



Keep Coding and Nobody Explodes

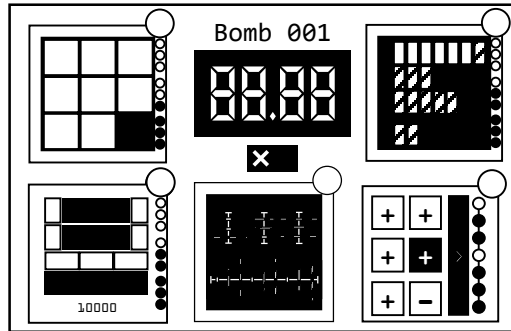
This isn't what you expected from Bomb Lab...

Bomb Defusal Manual

Bomb can be accessed at: bit.ly/cspuzzle9

Defusing Bombs

A bomb will explode when its countdown timer reaches 0:00 or when too many strikes have been recorded. The only way to defuse a bomb is to disarm all of its modules before its countdown timer expires. For best results, we recommend using Chrome.



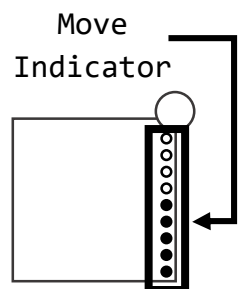
Example Bomb

Modules

Each bomb will include 5 modules that must be disarmed. Each module is independent and can be disarmed in any order. Instructions for disarming modules can be found in the following pages.

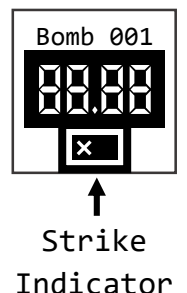
Moves

Most modules require you to solve them with the fewest possible moves. The number of moves needed will be displayed in lit LEDs on the right side of the module. Every move you make will turn off one LED, and if the module is not solved when all LEDs are off, you will incur a strike.



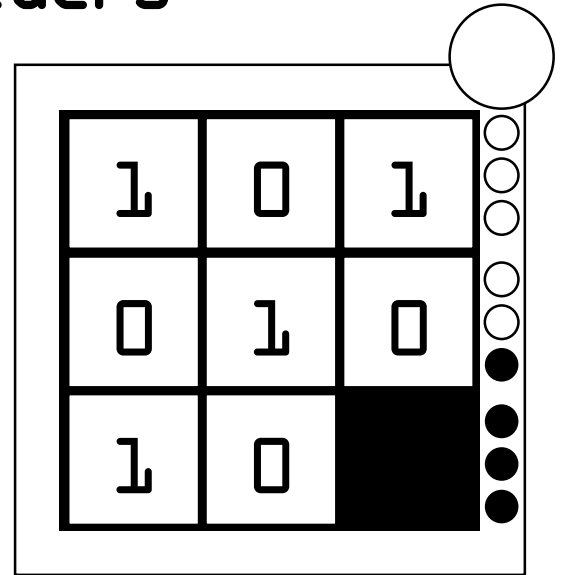
Strikes

When you make an invalid move or run out of moves the bomb will record a strike which will be displayed on the indicator above the countdown timer. Bombs will explode upon the next error after the strike indicator is filled.



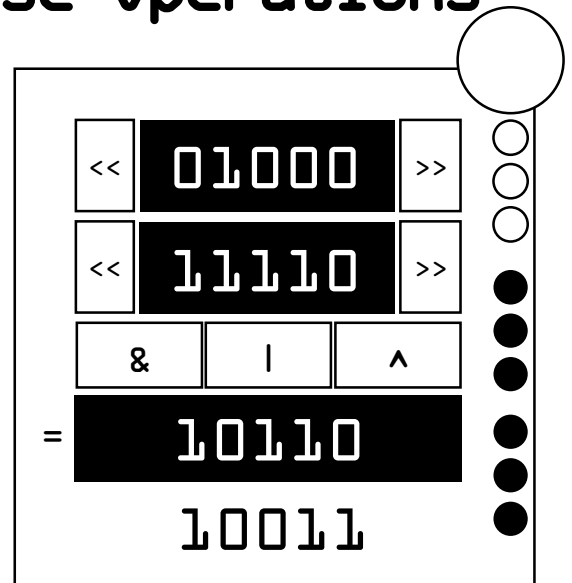
On the Subject of Sliders

- Bit tiles can only be moved in cardinal directions.
- Click on a tile adjacent to the open space to slide it into that space.
- To complete the module, all four corners must contain tiles with matching bits.



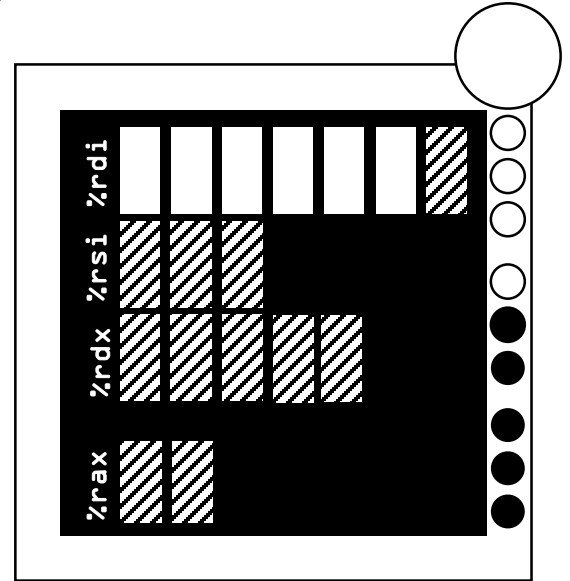
On the Subject of Bitwise Operations

- Both upper windows hold a five digit binary number.
- Click on a button next to a number to perform a single bit-shift operation on that number (padded with 0's in both directions).
- Click on a button below both numbers to perform the corresponding binary operation, storing the result in the third window. This can only be done once, does not count as a move, and will immediately incur a strike if the result is not correct.
- To complete the module, the third window must contain a number matching the one printed at the bottom of the module.



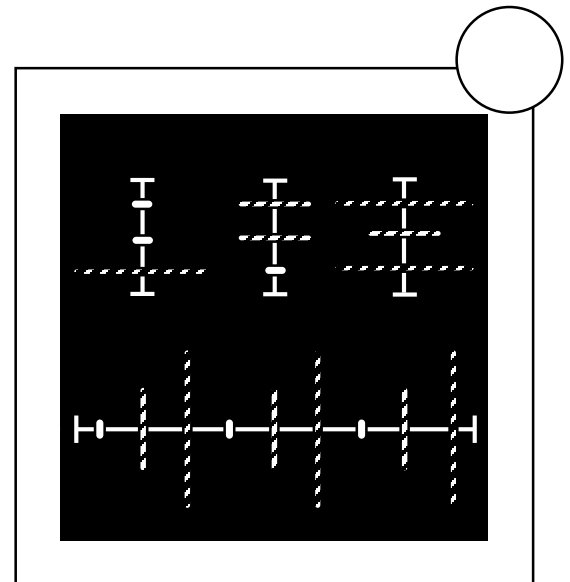
On the Subject of Registers

- Each register can hold up to a certain number of bits, indicated by the number of rectangles in that row.
- Lit rectangles are considered “filled.”
- To move bits between registers, click first on the source, then the destination. The register will move as many bits as possible from the source to the destination.
- Bits can only be moved to %rax if their source has exactly as many bits as the capacity of %rax.
- To complete the module, fill %rax.



On the Subject of Concurrency

- Three threads are displayed at the top, their contents ordered from top to bottom.
- Each thread contains an arrangement of three signals, differentiated by size.
- A desired order of nine signals is at the bottom.
- Click on a thread to send its next available signal to the bottom. If it matches the next required signal below, that signal will expand to show that it has been received. If the signals do not match, you will incur a strike.
- To complete the module, all nine signals must be received.



On the Subject of Control Flow

- The instruction pointer always begins at the top.
- Click a button on the left to make the pointer jump down (addition) or up (subtraction) the corresponding number of LEDs.
- Each button can only be pressed once.
- After an LED has been visited, it turns on. Attempting to jump to an already lit LED will incur a strike.
- To complete the module, turn every LED on.

