Redstone circuits/Miscellaneous

re article feedback

< Redstone circuits

This article covers types of circuits with insufficient content to justify their own articles.

Contents

ABBA circuit

An **ABBA circuit** has one input and multiple outputs. When the input turns on, the outputs turn on in order (e.g., A then B), but when the input turns off, the outputs turn off in reverse order (e.g., B then A).

The timing of the output activation and deactivation can be changed by adjusting the delay on the repeaters.

All of these designs can be extended indefinitely.

Earliest Known Publication: 5 July 2011 (basic concept),[1] 20 June 2012 (Floor ABBA),[2] and 18 July 2012 (Ceiling ABBA).[3]

Schematic Gallery: ABBA Circuit

Block update detector

See also: Tutorial:Block update detector

The **block update detector** switch, or simply BUD, detects any time an adjacent block receives an update. An update is anything that changes that block's state: block placed, destroyed, door opened, repeater delay changed, cake eaten, grass growing, snow falling, furnace used (or turns off), and so on (chests opened and crafting tables used do not cause updates, sleeping in a bed does). When a block is updated, all adjacent blocks are also eligible for an update, and this can be used to produce a redstone signal or for other purposes.

BUDs have been used for all kinds of things, from traps to detecting daylight to locking mechanisms on hidden doors.

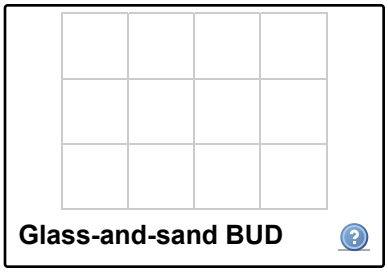
Current BUD switches take advantage of a quirk where pistons can receive power, but not updates, from blocks adjacent to the space the piston head occupies when extended. That is: blocks adjacent to the extended state and not adjacent to the retracted state *provide power* to extend the piston, but as they are not adjacent to the retracted state do not cause the piston to update when that power state changes.

There are many models of auto resetting BUDs. This is done by having the firing of the BUD switch somehow cause the BUD switch to restore the pre-fired state often when fired it cuts the power to the piston, allowing it to retract. This is done either by extra circuitry or, as in the example below, by the properties of blocks themselves.

Version 1.11 introduced the observer block, which sends a one-redstone-tick signal when the block it is facing is updated. However, many designs using BUDs still incorporate older piston variations.

Glass-and-sand BUD

1×3×4 (12 block volume)
1-wide


Glass-and-sand BUD

The sticky piston should be activated by quasi-connectivity, but it doesn't know that yet. When a block next to the piston is updated, the piston "notices" it should activate, and extends. However, that pushes a block of glass into the sand's place, so the piston is no longer activated by quasi-connectivity. Accordingly, the piston retracts, resetting the circuit. This circuit outputs an off-pulse; invert the output for a regular on-pulse.

Earliest Known Publication: 28 December 2011[4]

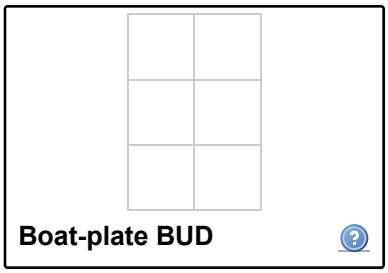
Boat-plate BUD

1×2×4 (8 block volume)

When placed on the sand, the boat overlaps the pressure plate, powering the block below, which should activate the piston by quasi-connectivity. When the piston is updated, it

pushes the sand and boat up, unpowering the plate and block, deactivating the piston again. By the time the boat lands on the pressure plate, the piston has stopped checking for activations, resetting the circuit.

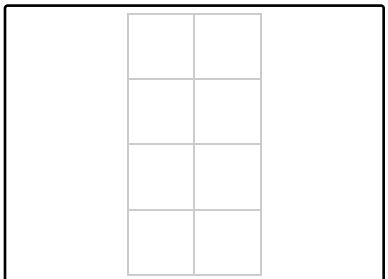
Boats are easily moved by passing mobs (including players) — use this BUD only in areas where mob movement is absent.


Boat-plate BUD

Slime block BUD

1×1×4 (4 block volume)
1-wide

The piston should be activated by quasi-connectivity, but doesn't react until updated. It takes 1 tick to give an output but 0 ticks to reset itself, so its a powerful and easy BUD.


Slime Block BUD

Multiplex circuit

A **multiplex circuit** transmits multiple signals on a single transmission line by using a *control line* to choose which signal should be transmitted or received. The control line may be either *clocked*, to transmit the signals sequentially, or *selection*, to simply choose which signal to transmit. The control line(s) can also be substituted with two *synchronized* pulse generating clocks. Then only one line is necessary. However, they have to stay synchronized for the transmission to be correct. This can be done using a daylight detector.

A multiplex circuit may consist of a *multiplexer*, a *demultiplexer*, or both.

Multiplexer

A **multiplexer** (a.k.a. "mux") is a device that selects one of two or more inputs and outputs the selected input. This multiplexer can be chained together, allowing for multiple bit multiplexing.

For these designs, the inputs are labeled A and B. C is the "control" input: whether A or B passes through depends on whether C is powered or not.

AND-Gate Mux

3×5×3 (45 block volume), silent
circuit delay: 2 ticks

When C is off, the A input can pass through but the B input is suppressed. When C is on, the B input can pass through but the A input is suppressed.

Horizontal Dust-Cut Mux

3×3×3 (27 block volume)
circuit delay: 0 ticks

While the block is moving, both A and B can pass through at the same time -- in practice, this just means the transition takes 1.5 ticks when going from a high signal to a low, and instantly for all other cases.

Piston Mux

4×4×2 (32 block volume), flat
circuit delay: 1 tick

Vertical Piston Mux

1×4×6 (24 block volume), 1-wide
circuit delay: 1 tick

Schematic Gallery: Multiplexer

Demultiplexer

AKA "demultiplexer" or "DEMUX".

A 2-bit demultiplexer (A)

Hopper–dropper multiplexer (silent; hopper contains 1 64-stackable item)

Another 2-bit demultiplexer (B)

The demultiplexer is the reverse of the multiplexer. It allows you to have one input be sent to either of one of two or more different outputs. Here (https://upload.wikimedia.org/wikipedia/commons/thumb/1/15/Demultiplexer_Example01.svg/450px-Demultiplexer_Example01.svg.png) is a schematic of a demultiplexer with 4 outputs. 'I' represents the input, 'S' represents the selectors and 'f' the outputs. The two selector wires, holding a two digit binary number (2 wires = 2 digits), are passed to a decoder. If the input wire is on, then the output wire of the demultiplexer corresponding to the output of the decoder turns on. For example, if the input is on and the binary number 10b is passed onto the select wires (S0=on,S1=off. Start counting at zero!), then the decoder turns on the third wire (10 in binary is two, but because we start count at zero it becomes the third). An AND operation is performed between the input wire of the demultiplexer and each of the outputs of the decoder. As stated earlier, because both the input wire and only the 3rd output of the decoder is on, only the third output of the whole demultiplexer is on. Demultiplexers are used as

decoders with the option to have ALL outputs turned off (done by turning off the input wire). The two schematics on this page are those of a 2 bit demultiplexer. A demultiplexer is not a relay. A relay is an old electronic component that was replaced by transistors in the 70s. Even though a 1-pole relay is the same as a 2 bit demultiplexer, a more complex demultiplexer would be made of many relays (and in the current century of transistors). The closest thing in Minecraft resembling a relay or transistor is the Redstone torch.

Random number generator

See also: Tutorial:Randomizers

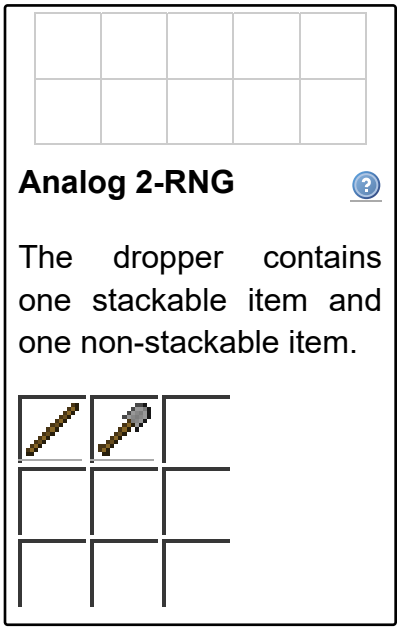
A **random number generator** (aka RNG, or "randomizer") is a circuit that can generate numbers with no recognizable pattern.

The RNGs below depend on the stack maximums of the items they push:

- A **64-stackable item** is any item with a stack maximum of 64 items (for example, sticks, cobblestone, etc.).
- A **16-stackable item** is any item with a stack maximum of 16 items (for example, signs, snowballs, ender pearls, etc.).
- A **non-stackable item** is any item that cannot be stacked (for example, tools, weapons, armor, boats, etc.).

Analog 2-RNG

1×3×2 (6 block volume), 1-wide, flat, silent
circuit delay: 3 ticks (rising) and 1 tick (falling)



Outputs either power level 1 or 3 while on, power level 0 while off.

When the input turns on, the dropper randomly chooses to push either the stackable item or the non-stackable item into the hopper, causing the comparator to output either power level 1 or 3. Because the powered dropper is a solid/opaque block, it also deactivates the hopper, preventing it from pushing the item back to the dropper until the input turns off.

The output power level can be used as is (for example, to subtract 1 or 3 from a comparator in subtraction mode), but more often the output is connected to a line of two redstone dust so that the output is 0 or not 0 (to randomly power a repeater, activate a mechanism component, etc.).

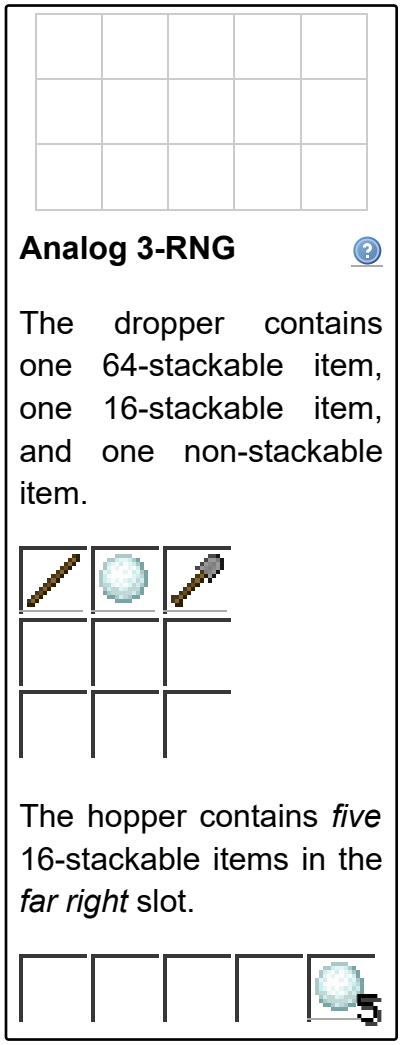
Variations: If the dropper is powered indirectly (for example, by quasiconnecitvity or an adjacent powered block), the hopper is not be deactivated and immediately pushes the item back into the dropper. This turns the circuit into a monostable rising edge detector with a 3.5-tick output pulse (still with a random power level of 1 or 3).

With only two items in the dropper, both output power levels are chosen with equal probability. The probability of the output levels can be changed by adding additional stackable and non-stackable items to the dropper (which must all be different from each other so they don't stack). For example, with two different stackable items and three different non-stackable items, the RNG outputs power level 1 40% of the time and power level 3 60% of the time.

Earliest Known Publication: March 14, 2013[5]

Analog 3-RNG

1×3×3 (9 block volume), 1-wide, silent
circuit delay: 3 ticks (rising) and 1 tick (falling)



Outputs power levels 1, 2, or 4 while on, power level 1 while off (but see variations below).

When building this circuit, wait until the hopper is deactivated by the powered dust before putting five 16-stackable items in its far right slot. Then put a 64-stackable item, a 16-stackable item, and a non-stackable item in the dropper.

Before the input turns on, the hopper's five 16-stackable items are enough to produce a power level 1 output from its comparator (even a single 64-stackable item would be enough for that). These five items should never be returned to the dropper, so the comparator's output never drops below power level 1.

When the input turns on, the dropper pushes an item into the hopper, which is placed in the hopper's left slot. It takes 23 64-stackable items (or five 16-stackable items and three 64-stackable items, or six 16-stackable items) to produce power level 2, so if the 64-stackable item is pushed that isn't sufficient to increase the output power level, but if the 16-stackable item is pushed the output power level increases to 2. And if the non-stackable item is pushed, the output power level increases to 4.

The hopper is held deactivated by the powered dust when the circuit is off, and by the powered dropper when the circuit is on. But, when the input turns off, there is a brief 1-tick moment when the dropper has just turned off, but the torch attached to it hasn't turned on again. This allows the hopper to activate for 1 tick, pushing an item back into the dropper. A hopper always pushes items from its left slots first, so the hopper pushes back the item the dropper pushed into it, rather than any of the 16-stackable items in its far right slot, allowing the circuit to reset itself.

Variations: The player can remove one of the items from the dropper to create a 2-RNG with different power level outputs than the regular 2-RNG: removing the 64-stackable item outputs power levels 2 or 4, removing the 16-stackable item outputs power levels 1 or 4,

and removing the non-stackable item outputs power levels 1 or 2.

You can add additional redstone dust leading from the hopper to a block next to it, and then down to the side of the comparator. This 2-wide variation keeps the comparator's output off while the input is off.

With only three items in the dropper, all three output power levels are chosen with equal probability. The probability of the output levels can be changed by adding additional 64-stackable, 16-stackable, and non-stackable items to the dropper (which must all be different from each other so they don't stack). For example, with one 64-stackable item, one 16-stackable item, and two different non-stackable items, the RNG outputs power level 1 25% of the time, power level 2 25% of the time, and power level 4 50% of the time.

Additional items can be added to the hopper to increase all of the output power levels.

Earliest Known Publication: 16 April 2013[6]

Analog 16-RNG

5×8×4 (160 block volume)
circuit delay: 8.5 ticks

Outputs power levels 0 to 15 while on, power level 0 while off.

Uses four 2-RNGs to subtract 1, 2, 4, and/or 8 from 15.

Reducing the number of 2-RNGs reduces the possible outputs: three 2-RNGs produces an 8-RNG, and two 2-RNGs produces a 4-RNG (the exact power levels depend on the power level provided to the subtraction comparators).

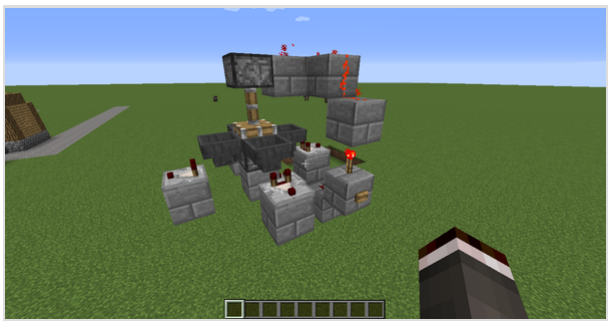
Earliest Known Publication: June 10, 2013[7]

Schematic: Analog 16-RNG

Random selector

Main article: Tutorial:Randomizers

A **random selector** is a circuit that selects an available output at random.



Random Selector

The random selector (right) depends on the random flight of an item from a dropper.

The input (button) in this random selector turns off the torch, deactivating the piston. It also powers a redstone dust directed into the block, powering the repeater and activating the dropper. The piston then reactivates to push the stuck item (if any). A hopper collects the item and puts it back into the dropper, activating a comparator.

NOTE: The dropper must be facing up. If facing down, the item glitches out of the machine.

Earliest Known Publication: 24 September 2014[8]

An old kind of random selector is the "chicken randomizer", which uses a chicken walking on multiple pressure plates in a containment chamber. For this and other improved randomizers, view the randomizers tutorial.

Another way to construct a random selector is to feed the output of a digital randomizer into a demuxer. You can create a 1-bit digital randomizer from an analog 2-RNG by discarding one of the outputs, and then combine N of them for an N-bit digital randomizer.

Thyristor

A thyristor is a bit like a D flip-flop but has a different functionality. It has two inputs: the "gate" line G and the "anode" line A, and has one output: the "cathode" line K. When triggered by G, the circuit sets its output K to A **if A is high**, then holds that output state **until A goes low**. Otherwise K stays low.

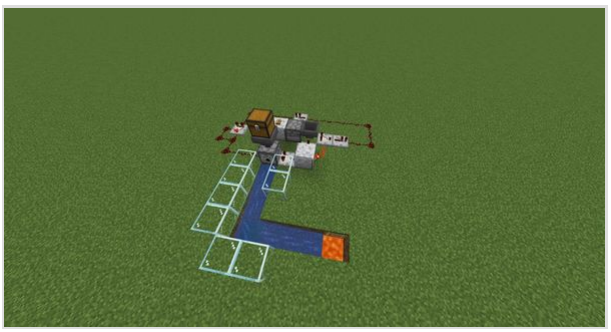
Design **A** is the simplest form of the circuit. The hopper is initially empty so K stays low regardless of whether A is high or low. When G goes high, the item in the dropper moves to the hopper so K goes high as well. The hopper is suspended by A so as long as A is high, the item doesn't go back to the dropper so K is kept high. *But there is a downside to this design. When triggered by G while A is low, a 1 tick on-pulse is emitted from K.*

Design **B** is an improved form at the cost of complexity / size / delay. There is an AND gate between G and A so the gate input gets completely discarded when A is low.

Earliest Known Publication: 28 March 2016[9]

Schematic Gallery: Thyristor

Use case



This is an automatic item disposer with a start button. Put items in the chest, then push the button behind it. All the items in the chest are thrown into lava via clocked dropper below the chest. While design **B** is used in this screenshot, design **A** is fine too because the bogus 1-tick output would activate only the empty disposer.

Hysteresis circuit

Hysteresis circuit a.k.a. Schmitt trigger is a circuit that activates the output if input signal power reaches the 'high' threshold, switches it off if power drops to 'low' threshold, and sustains previous state if power remains between the two. Typical use is activation of disposal from storage overflows (which tends to be noisy, so this allows keeping the run-time short), activation of automatic farms to re-fill storage if it's running low, and various storage systems where startup and shutdown are comparably heavy-weight relative to continued operation itself.

0 – 15 hysteresis circuit.

0 – n+1 hysteresis circuit.

Container has enough items for the comparator to produce power level *n*.

n-1 – n+1 hysteresis circuit.

m – n+1 hysteresis circuit.

Earliest Known Publication: May 20, 2017[10]

References

1. "VioletTheGeek" (5 July 2011). "Minecraft Redstone Reverse-order deactivation sequencer" (https://www.youtube.com/watch?v=JJH97AoUnio) (Video). *YouTube*.
2. "VioletTheGeek" (20 June 2012). "Minecraft Redstone Vertical Reverse-Order Deactivation Circuit" (https://www.youtube.com/watch?v=8p2v0C_X4DQ) (Video). *YouTube*.
3. "NiceMarkMC" (18 July 2012). "Minecraft - Tutorial: 1 Wide ABBA Switch" (https://www.youtube.com/watch?v=s1JvtPwn1Kg) (Video). *YouTube*.
4. "Tatarize" (28 December 2011). "Compact Minecraft BUD switch. 1x3x4. Auto-reset. Minecraft 1.0.0" (https://youtube.com/watch?v=xD-phtgkMK4) (Video). *YouTube*.
5. "HiFolksImAdam" (14 March 2013). "Minecraft 1.5 Tutorial: Simple Random Number Generator!" (https://www.youtube.com/watch?v=zJ_K4O6rAnY) (Video) – *YouTube*.
6. "Kwin van der Veen" (16 April 2013). "Video response: Sharing is Caring #010: 1 Bit Randomizer & Silent BUD " (https://www.youtube.com/watch?v=kL4rPRrkgFI) (Video). *YouTube*.
7. "Entity" (June 10, 2013). "Calling All Redstone Geniuses, I need Help. - advanced randomizer." (https://www.minecraftforum.net/topic/1839130-calling-all-redstone-geniuses-i-need-help-advanced-randomizer/page__st__20#entry22812780) – *Minecraft Forum*

8. "MaNgAkA_fReAk" (24 September 2014). "Random Selector for 1.8" (https://www.minecraftforum.net/forums/minecraft-discussion/redstone-discussion-and/redstone-creations/2218970-random-selector). *Minecraft Forum*.
9. depressed_pho (28 March 2016). "Does anyone know the name of this circuit?" (https://www.minecraftforum.net/forums/minecraft-discussion/redstone-discussion-and/2644718-does-anyone-know-the-name-of-this-circuit). *Minecraft Forum*.
10. "R.K.F. Walter" (May 20, 2017). "Tutorial: Simple, adjustable Hysteresis Circuits (Schmitt Triggers)" (https://www.youtube.com/watch?v=yx7jwrn2h0Q) (Video). *YouTube*.

Navigation

Redstone [hide]

Redstone circuits & tutorials [hide]

Redstone basics	Redstone components · Redstone mechanics · Conductivity
Redstone circuits	Clock circuits · Logic circuits · Memory circuits · Miscellaneous circuits · Pulse circuits · Transmission circuits
Featured tutorials	Advanced redstone circuits · Block update detector (BUD) · Comparator update detector (CUD) · Flying machine · Mechanisms · Piston circuits · Quasi-connectivity · Redstone music

Redstone components [show]