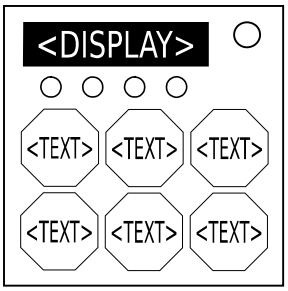
On the Subject of Bamboozled Again



**Dead*.*

This module consists of 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. ($0 \leqq A <$ 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. ($0 < B \leqq 26$)
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 their 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 tables 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 first table.
- Display texts 6, 7, and 8 cannot be found in the first two tables.

Please refer to the original manual with decryption table for more infomation.

Texts on display 1, 3, and 5											
Text	AP(n) − AP(n-1) or AP(1) − AP(9)									AP(4) − AP(1)	R
THE LETTER	14	23	7	19	15	0	11	13	2	18	40
ONE LETTER	25	17	7	19	15	0	11	13	23	23	24
THE COLOUR	14	23	24	12	23	3	6	23	2	9	32
ONE COLOUR	25	17	24	12	23	3	6	23	23	14	39
THE PHRASE	14	23	11	18	10	9	18	12	15	22	20
ONE PHRASE	25	17	11	18	10	9	18	12	10	1	15

Texts on display 2 and 4				
Text	AP(n) − AP(n-1) or AP(1) − AP(4)			
THEN	14	23	9	6
NEXT	17	19	22	20

4 Letters + 0 Symbol					
Text	AP(n) − AP(n-1) or AP(1) − AP(4)				R
ECHO	24	5	7	16	66
GOLF	8	23	20	1	46
KILO	24	3	3	22	68
MILO	22	3	3	24	45
LIME	23	3	18	7	58
JADE	17	3	1	5	47
ROSE	23	4	12	13	67

4 Letters + 1 Symbol					
Text	AP(n) − AP(n-1) or AP(1) − AP(4)				R
KI LO	24	3	3	22	46
HI-LO	1	3	3	19	86
WHAT?	11	19	19	3	49
THIS?	14	1	10	1	78
THAT?	14	19	19	0	68
BLUE!	10	9	10	23	75
ECHO!	24	5	7	16	45

5 Letters + 0 Symbol						
Text	AP(n) − AP(n-1) or AP(1) − AP(5)					R
ALPHA	11	4	18	19	0	70
BRAVO	16	9	21	19	13	84
DELTA	1	7	8	7	3	61
TANGO	7	13	19	8	5	80
YANGO	2	13	19	8	10	51
BLANK	10	15	13	23	17	44
AZURE	25	21	23	13	22	55

5 Letters + 1 Symbol						
Text	AP(n) − AP(n-1) or AP(1) − AP(5)					R
T GOLF	13	8	23	20	14	50
IN RED	5	4	13	25	5	48
5 Letters + 2 Symbols						
Text	AP(n) − AP(n-1) or AP(1) − AP(5)					R
BLANK?!	10	15	13	23	17	56

6 Letters + 0 Symbol							
Text	AP(n) – AP(n-1) or AP(1) – AP(6)						R
QUEBEC	4	10	23	3	24	14	56
VICTOR	13	20	17	21	3	4	65
YANKEE	2	13	23	20	0	20	41
WHISKY	11	1	10	18	14	24	64
CUEBEQ	18	10	23	3	12	12	57
COLOUR	12	23	3	6	23	11	77
CIPHER	6	7	18	23	13	11	55
BUTTON	19	25	0	21	25	14	67
ORANGE	3	9	13	19	24	10	69
VIOLET	13	6	23	19	15	2	74

6 Letters + 1 Symbol							
Text	AP(n) – AP(n-1) or AP(1) – AP(6)						R
QUOTE V	4	20	5	11	17	21	73
IN CYAN	5	15	22	2	13	21	45
IN BLUE	5	14	10	9	10	4	83

6 Letters + 2 Symbols							
Text	AP(n) – AP(n-1) or AP(1) – AP(6)						R
E THEN E	15	14	23	9	17	0	60

6 Letters + 3 Symbols							
Text	AP(n) – AP(n-1) or AP(1) – AP(6)						R
"QUOTE K"	4	20	5	11	6	6	52

7 Letters + 0 Symbol								
Text	AP(n) – AP(n-1) or AP(1) – AP(7)							R
CHARLIE	5	19	17	20	23	22	24	83
WHISKEY	11	1	10	18	20	20	24	54
WHISKEE	11	1	10	18	20	0	18	78
TANGOLF	7	13	19	8	23	20	14	62
VVICTOR	0	13	20	17	21	3	4	84
VICTORR	13	20	17	21	3	0	4	82
MESSAGE	18	14	0	8	6	24	8	70
NOTHING	1	5	14	1	5	19	7	72

7 Letters + 1 Symbol								
Text	AP(n) – AP(n-1) or AP(1) – AP(7)							R
IN GREEN	5	19	11	13	0	9	21	84

7 Letters + 2 Symbols								
Text	AP(n) – AP(n-1) or AP(1) – AP(7)							R
G IN JADE	2	5	22	17	3	1	2	59
G IN ROSE	2	5	4	23	4	12	2	63
I GIVE UP	24	2	13	9	16	21	19	58

8 Letters + 1 Symbol								
Text	AP(n) – AP(n-1) or AP(1) – AP(8)							R
ECHO ECHO	24	5	7	16	24	5	7	16	84
CHARLIE C **	5	19	17	20	23	22	24	0	43
C CHARLIE **	0	5	19	17	20	23	22	24	41
IN YELLOW	5	11	6	7	0	3	8	12	41
END QUOTE	9	16	13	4	20	5	11	0	66

8 Letters + 2 Symbols								
Text	AP(n) – AP(n-1) or AP(1) – AP(8)							R
YES BUT NO	6	14	9	19	25	20	1	10	89

9 Letters + 1 Symbol										
Text	AP(n) − AP(n-1) or AP(1) − AP(9)									R
ALPHA PAPA **	11	4	18	19	15	11	15	11	0	56
PAPA ALPHA **	11	15	11	0	11	4	18	19	15	86
PAPHA ALPA	11	15	18	19	0	11	4	11	15	69
DELTA NEXT	1	7	8	7	13	17	19	22	10	47
LIME BRAVO	23	4	18	23	16	9	21	19	23	78
BLUE BRAVO	10	9	10	23	16	9	21	19	13	47
THREE ONES	14	10	13	0	10	25	17	14	1	88
ONE ELEVEN	25	17	0	7	19	17	9	9	1	86
IN MAGENTA	5	25	14	6	24	9	6	7	8	51

9 Letters + 2 Symbols										
Text	AP(n) − AP(n-1) or AP(1) − AP(9)									R
BLUE IN RED	10	9	10	4	5	4	13	25	24	42
ONE ONE ONE	25	17	10	25	17	10	25	17	10	58
BLACK TEXT?	10	15	2	8	9	11	19	22	8	46

10 Letters + 1 Symbol											
Text	AP(n) − AP(n-1) or AP(1) − AP(10)										R
TWO BUTTONS	3	18	13	19	25	0	21	25	5	1	79
SIX BUTTONS	16	15	4	19	25	0	21	25	5	0	71
YELLOW TEXT	6	7	0	3	8	23	11	19	22	5	70

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
White	Do nothing
Red	Subtract the first digit
Orange	Replace the second digit with the first
Yellow	Add the second digit
Lime	Subtract the higher digit
Green	Subtract the sum of the digits
Jade	Subtract twice the first digit

Text Colour	Modification
Grey	Swap the digits
Cyan	Subtract the second digit
Azure	Replace the first digit with the second
Blue	Add the first digit
Violet	Subtract the lower digit
Magenta	Subtract the difference between the digits
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.
   - If the colour of the button is written on itself, add 70.
   - Otherwise, if the complementary colour of the button is written on it, add 35.
   - Otherwise, if any colour or the word **COLOUR** is written on the button, add 5.
3. Add 60 for each unencrypted display text that is the same as the text written on the button.
4. Add 15 for each display text that is the same colour as the button.
5. If the button is not grey, add 10 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, T_2, and T_3 for each button.

Position of button	T_1	T_2	T_3
TL	Display text 6	Display text 1	Display text 1
TM	Display text 7	Display text 3	Display text 1
TR	Display text 8	Display text 3	Display text 3
BL	Display text 6	Display text 5	Display text 3
BM	Display text 7	Display text 5	Display text 5
BR	Display text 8	Display text 1	Display text 5

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

$$F = 3I + 2(T_1 + T_2 + T_3)$$

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

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.

	#	'	"	?	–	*	~	!
#	#	'	"	?	–	*	~	!
'	'	'	?	–	*	~	!	#
"	"	?	"	*	~	!	#	'
?	?	–	*	?	!	#	'	"
–	–	*	~	!	–	'	"	?
*	*	~	!	#	'	*	?	–
~	~	!	#	'	"	?	~	*
!	!	#	'	"	?	–	*	!

Appendix: Button and Display Text Colours