- $m = 14$ NV-bits

ND-Timers

1. Intro
   - Attack Model
   - Background
2. Deterministic Timers
   - Hardware
   - Single Stage D-Timer
   - Dual Stage D-Timer
3. ND-Timers
   - Single Shot Timer
   - ND-Timer
4. TRNG
   - Clock Jitter
   - Ring Oscillators
   - Sampling
5. Results
   - Hardware
   - 24 Hour Experiment
6. Detection
   - Hardware
   - Simulation
   - Analysis
7. Summary

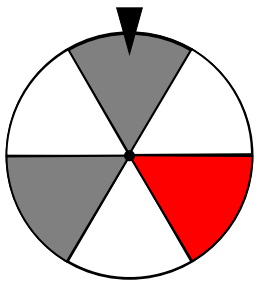
Single Shot Timer

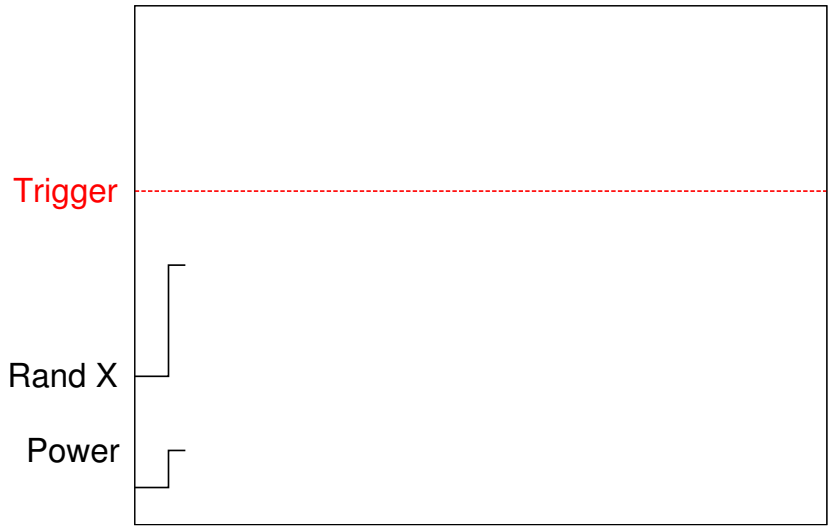




- Not easy to control
- Chance of early detection
- Large Variance

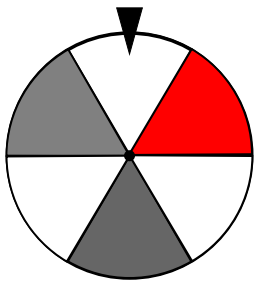
Single Shot Timer

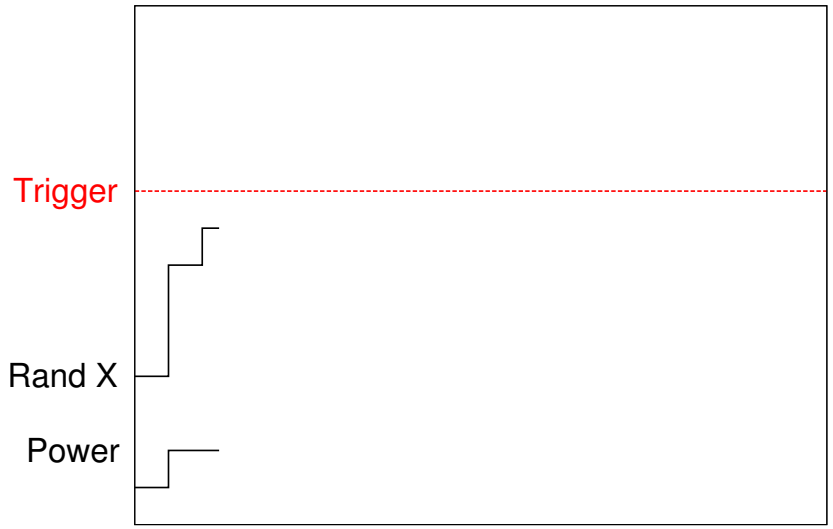




- Not easy to control
- Chance of early detection
- Large Variance

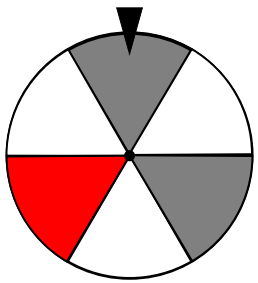
Single Shot Timer

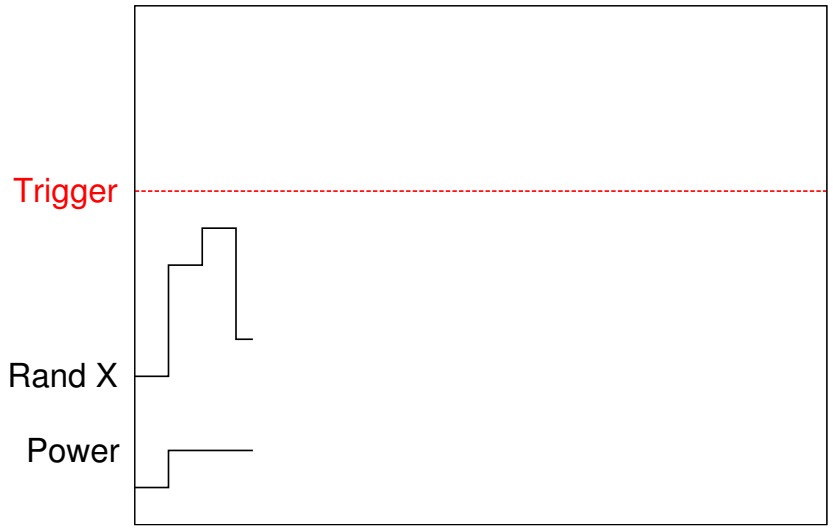




- Not easy to control
- Chance of early detection
- Large Variance

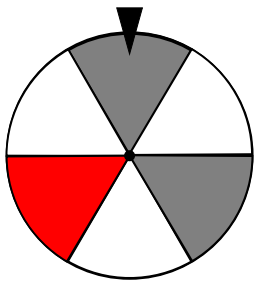
Single Shot Timer

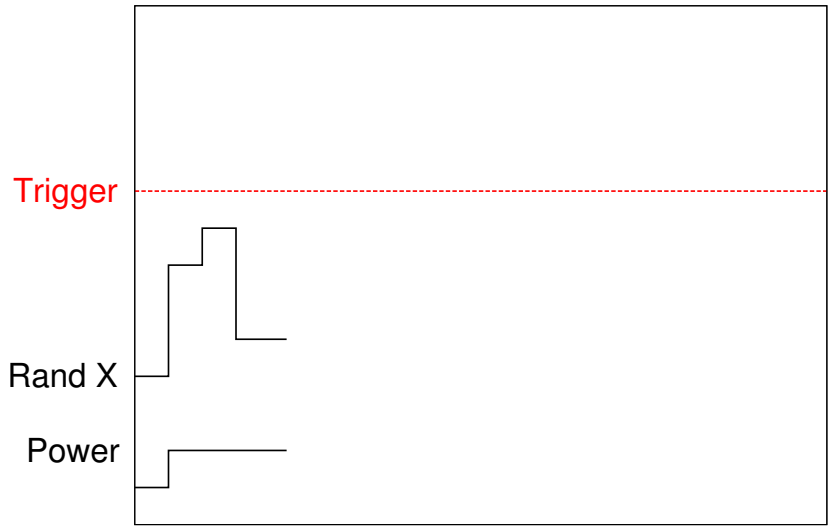




- Not easy to control
- Chance of early detection
- Large Variance

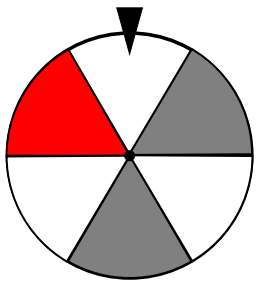
Single Shot Timer

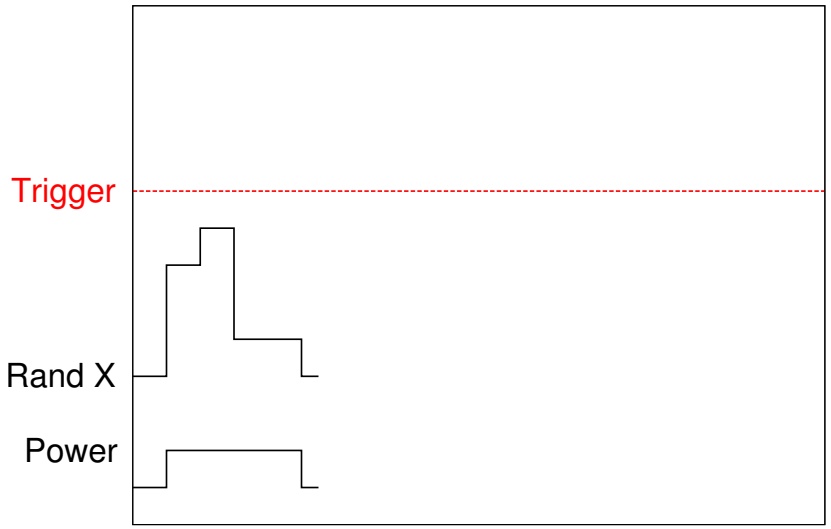




- Not easy to control
- Chance of early detection
- Large Variance

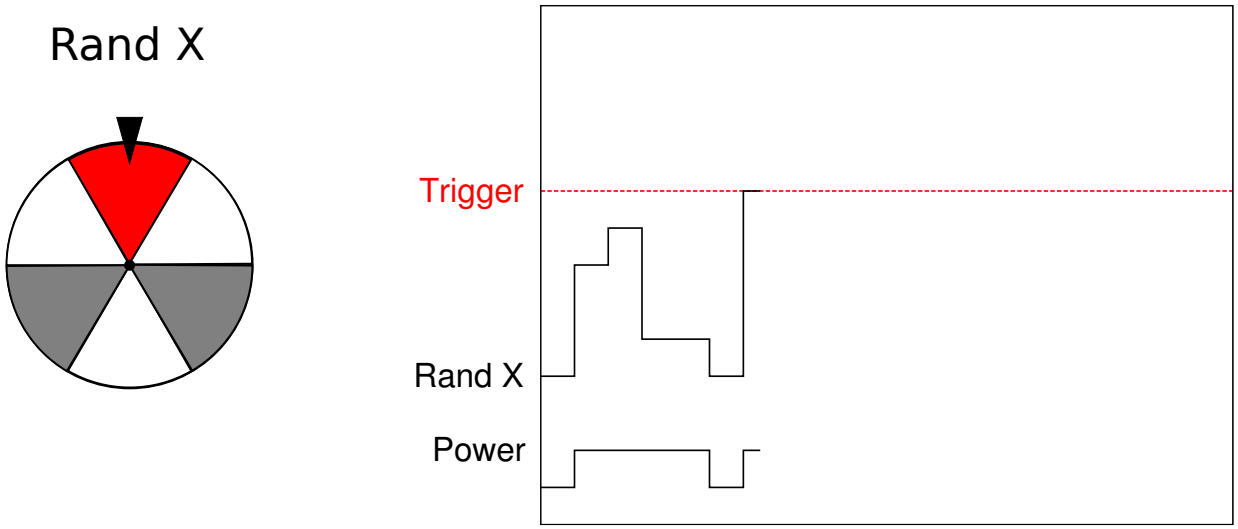
Single Shot Timer





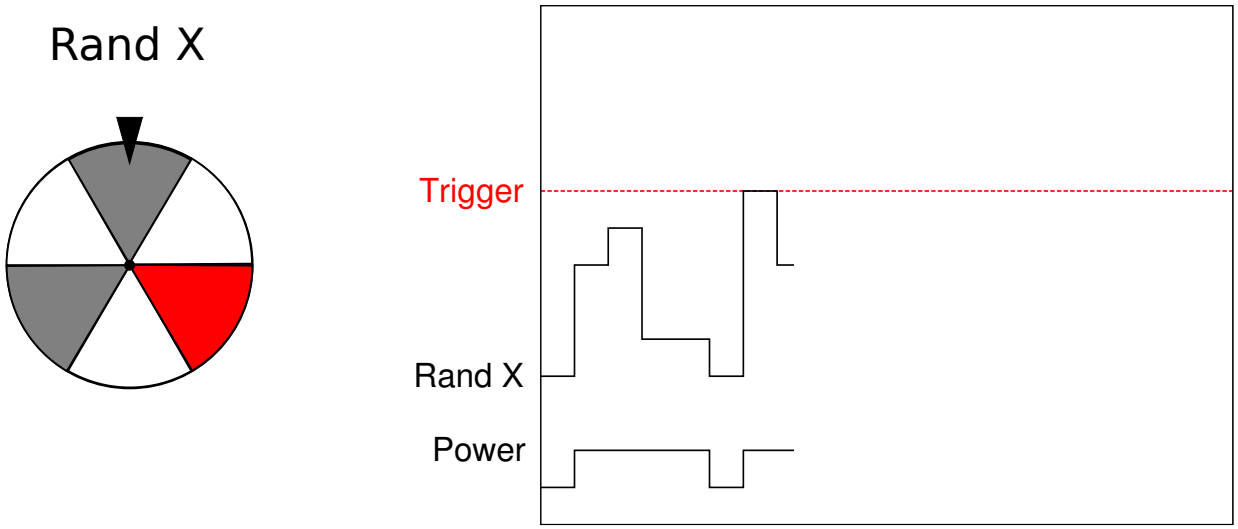
- Not easy to control
- Chance of early detection
- Large Variance

Single Shot Timer



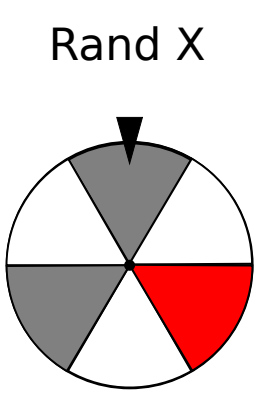
- Not easy to control
- Chance of early detection
- Large Variance

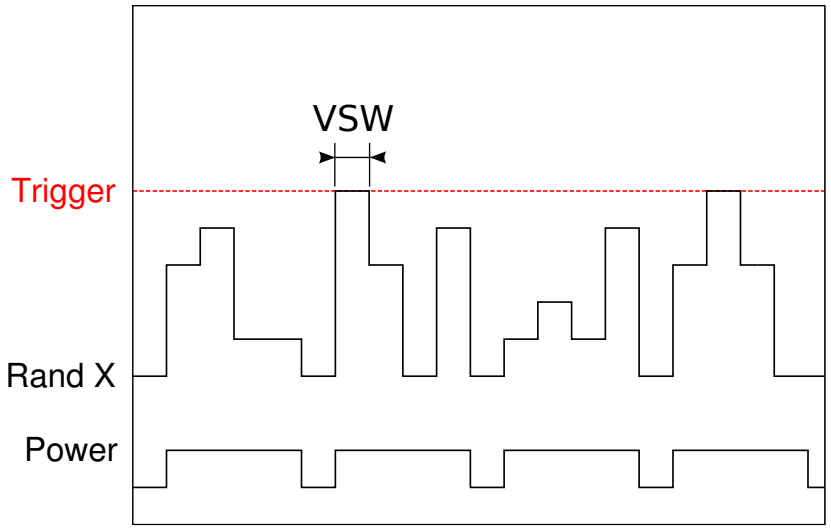
Single Shot Timer



- Not easy to control
- Chance of early detection
- Large Variance

Single Shot Timer

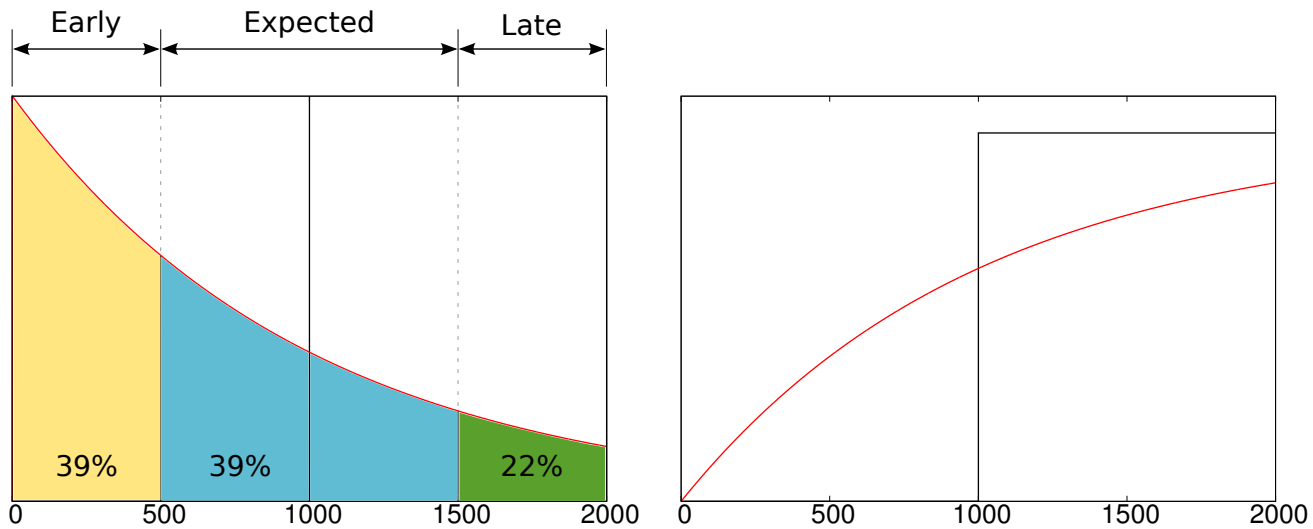




- Not easy to control
- Chance of early detection
- Large Variance
- Small VSW

Statistical Comparison

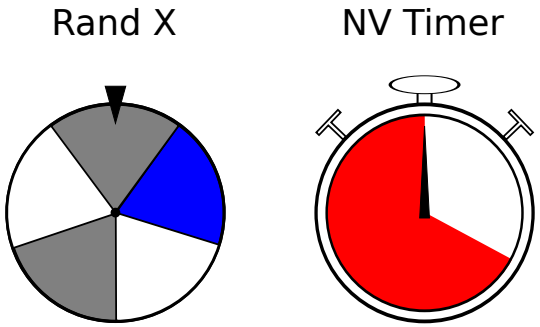
	Early	Expected	Late
D-Timer	0%	100%	0%
SS-Timer	39%	39%	22%

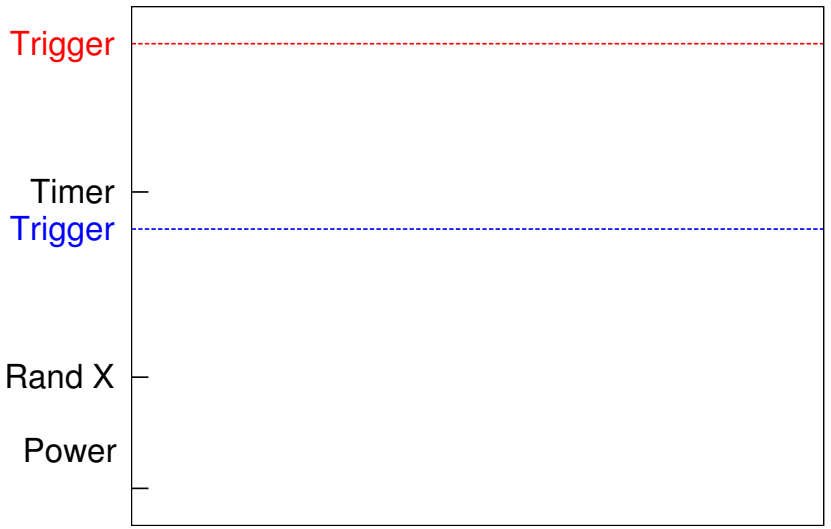


(a) PMFs (b) CDFs

Figure: The ND-Timer$_1$ was designed with an expectation of 1000. The corresponding deterministic timer is shown for comparison.

ND-Timer: Counts k Independent Random Events

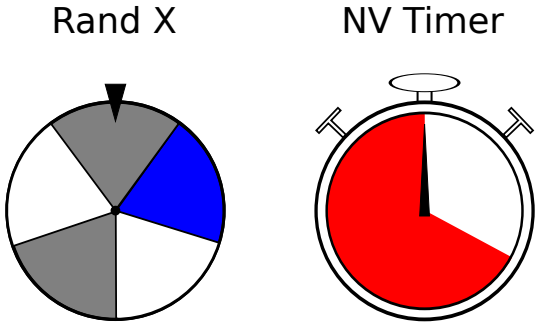


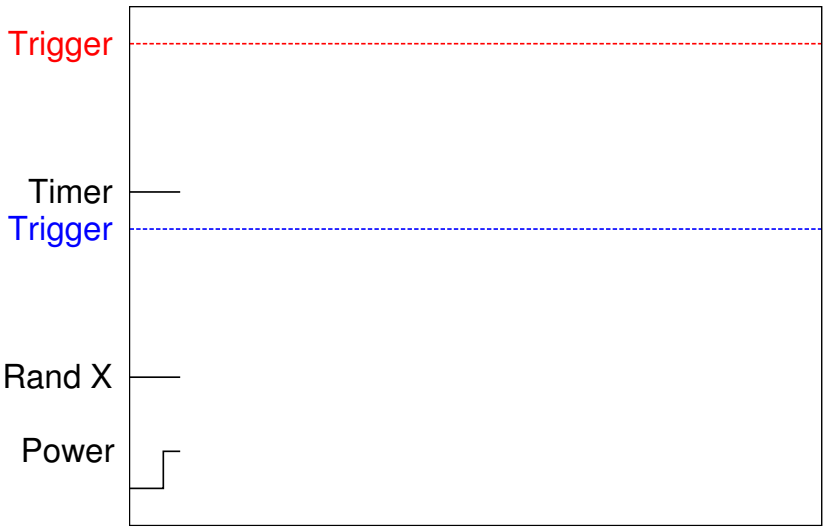


- Controllable
- Small Variance



ND-Timer: Counts k Independent Random Events

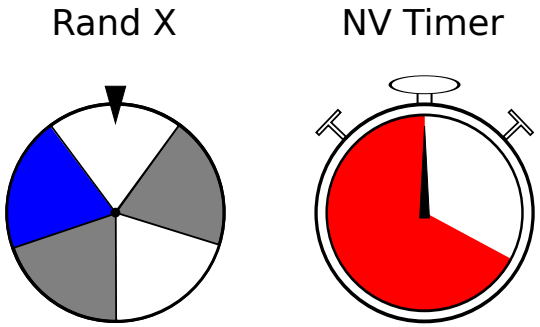


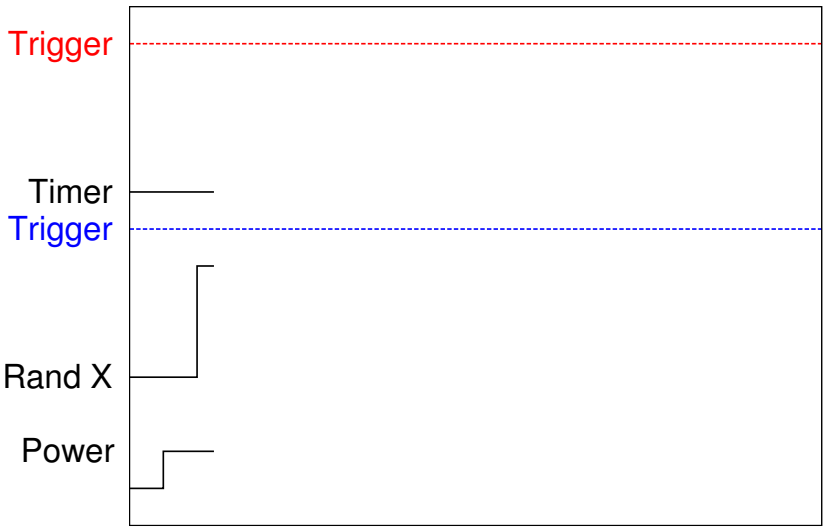


- Controllable
- Small Variance



ND-Timer: Counts k Independent Random Events

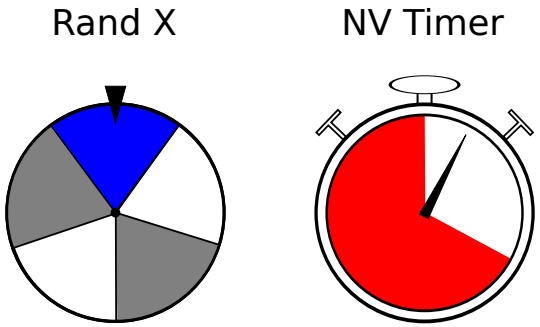


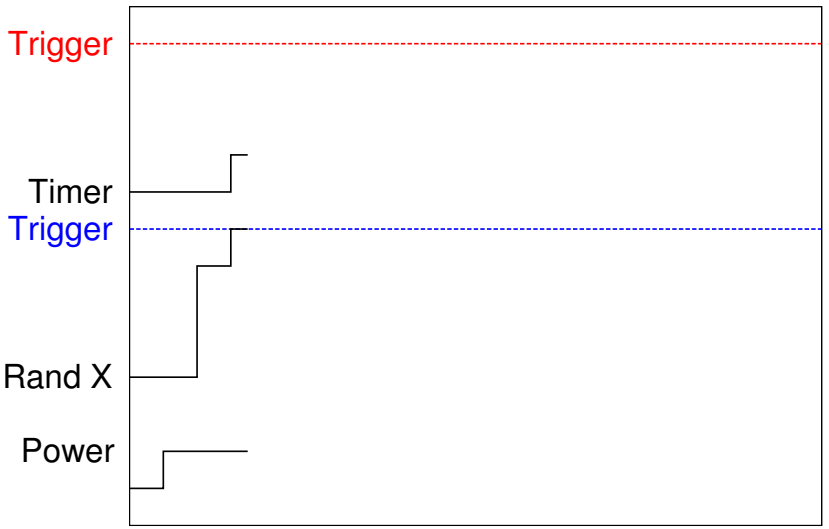


- Controllable
- Small Variance



ND-Timer: Counts k Independent Random Events

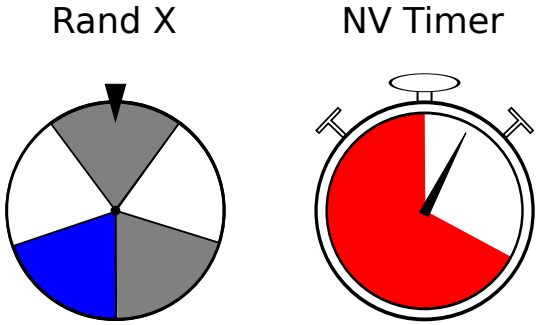


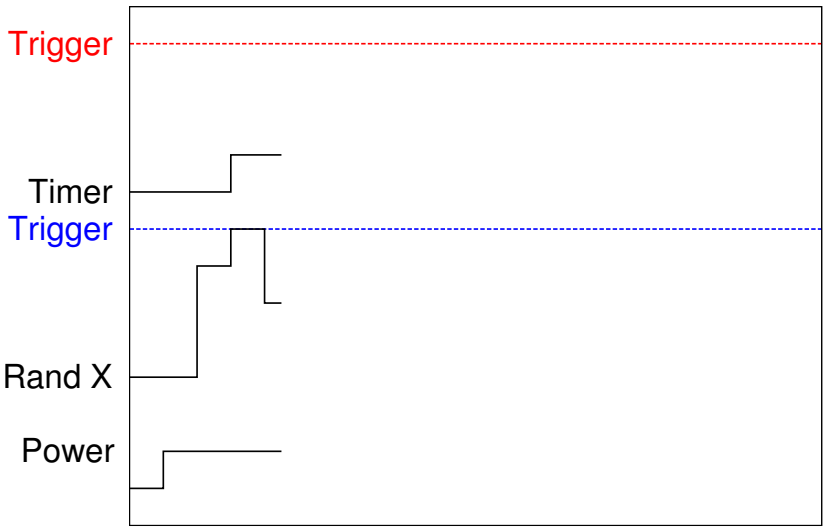


- Controllable
- Small Variance



ND-Timer: Counts k Independent Random Events

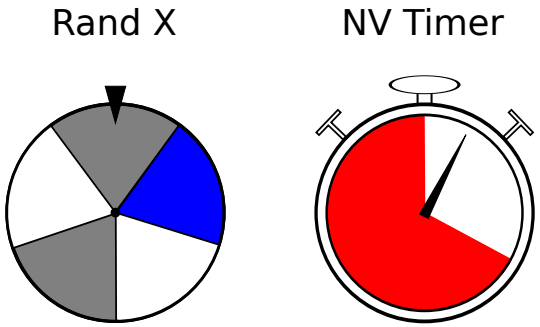


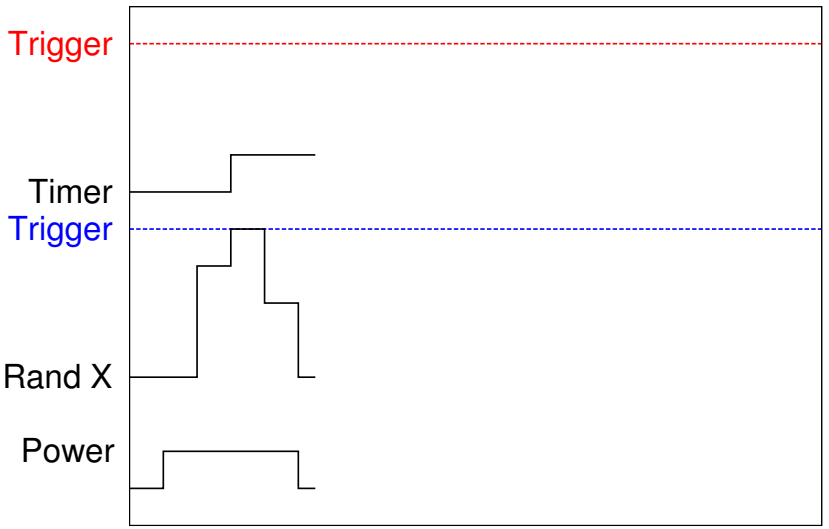


- Controllable
- Small Variance



ND-Timer: Counts k Independent Random Events

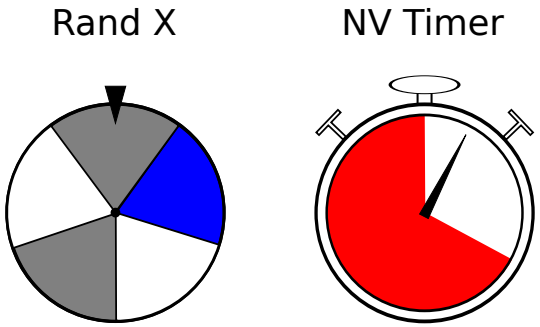


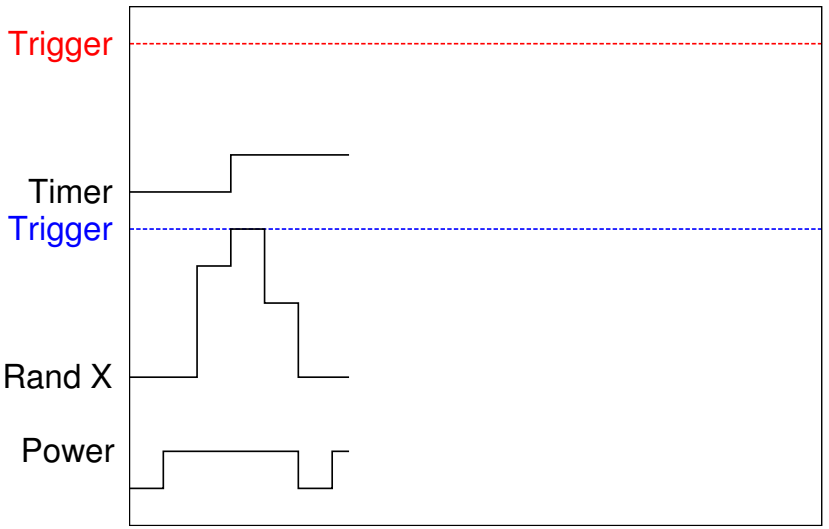


- Controllable
- Small Variance



ND-Timer: Counts k Independent Random Events

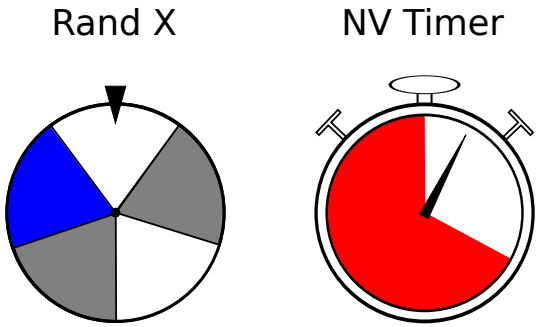


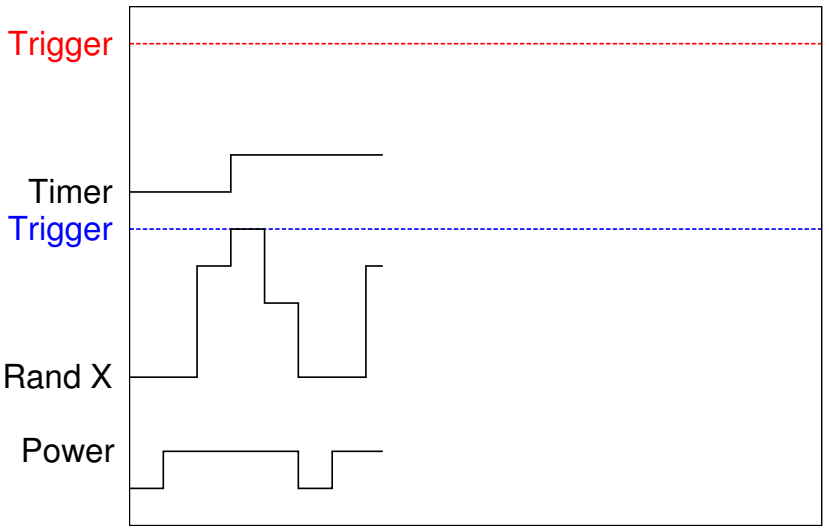


- Controllable
- Small Variance



ND-Timer: Counts k Independent Random Events

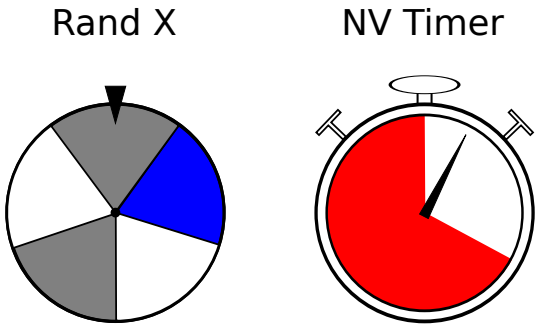


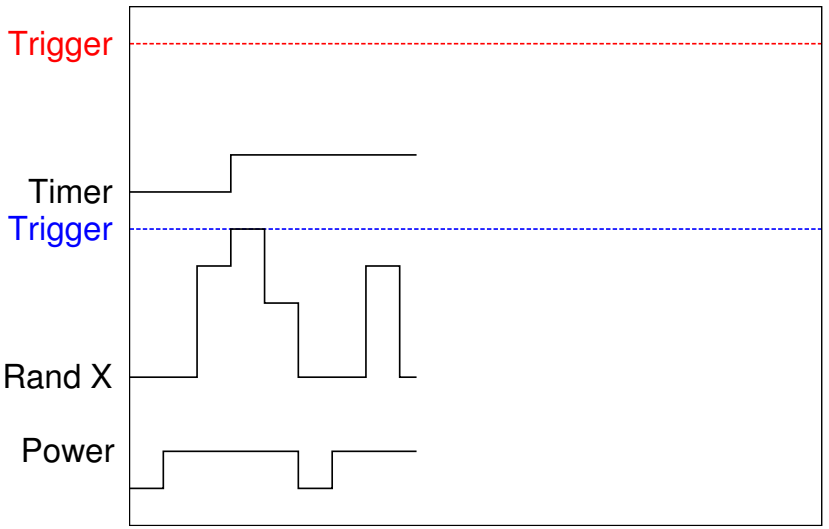


- Controllable
- Small Variance



ND-Timer: Counts k Independent Random Events

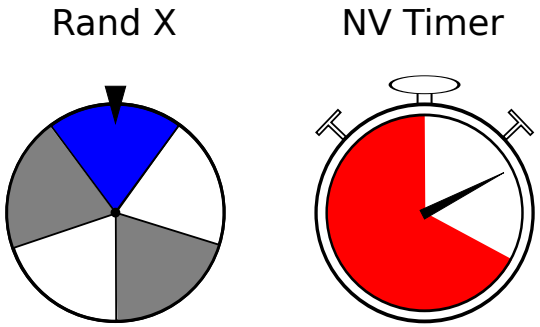


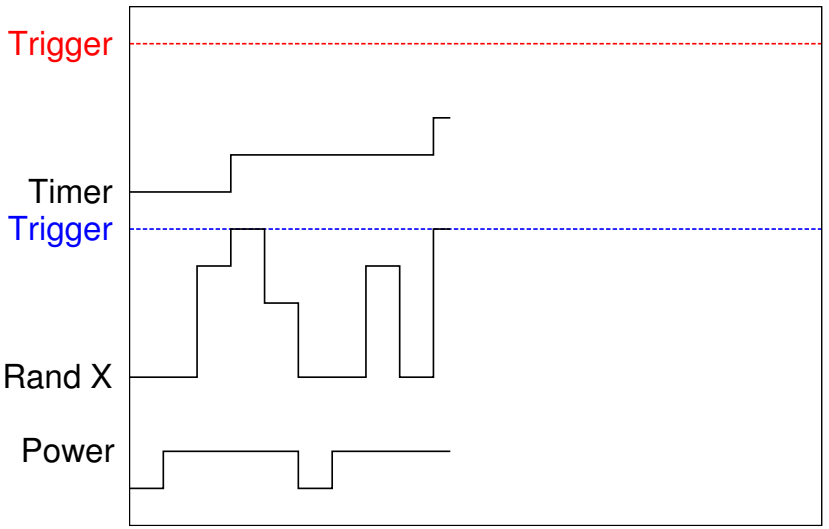


- Controllable
- Small Variance



ND-Timer: Counts k Independent Random Events

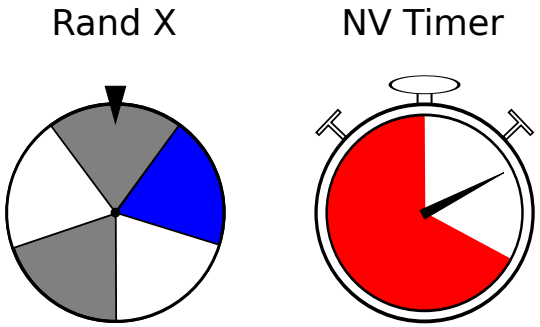


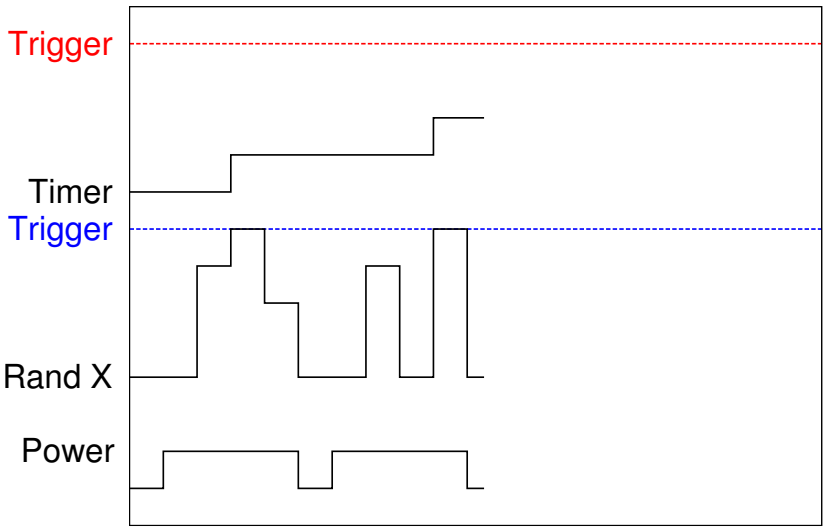


- Controllable
- Small Variance



ND-Timer: Counts k Independent Random Events

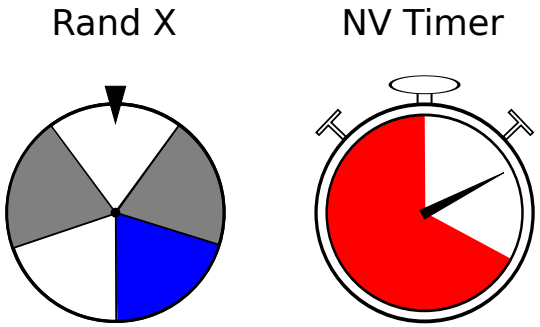


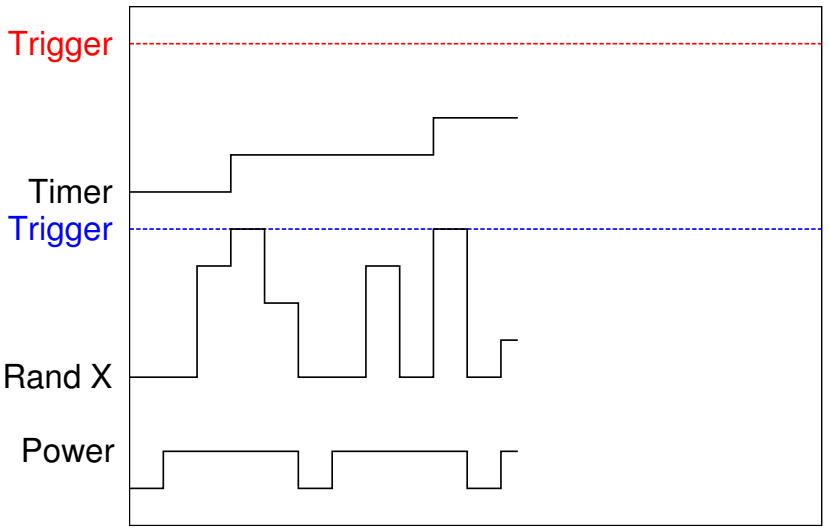


- Controllable
- Small Variance



ND-Timer: Counts k Independent Random Events

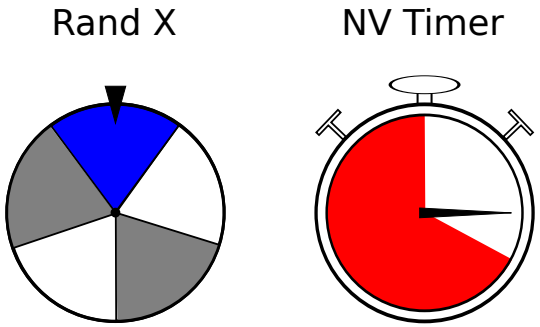


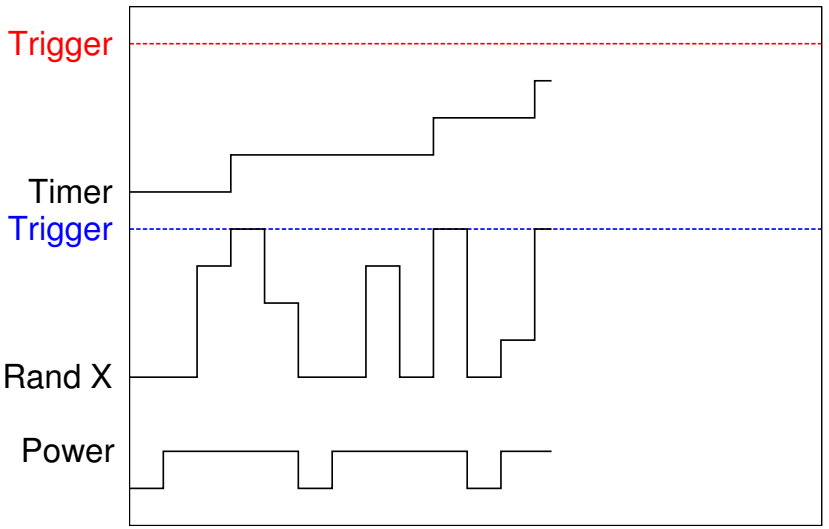


- Controllable
- Small Variance



ND-Timer: Counts k Independent Random Events

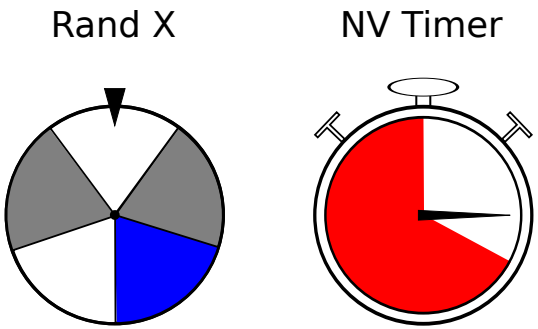


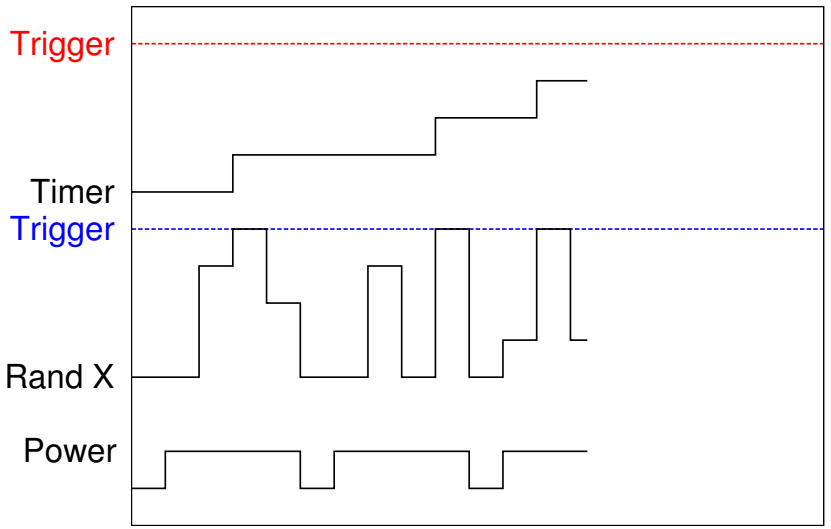


- Controllable
- Small Variance



ND-Timer: Counts k Independent Random Events

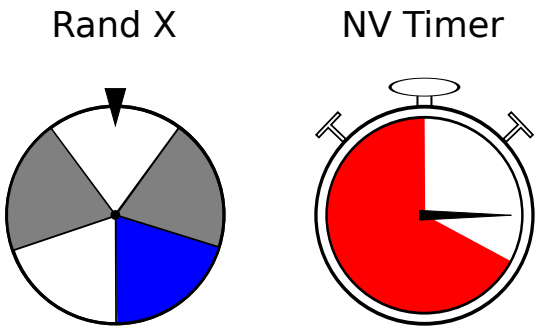


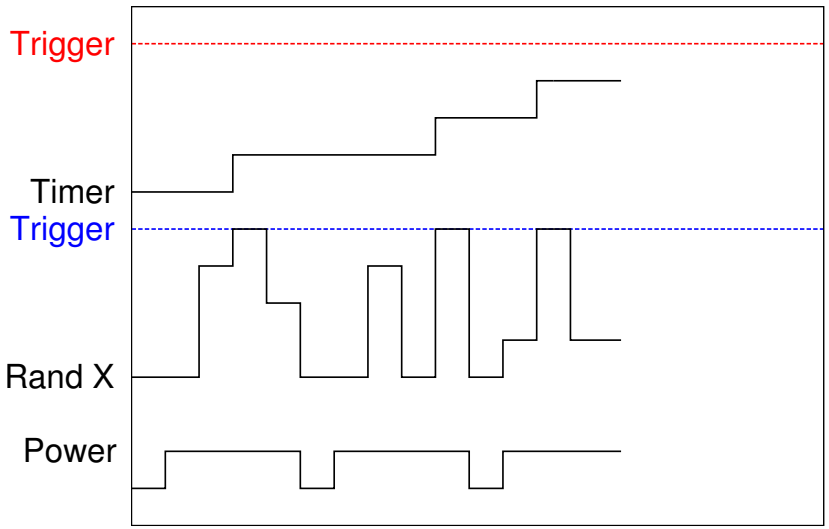


- Controllable
- Small Variance



ND-Timer: Counts k Independent Random Events

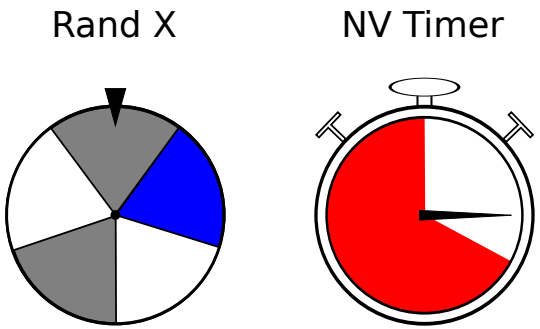


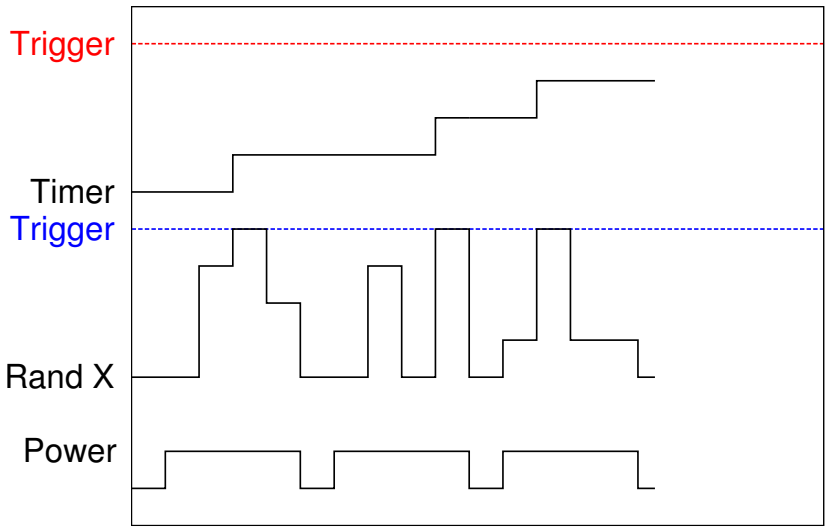


- Controllable
- Small Variance



ND-Timer: Counts k Independent Random Events

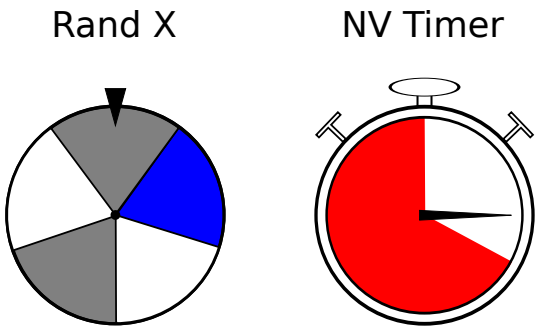


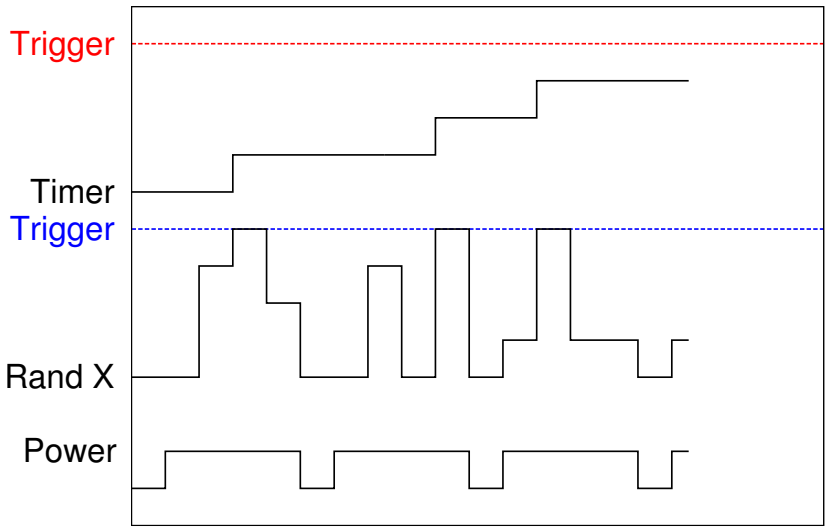


- Controllable
- Small Variance



ND-Timer: Counts k Independent Random Events

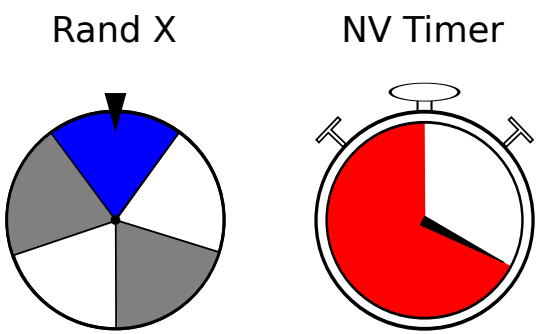


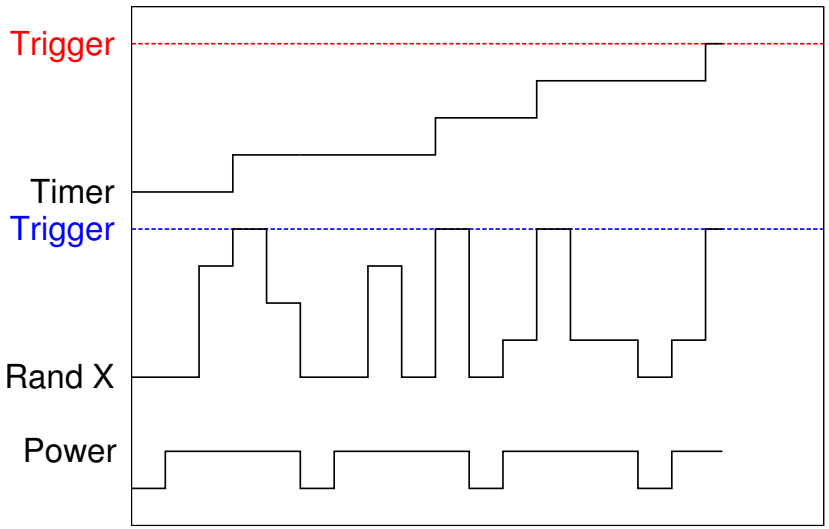


- Controllable
- Small Variance



ND-Timer: Counts k Independent Random Events

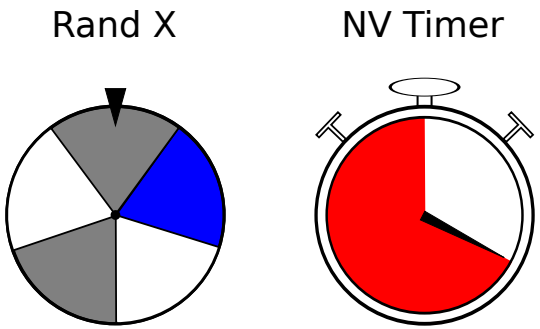


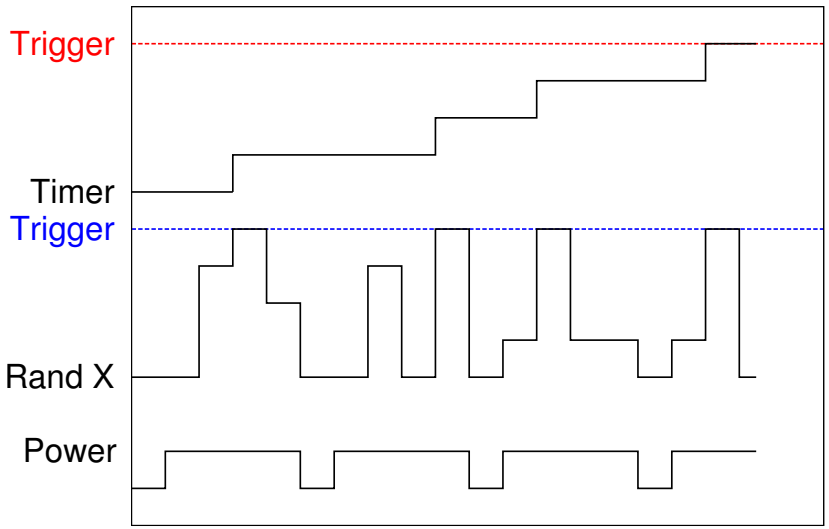


- Controllable
- Small Variance



ND-Timer: Counts k Independent Random Events

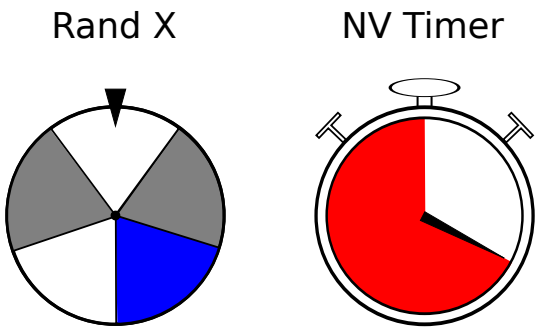


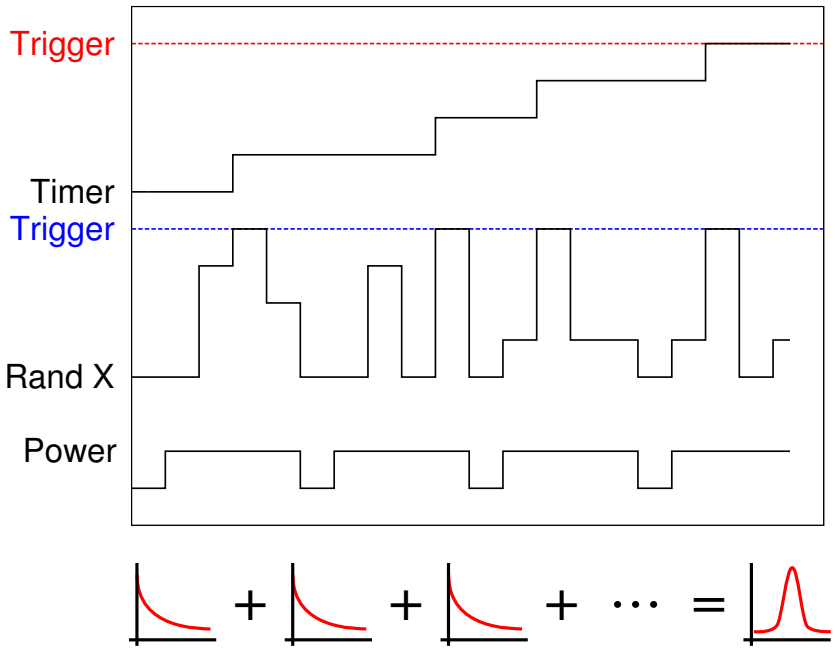


- Controllable
- Small Variance



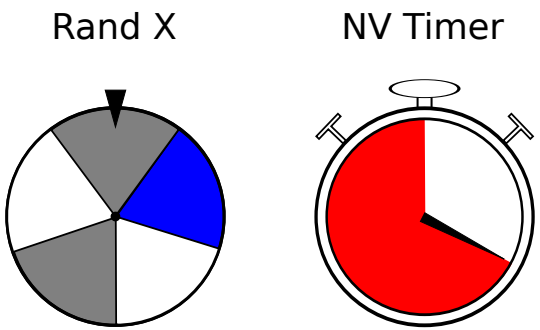
ND-Timer: Counts k Independent Random Events

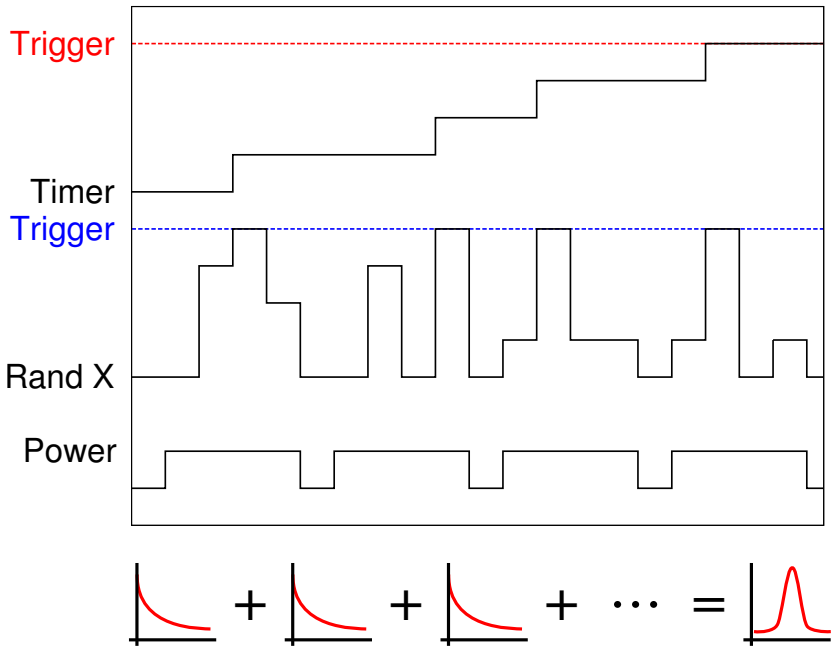




- Controllable
- Small Variance

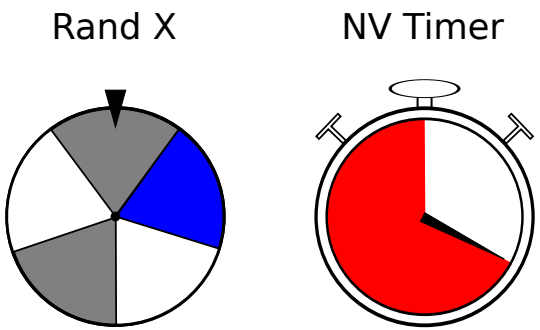
ND-Timer: Counts k Independent Random Events

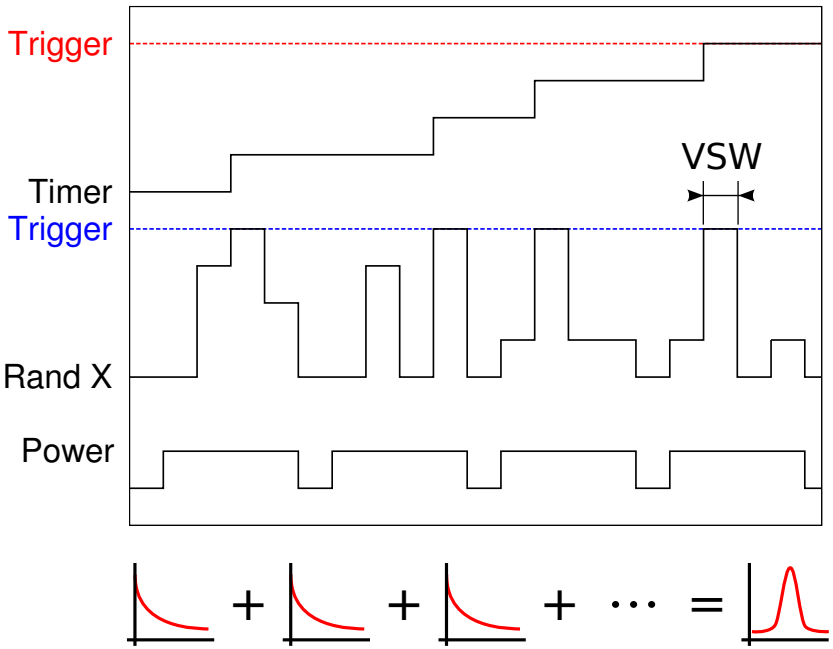




- Controllable
- Small Variance

ND-Timer: Counts k Independent Random Events

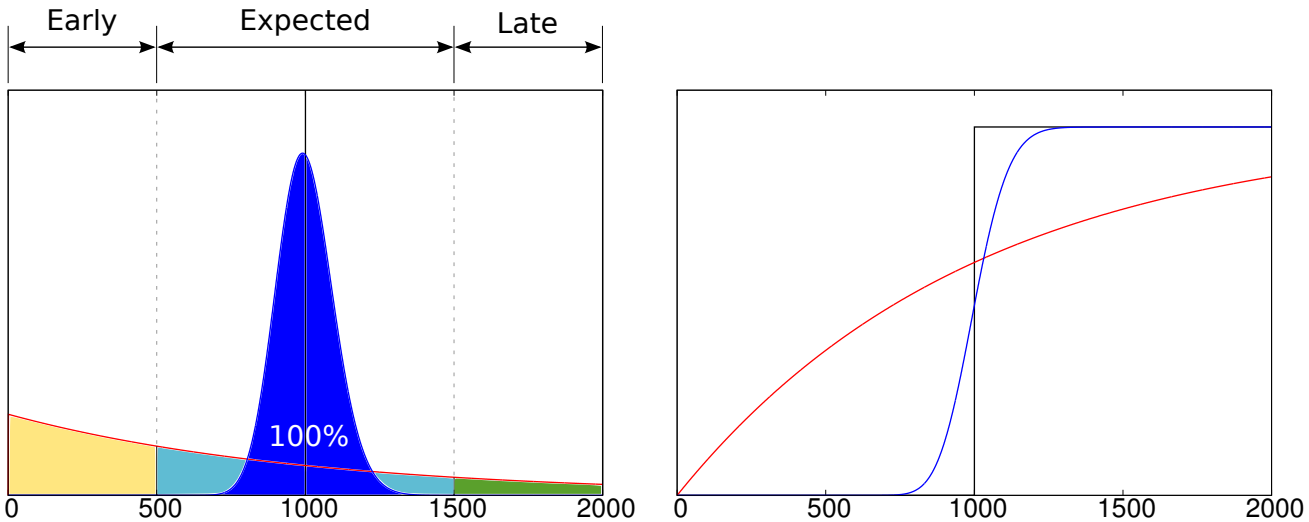




- Controllable
- Small Variance
- Small VSW

Comparison

	Early	Expected	Late
D-Timer	0%	100%	0%
SS-Timer	39%	39%	22%
ND-Timer	1.8e-9%	∼100%	2.22e-4%



(a) PMFs (b) CDFs

Figure: The ND-Timer was designed with an expectation of 1000. The corresponding deterministic timer and single event is shown for comparison.

Benefits of the ND-Timer

ND-timers break the trade-off between size of NV-memory and susceptibility to periodic power cycling.

- Accuracy $E(N) = \frac{k}{p}$
- Precision $\frac{\sigma}{E(N)} = \frac{1}{\sqrt{k}}$

	D-Timer01	D-Timer02	ND-Timer
Expected Trigger	1y	1y	1y
99% Confidence Interval	$\pm$0m	$\pm$0m	$\pm$12d
VSW	**52m**	$27.6\mu s$	$27.6\mu s$
NV-Bits	14	**114Mbits**	14

Table: Performance and resource comparison of each timers.

TRNG

1. Intro
   - Attack Model
   - Background
2. Deterministic Timers
   - Hardware
   - Single Stage D-Timer
   - Dual Stage D-Timer
3. ND-Timers
   - Single Shot Timer
   - ND-Timer
4. TRNG
   - Clock Jitter
   - Ring Oscillators
   - Sampling
5. Results
   - Hardware
   - 24 Hour Experiment
6. Detection
   - Hardware
   - Simulation
   - Analysis
7. Summary

TRNG

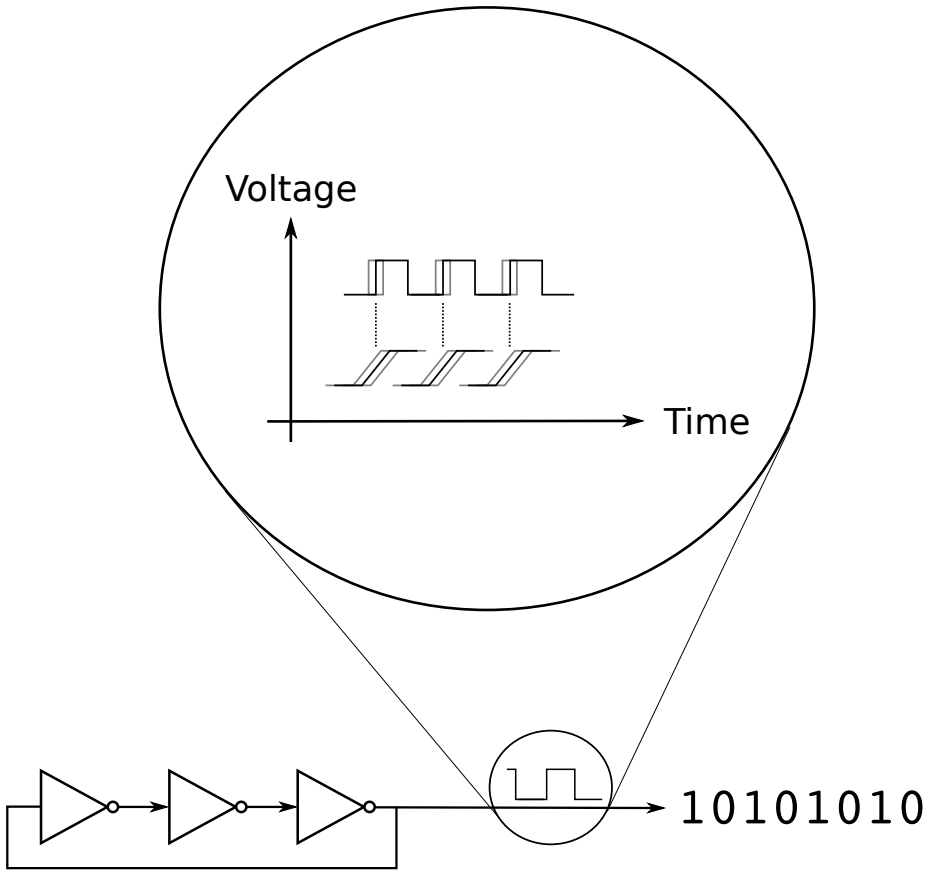
The generation of random numbers is too important to be left to chance.

Robert R. Coveyou

- True random sources exist in the physical world
  - Temperature
  - Timing
  - Power
  - ...

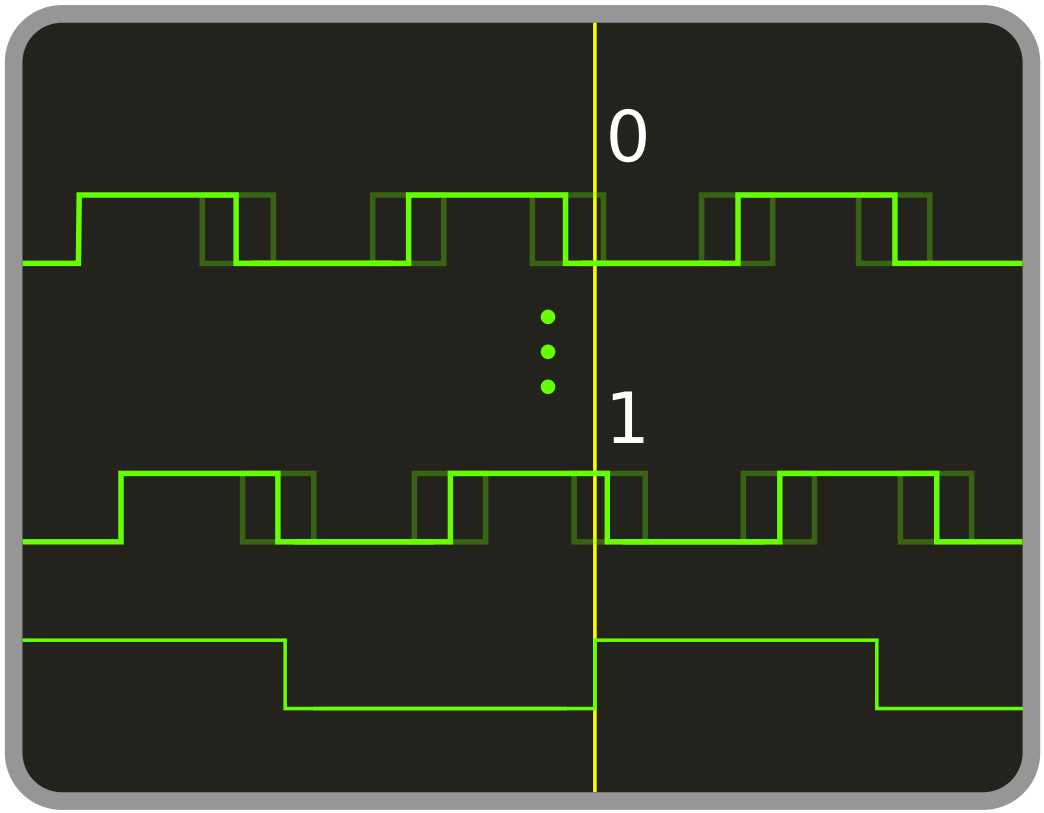
Clock Jitter

- Clock jitter contains true unbiased randomness [1]
- Easy to construct (minimum of 3 gates)



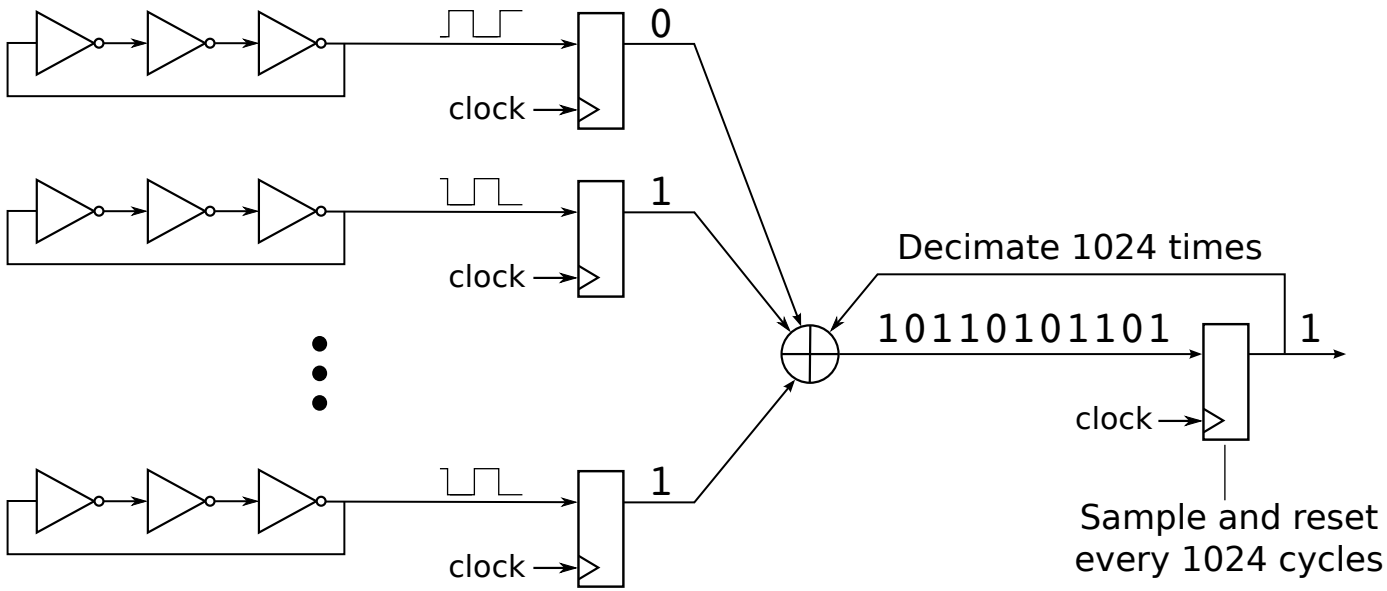
Sampling

Sample ROs with a slower clock to allow for multiple *separate* transitions.



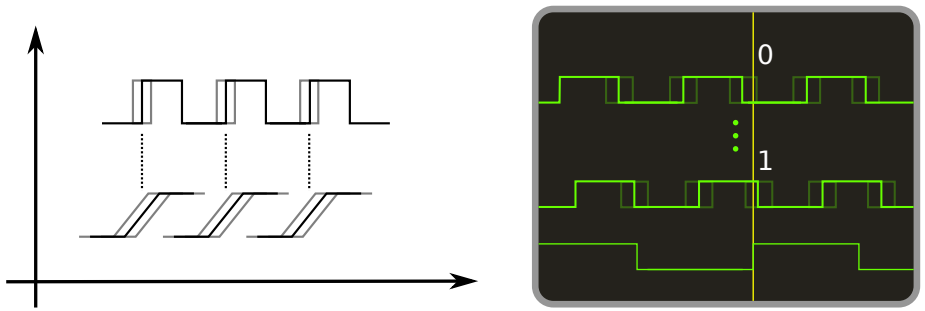
Ring Oscillators

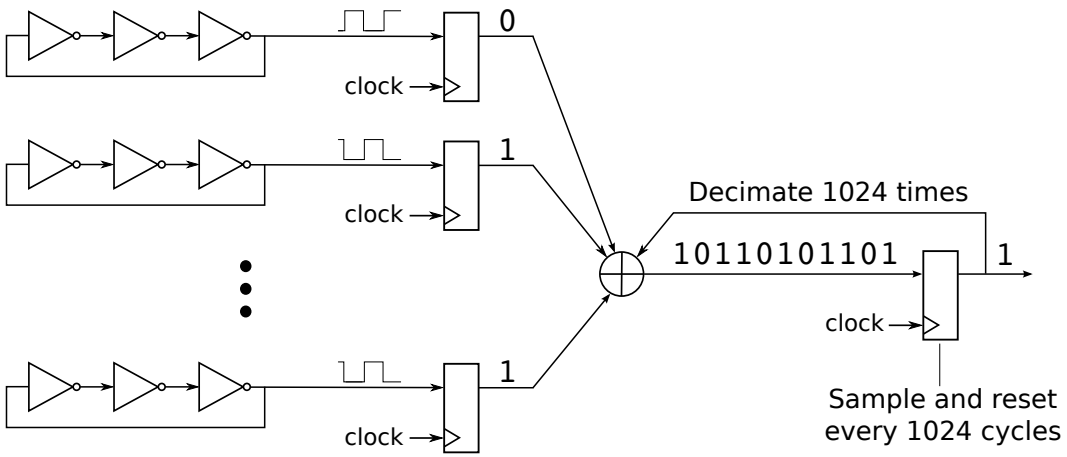
Use multiple ROs to fill the time spectrum with transitions. Down sample results 1024 times to remove bias and ensure *true* randomness. [8]



Summary

- 16 ROs
- 80 gates
- 1 bit/1024 cycles
- Passes NIST





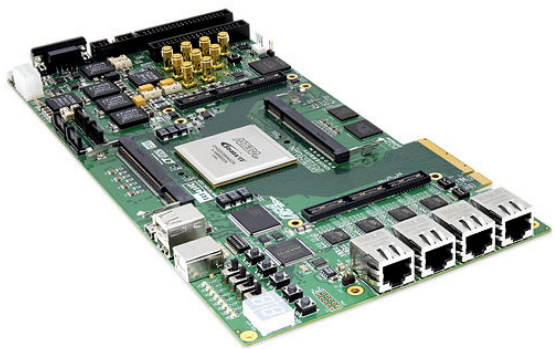
Results

1. Intro
   - Attack Model
   - Background
2. Deterministic Timers
   - Hardware
   - Single Stage D-Timer
   - Dual Stage D-Timer
3. ND-Timers
   - Single Shot Timer
   - ND-Timer
4. TRNG
   - Clock Jitter
   - Ring Oscillators
   - Sampling
5. Results
   - Hardware
   - 24 Hour Experiment
6. Detection
   - Hardware
   - Simulation
   - Analysis
7. Summary

Hardware

Experimental Hardware Setup:

- Altera DE4 development board
- Simulated power cycling and NV memory
- Two different 24h timer experiments
- Each experiment repeated 15 times
- Both experimental trials passed the χ^2 test for expected distributions.



24 Hour Experiment

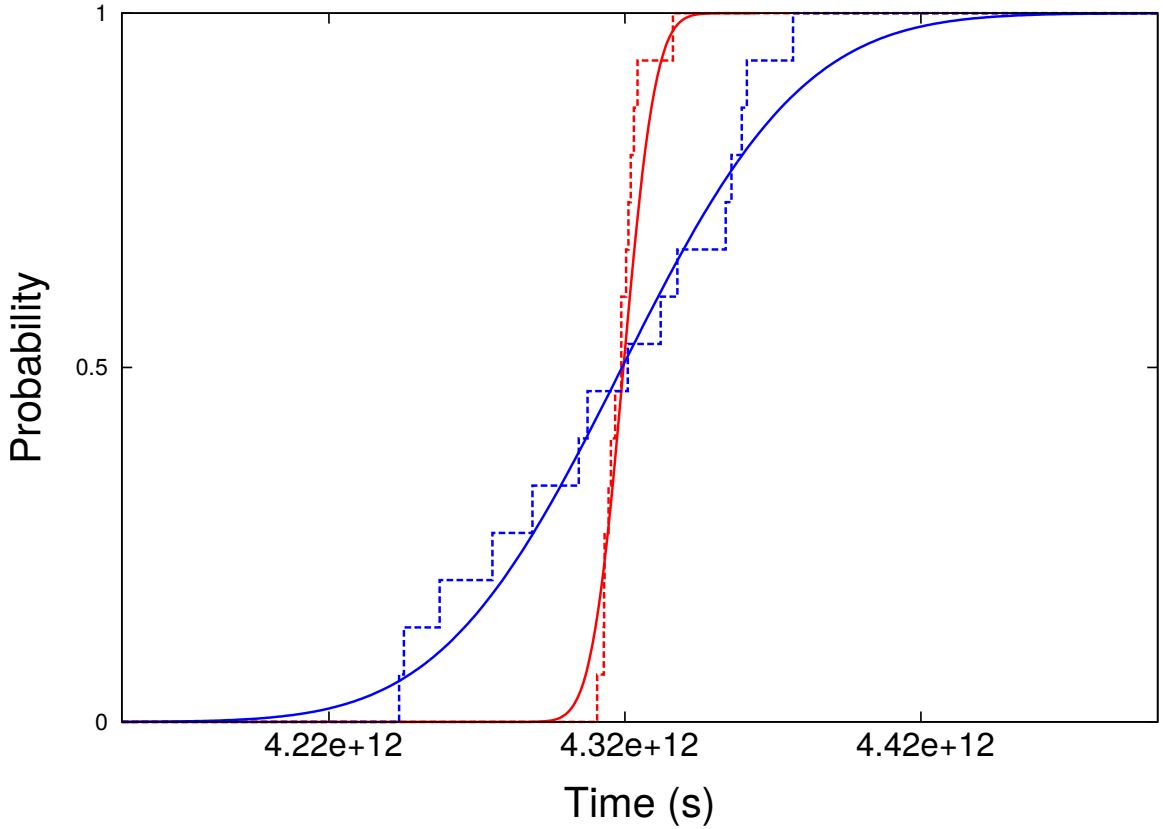


Figure: Measured and predicted CDFs for the two 24 hour triggers. Trigger one (red) has a standard deviation of 2.84 minutes, while trigger two (blue) the other has a standard deviation of 16 minutes.

Detection

1. Intro
   - Attack Model
   - Background
2. Deterministic Timers
   - Hardware
   - Single Stage D-Timer
   - Dual Stage D-Timer
3. ND-Timers
   - Single Shot Timer
   - ND-Timer
4. TRNG
   - Clock Jitter
   - Ring Oscillators
   - Sampling
5. Results
   - Hardware
   - 24 Hour Experiment
6. Detection
   - Hardware
   - Simulation
   - Analysis
7. Summary

Hardware Resource Identification

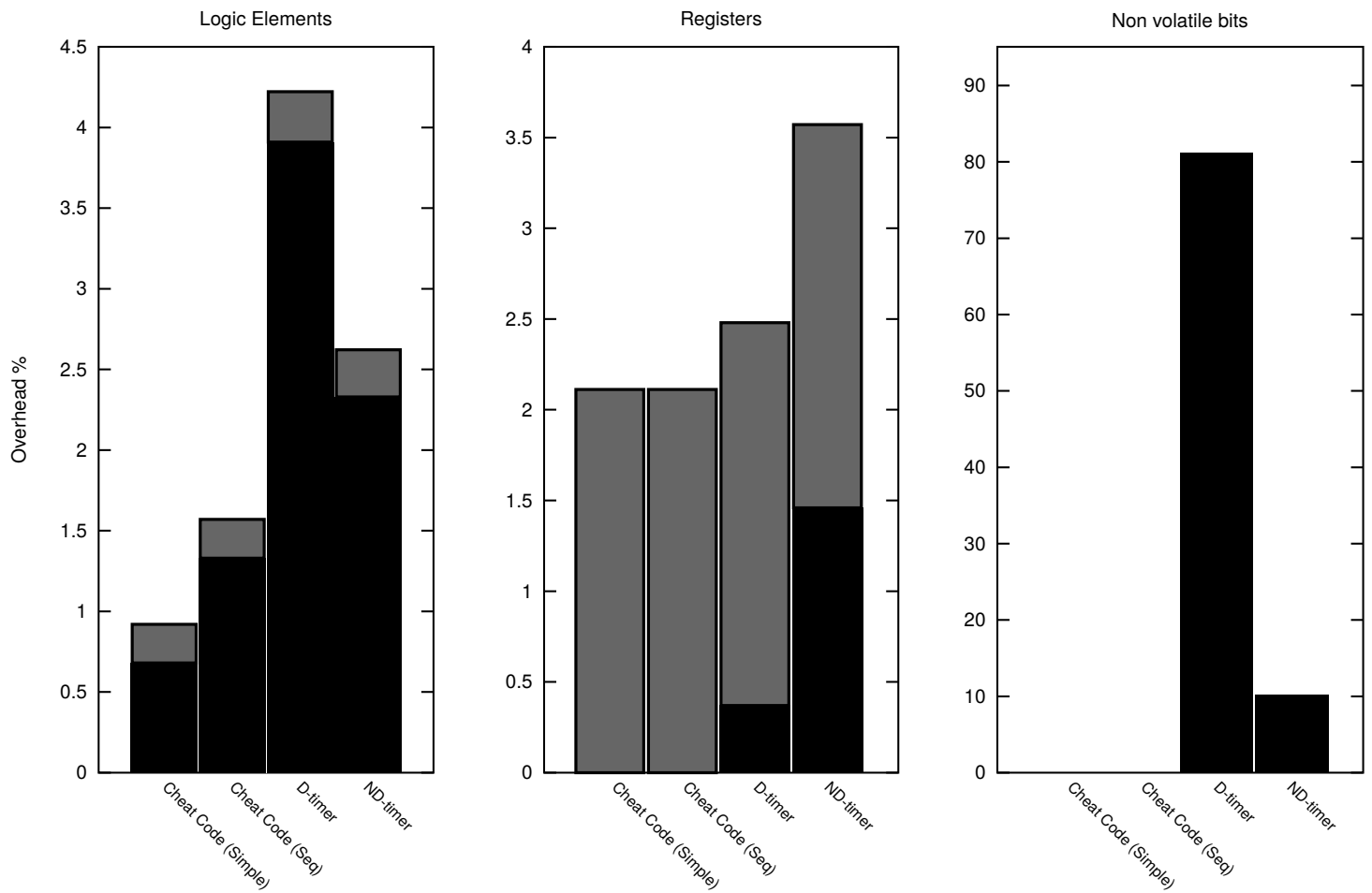


Figure: Hardware overhead comparison for two data trigger and two timer attacks. Black represents the amount used by the trigger and grey represents the amount used by the Trojan.

Simulation: Does not have randomness!

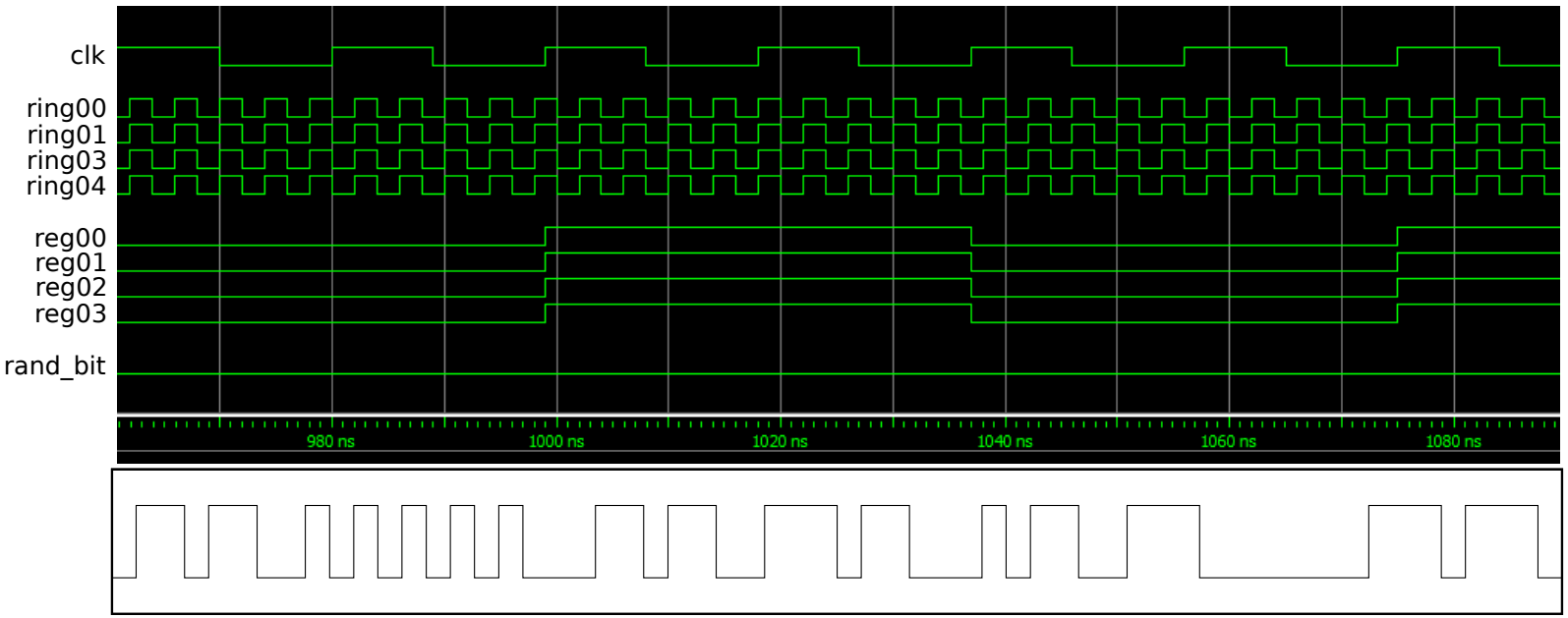
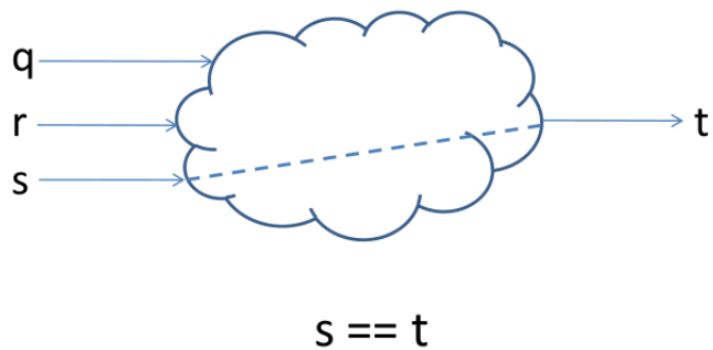


Figure: A simulation results from Modelsim shown in the top diagram for ND-timer. Bottom diagram shows a typical pattern for `rand_bit` gathered from FPGA.

Logic Analysis: Unused Circuit Identification

UCI & FANCI:

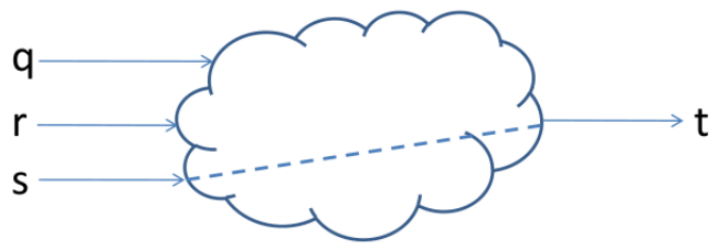
- Looks for unused or nearly unused logic (circuitry).
- Redundant circuitry could be replaced with a wire.
- Flagged as suspicious.



Logic Analysis: Unused Circuit Identification

UCI & FANCI:

- Looks for unused or nearly unused logic (circuitry).
- Redundant circuitry could be replaced with a wire.
- Flagged as suspicious.
- Known methods to defeat UCI [6] and FANCI [9].

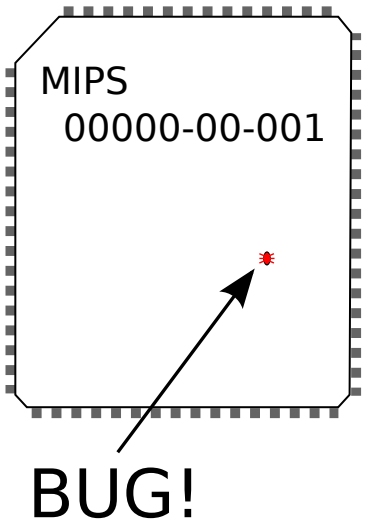


s == t

Summary

ND-Timers:

- Defeat power cycling defences.
- Use less resources than D-Timers.
- Controllable.
- Do not simulate.



References I

[1] B.J. Abcunas.
Evaluation of random number generators on FPGAs.
PhD thesis, WORCESTER POLYTECHNIC INSTITUTE, 2004.

[2] Alex Baumgarten, Michael Steffen, Matthew Clausman, and Joseph Zambreno.
A case study in hardware trojan design and implementation.
International Journal of Information Security, 10(1):1–14, 2011.

[3] Yier Jin, Nathan Kupp, and Yiorgos Makris.
Experiences in hardware trojan design and implementation.
In *Hardware-Oriented Security and Trust, 2009. HOST'09. IEEE International Workshop on*, pages 50–57. IEEE, 2009.

[4] Justin Rilling, David Graziano, Jamin Hitchcock, Tim Meyer, Xinying Wang, Phillip Jones, and Joseph Zambreno.
Circumventing a ring oscillator approach to fpga-based hardware trojan detection.
In *Computer Design (ICCD), 2011 IEEE 29th International Conference on*, pages 289–292. IEEE, 2011.

[5] S. Skorobogatov and C. Woods.
Breakthrough silicon scanning discovers backdoor in military chip.
Cryptographic Hardware and Embedded Systems–CHES 2012, pages 23–40, 2012.

References II

[6] C. Sturton, M. Hicks, D. Wagner, and S.T. King.
Defeating uci: Building stealthy and malicious hardware.
In *IEEE Symposium on Security and Privacy*, pages 64–77, 2011.

[7] A. Waksman and S. Sethumadhavan.
Silencing hardware backdoors.
In *IEEE Symposium on Security and Privacy*, pages 49–63, 2011.

[8] K. Wold and C.H. Tan.
Analysis and enhancement of random number generator in fpga based on oscillator rings.
International Journal of Reconfigurable Computing, 2009:4, 2009.

[9] Jie Zhang, Feng Yuan, and Qiang Xu.
Detrust: Defeating hardware trust verification with stealthy implicitly-triggered hardware trojans.
In *Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security*, pages 153–166. ACM, 2014.