

The Subject of Bamboozling Time Keeper

I needed a few more minutes, not just a few more seconds!

See Appendix A for indicator identification reference.

See Appendix B for battery identification reference.

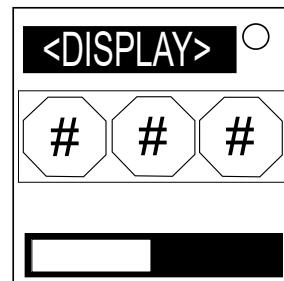
See Appendix C for port identification reference.

See Appendix Two Factor for Two Factor identification reference.

See Appendix Color References for color referencing.

See Appendix One Tap Holds for referring to One Tap Hold.

See Appendix Extended Terminlogy for other terminlogies used in this manual.



This module consists of a screen displaying various phrases; 3 buttons revealed by a sliding hatch, each with a digit on the button; and a stage indicator. The defuser can switch between the 2 stages by pressing the respective half at the bottom indicator. A flashing yellow bar represents the current stage and a green bar represents a completed stage. To complete a stage, the defuser must hold or tap the button at a very specific time and release the button correctly. Complete both stages to disarm the module.

A strike incurred from this module will NOT generate a new set of buttons for both stages, generate a new starting value for both stages, nor reset the progress of the number of stages completed. If you have to hold the button for that stage, the condition to release will alter upon holding the button again.

For All Stages

Each screen will display 3-7 parts of the display ranging from a set of possible values.

All displays follow these rules:

1. All displays will ALWAYS start with a phrase from Table Start in white.
2. The last phrase will always read "POINT ZERO" in white.
3. The phrase "HUNDRED" will ALWAYS be white.
4. All other phrases will be colored in one of the 8 possible colors.

Table Start

| Undefinable Amount | | | Definable Amount | | |
|--------------------|---------------|-----------|------------------|---------------|--------------|
| SOME NUMBERS | THE NUMBERS | NUMBERS | TWO NUMBERS | THREE NUMBERS | FOUR NUMBERS |
| SOME NUMBER(S) | THE NUMBER(S) | NUMBER(S) | 2 NUMBERS | 3 NUMBERS | 4 NUMBERS |
| SOME NUMBER | THE NUMBER | NUMBER | ONE NUMBER | A NUMBER | 1 NUMBER |

An "**END**" denotes to stop on the given step. Upon approaching, that value up to this point in the calculations will be your final value. Unless the final value is less than 10, repetitively add 16 until it isn't.

Process for All Stages:

How much calculations do I have to do again? Sin(Cos(Tan(ALGORITHMS + CALCULUS)))?

Stages 1 and 2 are referred by the left and right halves of the stage indicator.

1. Get the starting value from the display on that stage. Refer to "Appendix: Grabbing the Starting Value" for examples.
2. Subtract the sum of the first 3 base-36 digits in the serial number.
3. Add the product of 4 and the sum of the number of ports on the bomb and the number of port plates on the same bomb.
4. Multiply the number by the positive difference between the number of modules on the bomb with the word "Forget" in its name (1 minimum, "Forget Them All" counts as 5, "Forget Me Now" counts as 2) and the number of modules made by SpeakingEvil on the bomb (1 minimum, "UltraStores" count as 5, "Simon Stores" count as 2).
5. If there are no modules named "Simon Stops", divide the current number by 2, dropping remainders and/or decimal answers. Otherwise, multiply the number by 1.5, dropping decimals after the operation.
6. If all 3 buttons on the current stage have the same color, set the correct button to the position of the current stage's number from the left.
7. If the 2nd phrase is not red and at least 1 button is red, add the starting value from the other stage. (I.E. Stage 1's condition is met, add stage 2's starting value and vice versa.)
8. Add the product of the number of "RT Controlled"^{*} modules on the bomb and the number of batteries on the bomb.
9. If the middle button is green or magenta, change or set the correct button to be this button for the current stage.
10. If the current value is within 5 of a number divisible by 24 and 49, **END**.
11. Add the day of the month that the bomb was started on.
12. Add 20 for each lit vanilla indicator on the bomb and subtract 20 for each unlit vanilla indicator on the bomb.
13. Get the number of modules that contain exactly 3 stages (1 minimum)^t. If the starting value is less than 5000, multiply the current value by this number, otherwise add this number.
14. Subtract 6 for every modded port on the bomb.
15. Go to "Process for Stage 1" if you are calculating on the first stage.
16. Otherwise, go to "Process for Stage 2" if you are calculating on the second stage.

^{*} See Appendix RT Controlled Modules for a list of RT controlled modules.

^t See Appendix 3 Stage Modules for a list of 3 stage modules.

Process for Stage 1:

16. Get the current value from the previous steps.
17. Use the table on the next page to modify the current value or to modify the correct button.
 - Use the 2nd part's phrase with the 2nd part's color, 3rd part's phrase with the 3rd part's color etc until you used all parts for this step or have reached a "**STOP.**"
 - Don't use the part if the phrase is not on the table.
18. If the starting phrase is a definable amount, add the alphabetical position of the first letter on each unlit indicator.
19. If the phrase "HUNDRED" is included, divide your current value by 100. Drop remainders.
20. If there are no modules named "Simon's Stages" but there is a module named "Übermodule," **END**.
21. If there are needies on the bomb, add the product of 99 and the number of needy modules on the bomb.
22. If the starting phrase contains parathenses or a numeral digit, perform the operations from Step 17 with each button's label and color.
23. If your current value is less than 0, multiply this by -6.
24. Add the sum of the last 3 base-36 digits in the serial number.
25. **END**

On the next page's table:

- The row on the top dictates the color of the current part.
- The column on the left dictates the exact phrase that was shown for the current part.
- "0" on the left column starts with an index of 0.
- "RED" on the top row starts with an index of 0.
- If the cell contains an operator and a number, you must preform the given operation with that number and your current value. Drop reminders and decimals after each division operation.
- If the cell is "B#", the correct button is changed to where "#" is the position of the correct button starting from the left. This does not affect your current value unless otherwise stated.
- A "**STOP**" will stop you from reading this table. This seems obvious enough unless otherwise stated.

| | RED | YELLOW | GREEN | CYAN | BLUE | MAGENTA | WHITE | BLACK |
|-----------|------|--------|-------|------|------|---------|-------|-------|
| 0 | STOP | *5 | +50 | /2 | -25 | B3 | B1 | B2 |
| 1 | +1 | STOP | -1 | B2 | B1 | *3 | /3 | B3 |
| 2 | -10 | +25 | STOP | -25 | +66 | -512 | +12 | /5 |
| 3 | B1 | /2 | *2 | STOP | B2 | /9 | *6 | B3 |
| 4 | B3 | B2 | B1 | B2 | STOP | B3 | B1 | B2 |
| 5 | +559 | -58 | +25 | -48 | +99 | STOP | -454 | *5 |
| 6 | *3 | /3 | *6 | /6 | +9 | -6 | STOP | -3 |
| 7 | B1 | +158 | B2 | -264 | *56 | /7 | B3 | STOP |
| 8 | -65 | +46 | -415 | +48 | -56 | -99 | +284 | -54 |
| 9 | B1 | B1 | B1 | B1 | B1 | B1 | B1 | B1 |
| 10 | *10 | /10 | *10 | /10 | *10 | /10 | -100 | -100 |
| 11 | +84 | -46 | +94 | +54 | +49 | +21 | +86 | *11 |
| 12 | /4 | *3 | /3 | *4 | /2 | *2 | /6 | *6 |
| 13 | +485 | +84 | +95 | +24 | -48 | -78 | -459 | -101 |
| 14 | +459 | -485 | -458 | +478 | +0 | -45 | *5 | /5 |
| 15 | +45 | -90 | +30 | +120 | -150 | +15 | +15 | +15 |
| 16 | *2 | +256 | -32 | +64 | -64 | +32 | -256 | /2 |
| 17 | +456 | -98 | -485 | +741 | -46 | +155 | -45 | +485 |
| 18 | +45 | *5 | /9 | +45 | -5 | +46 | +78 | -87 |
| 19 | B3 | B3 | B3 | B3 | B3 | B3 | B3 | B3 |
| 20 | STOP | -59 | /3 | +55 | -20 | +23 | +59 | -41 |
| 30 | +45 | STOP | -89 | *7 | /5 | +48 | +89 | -78 |
| 40 | -512 | +126 | STOP | *2 | /5 | *5 | /2 | +155 |
| 50 | *5 | /5 | *10 | STOP | /10 | +1 | -151 | +150 |
| 60 | *5 | *12 | -120 | -60 | STOP | +180 | /5 | /12 |
| 70 | *2 | *5 | *7 | +21 | /2 | STOP | /5 | /7 |
| 80 | *2 | /2 | *2 | /2 | *2 | /2 | STOP | +40 |
| 90 | B1 | B2 | B3 | B3 | B2 | B3 | B1 | STOP |
| ZERO | /5 | -50 | *2 | +25 | B1 | B2 | B3 | STOP |
| ONE | -1 | +1 | B3 | B2 | /3 | *3 | STOP | B1 |
| TWO | +10 | -25 | +25 | -66 | +512 | STOP | -12 | *5 |
| THREE | B3 | *2 | /2 | B1 | STOP | *9 | /6 | B2 |
| FOUR | *4 | /4 | +4 | STOP | -4 | *4 | /4 | +0 |
| FIVE | -559 | +58 | STOP | -25 | +48 | -99 | +454 | /5 |
| SIX | +3 | STOP | /3 | *3 | /6 | *6 | -9 | +6 |
| SEVEN | STOP | B3 | -158 | B1 | +264 | /56 | *7 | B2 |
| EIGHT | +65 | -46 | +415 | -48 | +56 | +99 | -284 | +54 |
| NINE | B2 | B2 | B2 | B2 | B2 | B2 | B2 | B2 |
| TEN | /10 | *10 | /10 | *10 | /10 | *10 | +100 | +100 |
| ELEVEN | -84 | +46 | -94 | -54 | -49 | -21 | -86 | /11 |
| TWELVE | +4 | -3 | +3 | -4 | +2 | -2 | -6 | +6 |
| THIRTEEN | -485 | -84 | -95 | -24 | +48 | +78 | +459 | +101 |
| FOURTEEN | -459 | +485 | +458 | -478 | -0 | +45 | /5 | *5 |
| FIFTEEN | -45 | +90 | -30 | -120 | +150 | -15 | -15 | -15 |
| SIXTEEN | /2 | -256 | +32 | -64 | +64 | -32 | +256 | *2 |
| SEVENTEEN | -456 | +98 | +485 | -741 | +46 | -155 | +45 | -485 |
| EIGHTEEN | -45 | /5 | *9 | -45 | +5 | -46 | -78 | +87 |
| NINETEEN | +0 | -25 | +25 | -25 | +25 | -25 | +25 | -0 |
| TWENTY | *2 | /2 | *5 | /5 | *10 | /10 | +20 | STOP |
| THIRTY | +330 | -330 | +90 | -90 | /3 | *3 | STOP | /10 |
| FOURTY | +512 | -126 | /2 | *5 | /5 | STOP | *2 | -155 |
| FIFTY | /5 | *5 | /10 | *10 | STOP | -1 | +151 | -150 |
| SIXTY | /5 | /12 | +120 | STOP | +60 | -180 | *5 | *12 |
| SEVENTY | /2 | /5 | STOP | /7 | -21 | *2 | *5 | *7 |
| EIGHTY | /2 | STOP | *2 | /2 | *2 | /2 | *2 | -40 |
| NINETY | STOP | B1 | B3 | B2 | B1 | B3 | B2 | B1 |

Process for Stage 2:

17. Get the current value from the previous steps.
18. If the first display is a definable amount, take the next screen's part and color and preform the operation from the previous table. Drop remainders after preforming the operation. **HOWEVER:**
 - If the operator is "**STOP**", divide your current value by 17. Drop remainders.
 - If the operator is "B#" multiply your current value by (#+1).
19. If the starting phrase is in the last row of Table Start, perform step 18 except use the 2nd to last part, unless the 2nd to last part is "HUNDRED", multiply your current value by 100 instead.
20. If the starting phrase contains parentheses or a numerical digit, do the next step. Otherwise, skip to step 22.
21. Perform Step 18 except with the cell at ($([P2_1]+[P2_2])\%56$, $7-[MBC_2]$); where $P2_n$ is the index of the 2nd phrase for the given stage and MBC_n is the color index of the middle button in the given stage; and with the cell at ($[[P2_1]-[P2_2]]$, $([LBC_2]+[RBC_2])\%8$); where RBC_n is the color index of the right button for the given stage, LBC_n is the color index of the left button for the given stage; in (row,col) format.
22. If the left and right buttons have the same color as the 2nd part of the display, but the middle button is a different button, change the correct button to be the middle button.
23. If step 9 was applied to stage 1, add 25. Otherwise if step 9 was applied to stage 2, add 150. Otherwise, subtract 250.
24. Subtract the product of the number of Kristy's modules on the bomb and 10.
25. If your current value is less than 0 at this point, multiply that value by -65 and **END**.
26. If by adding up the number of letters in each of the color's names from each button, you get a number greater than 12, divide your current value by the sum of the number of letters in each of colors' names of the non-white parts on stage 2. If you are attempting to divide by 0, double the current value instead.
27. If there are Two Factors on this bomb, add the product of the number of Two Factors on the bomb and 50 and **END**.
28. If there are 7 parts in this stage, divide the current value by 7, dropping remainders if present.
29. Change the correct button to the button with the most letters in the name of the button's color, unless there is a tie or an RT sensitive module is present.
30. Divide your value by 3, drop remainders if present.
31. Add product of 10 and the number of modded and vanilla indicators that share at least 1 letter in the phrase, "SPEAKINGEVIL" and **END**.

The Scale Factor

Now that you have made your final value for both stages, you must figure out a set of times that are calculated in the following format: $F_v \times X^N$, where N is any integer, F_v is the final value for that stage. To figure out X for each stage, you must use the table underneath. All operators with an "=" denotes to set the value as such. **Drop all modifications if the "=" is satisfied.** Drop remainders after each "/" operation if present. Note that each starting phrase only refers to the starting phrase on the given stage, not the other stages. Perform each operator from top to bottom.

| Stage 1 | | Stage 2 | |
|--|--------------------|--|---------------|
| This stage's starting value < 5000 | +1 | This stage's starting value < 5000 | +1 |
| Batteries | -1 per 2 batteries | Battery Holders | -1 per holder |
| "Forget This" is present | +1 | "Forget Enigma" is present | +1 |
| 4 or more Serial Number Digits | +3 | 4 or more Serial Number Letters | +3 |
| "The Time Keeper" is present | +2 | "Turn The Key" is present | +2 |
| Starting Phrase is not considered definable | +2 | Starting Phrase is considered definable | +2 |
| The bomb's starting time is 30:00 or greater | +2 | The bomb's starting time is 30:00 or greater | +2 |
| Exactly 3 Serial Number Letters | +2 | Exactly 3 Serial Number Digits | +2 |
| "Forget Me Not" is present | +1 | "Forget It Not" is NOT present | +1 |
| This stage's final value \geq 10000 | /2 | This stage's final value \geq 10000 | /2 |
| Exactly all of these: Lit BOB, Lit FRK, 4 batteries in 2 holders | | =2 | |

After using the tables, set X to 2 if X < 2 or X to 5 if X is > 5.

Determining the Correct Button

If you still have not found a correct button for both stages, refer to these for each stage:

1. If your final value for the given stage is between 0 and 9999 inclusive, the correct button is the left button.
2. Otherwise, if there is a parallel port on the bomb and exactly 1 magenta button is present, that magenta button is the correct button.
3. Otherwise, if there is a Stereo RCA port on the bomb and exactly one white button and one red button is present, the correct button is neither the white button nor the red button.
4. Otherwise, if there are no black colored parts and exactly one of the buttons is black, that button is correct.
5. Otherwise, if there are no shared colors between the buttons and the phrases (excluding white), the correct button is the right button.
6. Otherwise, if all of the phrases are white, and exactly one button's label is not present in numerical form or digit form, that button is the correct button.
7. Otherwise, if the buttons are red, green, and blue, in any order, the correct button is the blue button.
8. Otherwise, if all of the buttons' colors' name has a cipher module made by Sean (except Cyan, which is bundled within a big cipher module), the correct button is the middle button.
9. Otherwise take your final value and modulo by 3. Add one to this to get the number of buttons to count to get your correct button.
10. Count from left to right if you are on stage 1, right to left if you are on stage 2 with the number you got from the previous step.

Determining the State of the Correct Button

Now that you have determined the correct scale factor, value, and the correct button, note as many possible times as possible from the given value and possible scale factors. Round down if you are dividing the value by the scale factor. Then refer to the conditions underneath when pressing the button at the right time. A tap can be substituted as a hold at a cost of a strike if the bomb's timer is 120 or fewer seconds away from blowing up. The error margin for starting a tap or a hold are within 5 seconds on the bomb's timer. Refer to the conditions below for each stage.

1. If the module "The Very Annoying Button" is present or there is at least 1 needy module on the bomb, tap the correct button.
2. Otherwise, if both of the other button's labels are present in the display in either numerical form or digit form, hold the correct button.
3. Otherwise, if the correct button's label is present in the display in either numerical form or digit form, tap the correct button.
4. Otherwise, if the button's labels are sorted from greatest to least or least to greatest, hold the correct button.
5. Otherwise, if the other button's labels' positive difference is within 2 of the correct button's label, tap the correct button.
6. Otherwise, if the correct button's label is equal to the least significant digit of the sum of the other 2 button's labels, hold the correct button.
7. Otherwise, if the correct button's label is equal to the least significant digit of the product of the other 2 button's labels, tap the correct button.
8. Otherwise, hold the correct button.

The Override

If one of the stage's starting values is 2424, the other stage's starting value is 4949, an indicator labeled SIG is present, and a cyan button is present on both stages, you are blessed by Sig, a Puyo Puyo character whose obsession is with bugs. For both stages, the final value is overridden to 15 seconds, with the scale factor of 2, and the correct button is the leftmost cyan button, which must be tapped instead. If the defuser starts the bomb on the 9th of April, the 4th of February, or the 16th of June something special can happen.

Holding The Button:

You got this far... For what?

Upon holding the button, the defuser will hear a hum that gradually gets louder until the popping sound is heard. If you release the button before the popping sound is heard, the module will intercept the button interaction as a tap. Once the popping sound has occurred, the other buttons not held will start flashing a set of up to 4 primary colors. The other buttons rest for a short bit of time in black before repeating the flashes. The display will also show a timer counting up in the format "MM:SS."

"Bomb timer" will refer to the current time on the bomb, while "Display" will refer to the display's timer. If the current condition gives a value that is impossible to reach, skip to the next "Otherwise," when necessary.

1. If both buttons that are not held are flashing consistently the same color as its previous flashes four times in a row, use the table below to release the button:

| Leftmost button's 4 flashes | Rightmost button's 4 flashes | Release Condition |
|-----------------------------|------------------------------|---|
| Red | Red | Display's timer have at least 2 distant digits in common with bomb's timer |
| | Green | Display's second timer shows a product of 2 consecutive numbers (1*2, 2*3, 3*4, etc.) |
| | Blue | Bomb's seconds timer shows a product of 2 consecutive numbers (1*2, 2*3, 3*4, etc.) |
| Green | Red | At least 1 digit from the local time is shown from the bomb's timer (not 0) |
| | Green | The last digit from both the seconds timer from the display and the bomb are within 1 from each other (0/9 pairs are considered within 1 apart) |
| | Blue | At least 1 digit from the local time is shown from the display's timer (not 0) |
| Blue | Red | 04 or 40 on the display's seconds timer |
| | Green | 40 or 04 on the bomb's seconds timer |
| | Blue | 0 and 4 NOT present on both the display and the bomb's seconds timer |

Holding The Button (Continued):

2. Otherwise, if exactly 1 button is consistently flashing the same color four times in a row, refer to the table below to release the button:

| The Shared Flashing Color | Position of Flashing Button | | |
|---------------------------|--|--|--|
| | Left | Middle | Right |
| Red | Bomb timer's seconds digits are not in serial no. | Display's seconds timer shows part of a code from The Swan | At least 1 bomb timer's seconds digit in serial no. |
| Green | Bomb timer's seconds digits add up to a prime number | Matching seconds' digits from bomb's timer | Bomb timer's seconds digits don't add up to a prime number |
| Blue | Display's seconds timer add up to a prime number | Bomb's seconds timer shows part of a code from The Swan | Display's seconds digits not in serial no. |

3. Otherwise, if there are Two Factors on this bomb:

1. If there is exactly 1 Two Factor on this bomb, release the button when the display's seconds timer contains at least 1 digit from the rightmost digit in the Two Factor.
2. Otherwise, if there are exactly 2 Two Factors on this bomb, release when the bomb's second digits contain at least 1 digit from the average of the Two Factors, modulo 10. Round down when averaging. Unless the Two Factors are equal, go to the next "Otherwise."
3. Otherwise, release when the bomb's timer and the display have combined at least 4 digits in the highest Two Factor on the bomb. (02:45 on display, 65:31 on bomb, 5419879 being highest Two Factor, counts as having at least 4 digits) Unless the highest Two Factor on the bomb has 3 or fewer distant digits, go to the next "Otherwise."

Holding The Button (Continued)

4. Otherwise, if at least 1 button not held is flashing irregularly, follow these instructions.
- A button flashing irregularly is indicated by uneven times inbetween each flash, excluding the delay before the sequence of flashes repeat, which is longer.
 - The button flashing irregularly is flashing in Morse code.
 - If both buttons are flashing irregularly, use both.
 - Take the alphabetical position of the button that is flashing irregularly as the initial value.
 - Modify that initial value based on Table Irregular Flashes, one at a time.

Table Irregular Flashes

| Color of the Given Flash | Position of the Flash | | | |
|-----------------------------------|--|--|---|---|
| | Flashed 1st | Flashed 2nd | Flashed 3rd | Flashed 4th |
| Red | Add the number of modules on the bomb. | Add the digital root of the number of modules on the bomb. | Subtract the number of modules on the bomb. | Subtract the digital root of the number of modules on the bomb. |
| Green | Add the month of the bomb that was started on. | Add the digital root of your current value. | Subtract the month of the bomb that was started on. | Subtract the number of digits in your current value. |
| Blue | Add the number of ports on the bomb. | Subtract the digital root of your current value. | Subtract the number of ports on the bomb. | Add the number of digits in your current value. |

- If your value is currently less than 0, take the absolute value of that.
- Then take the least significant digit in your value(s) and release the button based on the next table.

| Values Used | Release Condition |
|-------------------------------|---|
| Exactly 1 value used | At least 1 bomb timer's seconds digit shows THAT value |
| 2 values used, 1 value is a 0 | The other non-zero value is not present on the display's seconds digits |

Holding The Button (Continued)

| Values Used | Release Condition |
|---|--|
| 2 values used, both values are identical and NOT 0 | The value is not present on the bomb timer's seconds digits |
| 2 values used, both values are 0 | Bomb's seconds timer shows the sum of the base 36 digits flashed in Morse, modulo 60 |
| 2 values used, 0 or 2 buttons flashed a vowel (excluding Y) | Bomb's seconds timer shows the product of the 2 values |
| 2 values used, Otherwise | Display's seconds timer shows the product of the 2 values |

5. Otherwise, take the count the number of flashes for each flashing button and use the next table.

| | Red | Green | Blue |
|---------------------------|-----|-------|------|
| Left Button is flashing | 1 | 2 | 3 |
| Middle Button is flashing | 2 | 3 | 1 |
| Right Button is flashing | 3 | 1 | 2 |

Add all of these valued flashes up for each button and modulo them by 5. Use each of these to get the conditions from the next table:

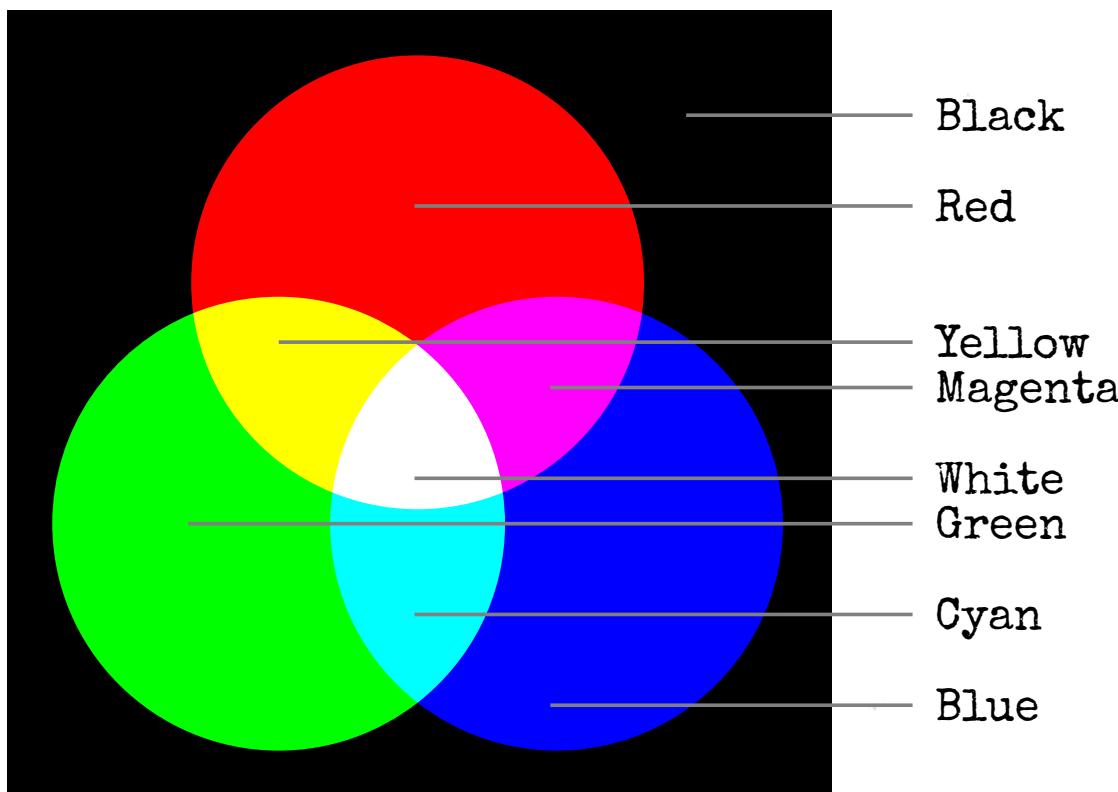
| Value % 5 | Release Condition from Bomb Timer |
|-----------|--|
| 0 | Total time remaining in seconds is divisible by 2 |
| 1 | Total time remaining in seconds is divisible by 3 |
| 2 | Total time remaining in seconds is divisible by 5 |
| 3 | Total time remaining in seconds is divisible by 7 |
| 4 | Total time remaining in seconds is divisible by 11 |

Both conditions must be satisfied from the calculated values on each of the flashing buttons. However, if the values after modulo 5 are the same, release the button when the total time remaining in seconds is a prime number or less than 5. You might need [Prime Checker](https://ktane.timwi.de/HTML/Prime%20Checker.html)

(<https://ktane.timwi.de/HTML/Prime%20Checker.html>) to assist you with this.

Appendix Color References: Color Referencing

For those who struggle to refer to the colors.



- Primary colors referred from the manual are red, green and blue.
- For the black text, the screen will momentarily flash white to indicate a black colored text.
- If colorblind mode is enabled on this module, the module will add "IN X" where X is the color of the previous text, unless the rules state a specific color.
- E.G. If the color of the text for "THREE" is magenta the module will display "THREE" first, then "IN MAGENTA," and then the rest of the text.

Example Text Displays without colorblind mode:

1. 1 NUMBER:
2. THIRTY
3. THREE

Example Text Displays with colorblind mode:

- | | |
|---------------|-------------|
| 1. 1 NUMBER: | 4. THREE |
| 2. THIRTY | 5. "IN RED" |
| 3. "IN GREEN" | |

Appendix: One Tap Holds

Carpal tunnel is a problem.

In the config, the defuser can find a setting for this module called "OneTapHolds." If enabled this will allow the module to be put into a hold state by just tapping the button and exit out of a holding state by tapping the button while the button is being "held."

This means that you can tap on the button to set the button into a hold state and not have to strain your finger for this module. By default this is forced on for Twitch Plays and left off otherwise. If this is enabled and you wish to tap the button instead, you have to tap the button twice before the display starts counting.

Appendix: Grabbing the Starting Value

Example displays and results can be found here.

| Display | Number |
|--------------------------------|--------|
| ONE | 1 |
| 3 | 3 |
| FIFTY | 50 |
| 50 TWO | 52 |
| 6 HUNDRED EIGHT | 608 |
| FOUR HUNDRED | 400 |
| TEN HUNDRED | 1000 |
| 50 6 HUNDRED | 5600 |
| 90 SEVEN HUNDRED | 9700 |
| NINETY SEVEN HUNDRED SIXTY ONE | 9761 |
| 13 HUNDRED 15 | 1315 |
| 20 9 HUNDRED 16 | 2916 |
| 20 HUNDRED 5 | 2005 |

Note that 1000 is actually written as ONE THOUSAND in numerical form however the displays will NEVER show "THOUSAND" in any instances.

Appendix: Extended Terminology

RT Controlled: An RT Controlled module is a module that has a built-in internal timer that only starts ticking as soon as the user has finished prepping the module and has pressed the "start" button. Notable RT controlled modules are The Wire, Module Homework, Burger Alarm, Burglar Alarm, Fast Math, and The Stopwatch. If the module requires holding for a specific amount of time to disarm itself, the bomb's timer, or start its internal timer immediately as the bomb loads or arms itself, these modules are NOT RT controlled modules. Needies are also NOT RT controlled modules since they activate by time, strikes, or upon solving a few modules.

RT Sensitive: An RT Sensitive module is a module that has a built-in internal timer that starts ticking as soon as the bomb starts or as the bomb loads. Notable RT sensitive modules are The Swan, Tax Returns, and The Very Annoying Button. Needies are also NOT RT sensitive modules since they activate randomly by time, strikes, or upon solving a few modules. "RT strike" is said to determine if the strike was based on running out of internal time set up by the module. An RT strike can repeat multiple times if the RT sensitive module is left unhandled for too long. Some RT sensitive modules may not strike but instead reset which makes the defuser have to reread the contents in the module again.

Appendix SpeakingEvil's Modules

This list was last updated on February 7th, 2020.

- 14
- Affine Cycle
- Bamboozled Again
- Bamboozling Button
- Bamboozling Button Grid
- Bamboozling Time Keeper
- Bordered Keys
- Caesar Cycle
- Cryptic Cycle
- Disordered Keys
- Double Arrows
- Faulty RGB Maze
- Forget Me Later
- Hill Cycle
- Jumble Cycle
- Misordered Keys
- Ordered Keys
- Pigpen Cycle
- Playfair Cycle
- Recorded Keys
- Reordered Keys
- RGB Maze
- Simon Stores
- Silhouettes
- Tallordered Keys
- Ultimate Cycle
- UltraStores
- Unordered Keys
- The Very Annoying Button

Appendix: List of 3 Stage Modules

There's so many of them!

A 3 stage module is classified as a relatively repetitive procedure that must be done successfully exactly 3 times to disarm a module. Note that the module must also have exactly 3 stages for this to count. The actions needed to submit cannot completely be unique to the 2 other stages to classify a module as a 3 stage module.

This list was last updated on February 7th, 2020.

- 3D Tunnels
- Algebra
- Alphabetical Ruling
- Binary Grid
- Binary Tree
- Book of Mario
- Challenge & Contact
- Color Decoding
- Cruel Keypads
- Divisible Numbers
- Edgework*
- Encrypted Dice
- English Test
- Find The Date
- Flavor Text EX
- Guitar Chords
- Ice Cream
- Kanji
- Krazy Talk
- Logical Buttons
- The London Underground
- Modern Cipher
- Module Movements
- Monsplode Trading Cards
- Number Nimbleness
- Orange Arrows
- Ordered Keys
- Partial Derivatives
- Passport Control
- Poetry
- Prime Checker
- The Screw
- Sea Shells
- Simon Samples
- Simon Screams
- Simon Selects
- Simon Shrieks
- Simon Sings
- Simon's On First
- Simon Stops
- Simon Stores
- Sonic the Hedgehog
- Symbolic Coordinates
- Third Base
- Timing is Everything
- UltraStores
- Wavetapping
- Who's on First
- Who's on First Translated
- Zoni

* This module's manual misleads the number of stages present but actually meets the criteria of a "3 Stage Module". A struck out name of a module indicates that the module is currently blanked/inaccessible from the Steam Workshop.

Appendix: Kritzy's Modules

This list was last updated on February 7th, 2020.

- Blackjack
- Command Prompt
- Connection Device
- Flip The Coin
- ~~Four-Card Monte~~
- Hold Ups
- Lockpick Maze
- Mirco-Modules
- Module Homework
- The Radio
- Scripting

Appendix: List of RT Controlled Modules

This list was last updated on February 7th, 2020.

- Brush Strokes
- Burglar Alarm
- Burger Alarm
- Countdown
- Cruel Countdown
- Coffeebucks
- The Crystal Maze
- Fast Math
- Going Backwards
- Jack Attack
- Lockpick Maze
- Manometers
- Module Homework
- Number Nimbleness
- QuizBuzz
- Simon Stops
- Sonic and Knuckles
- The Stopwatch
- Valves
- The Wire
- Zoo

Appendix: RT Sensitive Modules

This list was last updated on February 7th, 2020.

- Blinkstop
- Hyperactive Numbers
- Lunchtime
- The Number Cipher
- The Swan
- Tax Returns
- The Very Annoying Button
- Ultimate Custom Night

Appendix: Morse Code

How did that not fit in the main page again?

| | |
|---|-----------|
| A | ● — |
| B | — ● ● ● |
| C | — ● — ● |
| D | — ● ● |
| E | ● |
| F | ● ● — ● |
| G | — — ● |
| H | ● ● ● ● |
| I | ● ● |
| J | ● — — — — |
| K | — ● — |
| L | ● — ● ● |
| M | — — |
| N | — ● |
| O | — — — |
| P | ● — — — ● |
| Q | — — — ● — |
| R | ● — ● |
| S | ● ● ● |
| T | — |

| | |
|---|---------|
| U | ● ● — |
| V | ● ● ● — |
| W | ● — — |
| X | — ● ● — |
| Y | — ● — — |
| Z | — — ● ● |

| | |
|---|-------------|
| 1 | ● — — — — |
| 2 | ● ● — — — |
| 3 | ● ● ● — — |
| 4 | ● ● ● ● — |
| 5 | ● ● ● ● ● |
| 6 | — ● ● ● ● |
| 7 | — — — ● ● |
| 8 | — — — — ● |
| 9 | — — — — — ● |
| 0 | — — — — — |