# On the Subject of Bamboozled Again

"Wait, was that the letter Echo or the word E?", you say,
as you are baffled, befuddled, and bemused.
This module will beguile, and buffalo, and bewilder,
to make sure it is **never** defused.

This module consists of a six coloured buttons, each with a line of text written on them, and a screen that displays a message that is broken into eight parts.
Each part of the message is encrypted using three key values: A, B and, C in the following way:

1. Each character, including spaces, is shifted A characters to the left.
   (0 <= A < No. of characters in text)
2. A pair of symbols is appended to the ends of the text.
3. Each character, including symbols, is Caesar shifted B letters/symbols forwards. (0 < B <= 26)
4. The text is transcribed using one of six different sets of glyphs; the set used gives the C value.

Each unencrypted text, excluding texts 2 and 4, have values that are modified by an operation corresponding to that text's colour.

Using these values, together with A, B, and C, gives the final value of each text.

Each button has an initial value, given by its colour and the text written on it.

Using these values, together with the final values of the display texts, gives the final value of each button.

The correct buttons to push are given by thir final values.

Use the text on the buttons, their colours, and the symbols added to each of the display texts at step 2 of the encryption, to find the correct times to push the buttons.

Pushing a button will cause an LED to turn on. Once all four LEDs are on, the inputs will be submitted.

The LEDs will change colour according to the submitted inputs:
- Green - The correct button was pressed at the right time.
- Yellow - The correct button was pressed at the wrong time.
- Red - The wrong button was pressed.

If all four LEDs turn green, the module is solved.
Otherwise, if none of the LEDs turn red, the module will reset but a strike will not be issued.
Otherwise, the module will reset **and** a strike will be issued.

*Additional Module Info:*

The LEDs can be pressed at any time to affect the display cycle:

- Left - Cycles to the previous text while paused.
- Mid-left - Resumes automatic text cycling.
- Mid-Right - Pauses text cycling
- Right - Cycles to the next text while paused.




## Section 1: What to press

### Subsection 1.1: Glyph tables

The tables below show the glyphs for both the letters and symbols:

- Symbols are represented by the same glyphs regardless of which set they belong to.
- All glyphs are the same size on the display.
- '#' is used to represent spaces.
- Letters and symbols are independently shifted down their respective tables by step 3 of the encryption.

| | Set A | Set B | Set C | Set D | Set E | Set F |
|---|---|---|---|---|---|---|
| **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** | | | | | | |
| **Value C** | 11 | 12 | 13 | 14 | 15 | 16 |



| Symbol Table | |
|---|---|
| **#** | |
| **'** | |
| **"** | |
| **?** | |
| **-** | |
| ***** | |
| **~** | |
| **!** | |



## Subsection 1.2: Raw data

The unencrypted display texts and button texts can be found in the set below.
Each text has a corresponding raw value, R:

- Display texts 2 and 4 do not have raw values and will always be either **THEN** or **NEXT**.
- Display texts 1, 3, and 5 can all be found in the top row of the set.
- Display texts 6, 7, and 8 cannot be found in the top row of the set.

| THE LETTER | ONE LETTER | THE COLOUR | ONE COLOUR | THE PHRASE | ONE PHRASE |
|---|---|---|---|---|---|
| 40 | 24 | 32 | 39 | 20 | 15 |

| ALPHA | BRAVO | CHARLIE | DELTA | ECHO | GOLF | KILO | QUEBEC | TANGO |
|---|---|---|---|---|---|---|---|---|
| 70 | 84 | 83 | 61 | 66 | 46 | 68 | 56 | 80 |

| WHISKEY | VICTOR | YANKEE | ECHO ECHO | E THEN E | ALPHA PAPA | PAPA ALPHA | PAPHA ALPA | T GOLF |
|---|---|---|---|---|---|---|---|---|
| 54 | 65 | 41 | 84 | 60 | 56 | 86 | 69 | 50 |

| TANGOLF | WHISKEE | WHISKY | CHARLIE C | C CHARLIE | YANGO | DELTA NEXT | CUEBEQ | MILO |
|---|---|---|---|---|---|---|---|---|
| 62 | 78 | 64 | 43 | 41 | 51 | 47 | 57 | 45 |

| KI LO | HI-LO | VVICTOR | VICTORR | LIME BRAVO | BLUE BRAVO | G IN JADE | G IN ROSE | BLUE IN RED |
|---|---|---|---|---|---|---|---|---|
| 46 | 86 | 84 | 82 | 78 | 47 | 59 | 63 | 42 |

| YES BUT NO | COLOUR | MESSAGE | CIPHER | BUTTON | TWO BUTTONS | SIX BUTTONS | I GIVE UP | ONE ELEVEN |
|---|---|---|---|---|---|---|---|---|
| 89 | 77 | 70 | 55 | 67 | 79 | 71 | 58 | 86 |

| ONE ONE ONE | THREE ONES | WHAT? | THIS? | THAT? | BLUE! | ECHO! | BLANK | BLANK?! |
|---|---|---|---|---|---|---|---|---|
| 58 | 88 | 49 | 78 | 68 | 75 | 45 | 44 | 56 |

| NOTHING | YELLOW TEXT | BLACK TEXT? | QUOTE V | END QUOTE | "QUOTE K" | IN RED | ORANGE | IN YELLOW |
|---|---|---|---|---|---|---|---|---|
| 72 | 70 | 46 | 73 | 66 | 52 | 48 | 69 | 41 |

| LIME | IN GREEN | JADE | IN CYAN | AZURE | IN BLUE | VIOLET | IN MAGENTA | ROSE |
|---|---|---|---|---|---|---|---|---|
| 58 | 84 | 47 | 45 | 55 | 83 | 74 | 51 | 67 |

## Subsection 1.3: Data modification

Modify the raw text values using the rule that corresponds to that text's colour to obtain its modified value, S:

| Text Colour | Modification | Text Colour | Modification |
|---|---|---|---|
| White | Do nothing | Grey | Swap the digits |
| Red | Subtract the first digit | Cyan | Subtract the second digit |
| Orange | Replace the second digit with the first | Azure | Replace the first digit with the second |
| Yellow | Add the second digit | Blue | Add the first digit |
| Lime | Subtract the higher digit | Violet | Subtract the lower digit |
| Green | Subtract the sum of the digits | Magenta | Subtract the difference between the digits |
| Jade | Subtract twice the first digit | Rose | Subtract twice the second digit |



## Subsection 1.4: Final text values

The final value, T, for each of the six texts that can be evaluated, is given by:

$$T = S + 5A + 2(B + C)$$

## Subsection 1.5: Initial button values

Follow the instructions below to compute the initial value, I, of each of the six buttons:

1. 
   - If the button is black, begin with I = 30.
   - Otherwise, if the button is white or grey, begin with I = 20.
   - Otherwise, begin with I = 0.
2. Add 60 for each unencrypted display text that is the same as the text written on the button.
3. Add 15 for each display text that is the same colour as the button.
4. If the button is not grey, add 5 for each display text whose colour is complementary to the colour of the button.

## Subsection 1.6: Final button values

Use the table below to find which display text's final values are $T_1$ and $T_2$ for each button.

| Position of button | $T_1$ | $T_2$ |
|---|---|---|
| TL | Display text 1 | Display text 6 |
| TM | Display text 3 | Display text 7 |
| TR | Display text 5 | Display text 8 |
| BL | Display text 6 | Display text 1 |
| BM | Display text 7 | Display text 3 |
| BR | Display text 8 | Display text 5 |

The final value of each button, F, is then given by the equation:

$$F = 3I + 2(T_1 + T_2)$$

To solve the module, press the buttons with the four highest final value in ascending order.

**Note:** If more than one button has the desired final value, the correct one to push occurs first in reading order. Buttons change their colour and text when pressed and their initial values change accordingly.
This may change which button needs to be pressed next.



## Section 2: When to press

### Subsection 2.1: The first three buttons

### Subsection 2.1.1: Button text modification

Once the correct button to press has been identified, modify the raw value of the text written on that button using the rule corresponding to that button's colour:

| Button Colour | Value X |
|---|---|
| White | The highest digit |
| Red | The first digit subtract the second |
| Orange | The digital root |
| Yellow | The first digit |
| Lime | The first digit subtract the digital root |
| Green | The sum of the digits |
| Jade | Twice the first digit |
| Grey | The sum of the digits subtract the digital root |
| Cyan | The second digit subtract the first |
| Azure | The negative digital root |
| Blue | The second digit |
| Violet | The second digit subtract the digital root |
| Magenta | Ten minus the sum of the digits |
| Rose | Twice the second digit |
| Black | The lowest digit |



### Subsection 2.1.2: Y value computation

- If there are no lit LEDs, the Y value is the current X value.
- If there is one lit LED and, once deciphered,
  - display text 2 is **THEN**, Y is the current X value plus the previous X value.
  - display text 2 is **NEXT**, Y is the current X value minus the previous X value.
- If there are two lit LEDs and, once deciphered,
  - display text 4 is **THEN**, Y is the current X value plus the previous Y value.

- display text 4 is **NEXT**, Y is the current X value minus the previous Y value.

### Subsection 2.1.3: Final computation

The time to push the button is given by the Y value and the symbols appended to the text at step 2 of the the encryption.

- If there are no lit LEDs, use the symbols appended to display text 8.
- If there is one lit LED, use the symbols appended to display text 7.
- If there are two lit LEDs, use the symbols appended to display text 6.

| Symbol | Press the button when.. |
|---|---|
| # | the last digit of the timer is Y mod 10 |
| ' | the sum of the last two digits of the timer is (Y mod 9) + 3 |
| " | the sum of the last two digits of the timer is (2Y mod 9) + 3 |
| ? | the difference between the last two digits of the timer is Y mod 5 |
| - | the last digit of the timer is 9 - (Y mod 10) |
| * | the sum of the last two digits of the timer is 11 - (Y mod 9) |
| ~ | the sum of the last two digits of the timer is 11 - (2Y mod 9) |
| ! | the difference between the last two digits of the timer is 2Y mod 5 |

Page 6 of 8
Keep Talking and Nobody Explodes Mod Bamboozled Again

### Subsection 2.2: The final button

1. - Modify the raw value of the text on the button using the rule corresponding to the colour of display text 2 to obtain $S_1$.
   - Modify the raw value of the text on the button using the rule corresponding to the colour of display text 4 to obtain $S_2$.
2. - Modify $S_1$ using the rule corresponding to the colour of the button to obtain $X_1$.
   - Modify $S_2$ using the rule corresponding to the colour of the button to obtain $X_2$.
3. - If display texts 2 and 4 are the same when deciphered, Y is the sum of $X_1$ and $X_2$.
   - Otherwise, Y is the difference between $X_1$ and $X_2$.
4. Locate the symbol in the table below given by the symbols appended to display texts 2 and 4 at step 2 of the encryption, and press the button when the condition corresponding to that symbol is satisfied.

| | # | ' | " | ? | - | * | ~ | ! |
|---|---|---|---|---|---|---|---|---|
| **#** | # | ' | " | ? | - | * | ~ | ! |
| **'** | ' | ' | ? | - | * | ~ | ! | # |
| **"** | " | ? | " | * | ~ | ! | # | ' |
| **?** | ? | - | * | ? | ! | # | ' | " |
| **-** | - | * | ~ | ! | - | ' | " | ? |
| ***** | * | ~ | ! | # | ' | * | ? | - |
| **~** | ~ | ! | # | ' | " | ? | ~ | * |
| **!** | ! | # | ' | " | ? | - | * | ! |

Page 7 of 8
Keep Talking and Nobody Explodes Mod Bamboozled Again

# Appendix: Button and Display Text Colours