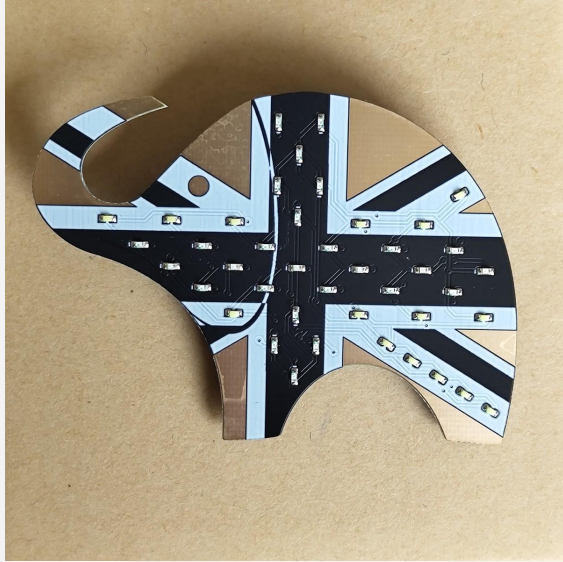


Electric Elephants



<https://intrbiz.com/electronics/led-pcb-art/>

Chris Ellis

Where It Started



Where It Started

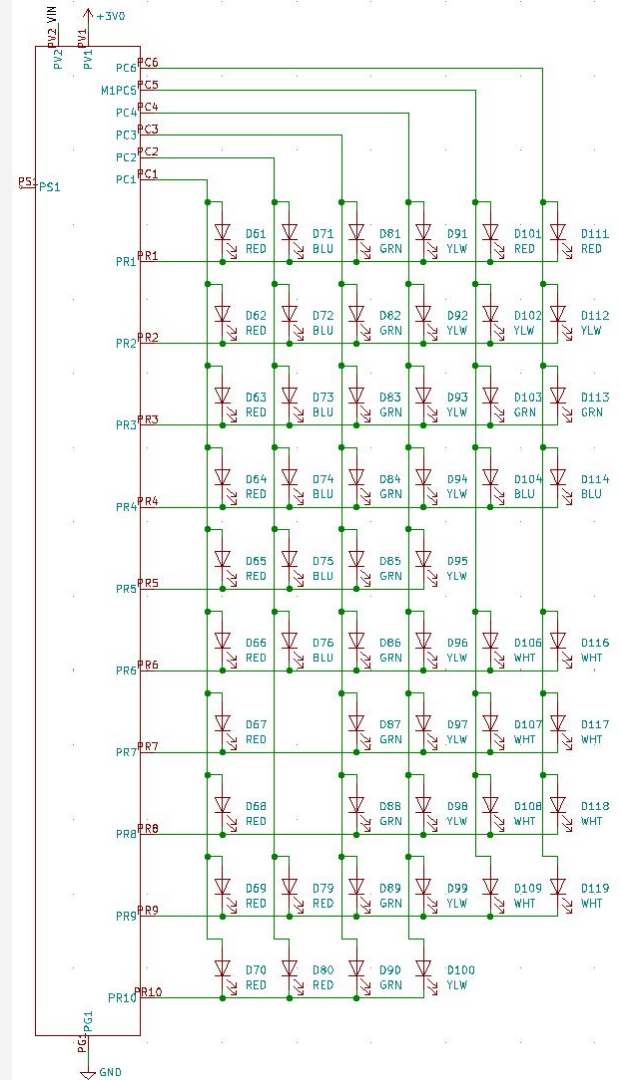
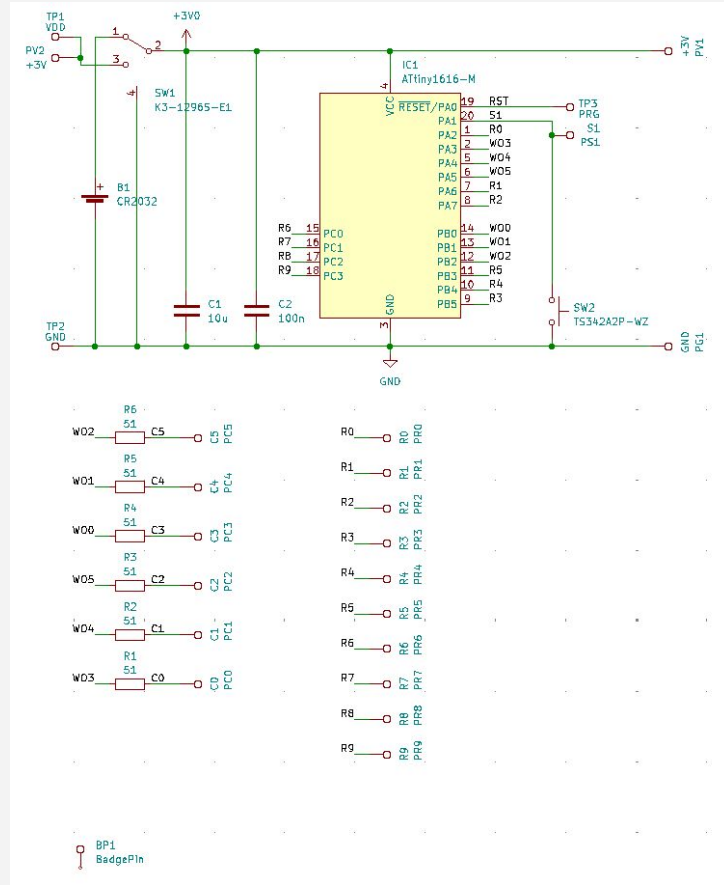


Design Process

ATTiny1616

- 2KiB RAM
- 16KiB Flash
- 1 MHz
- 6 PWMs

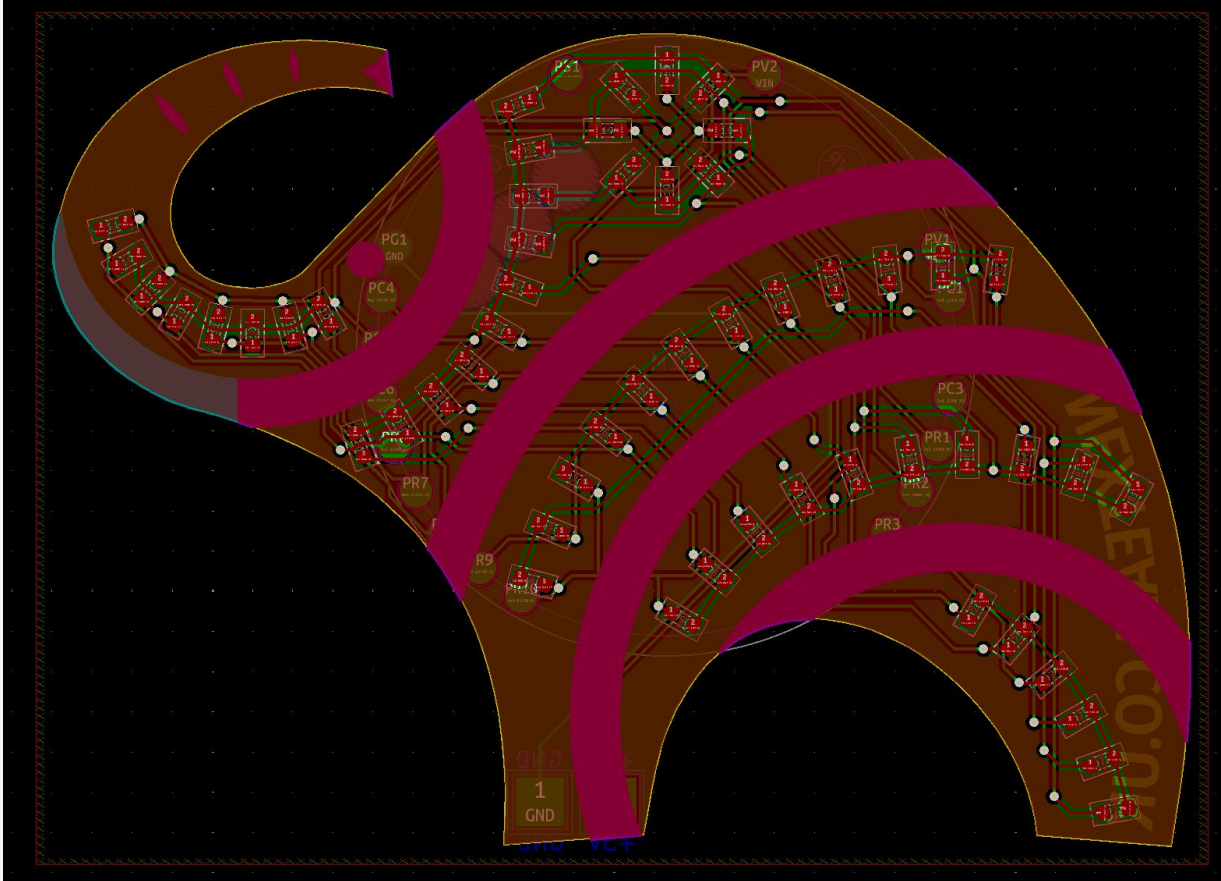
LEDs in a 6 x 10 matrix



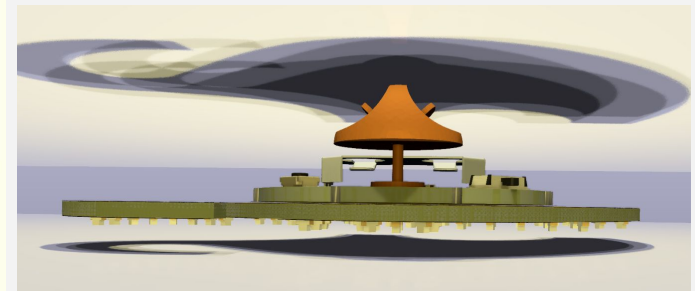
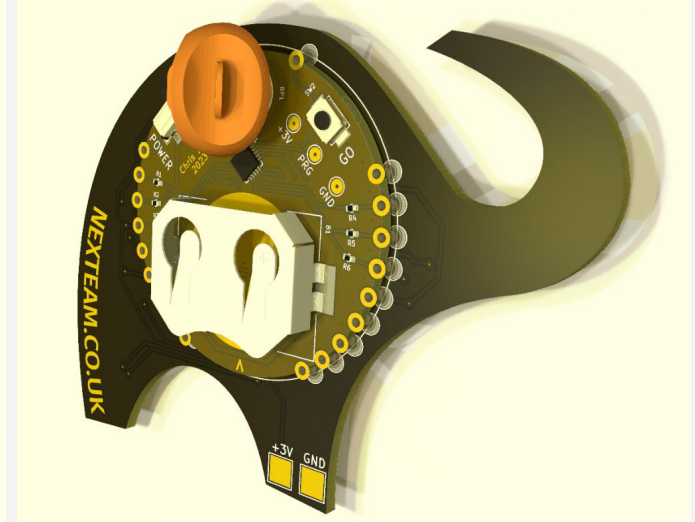
Design Process



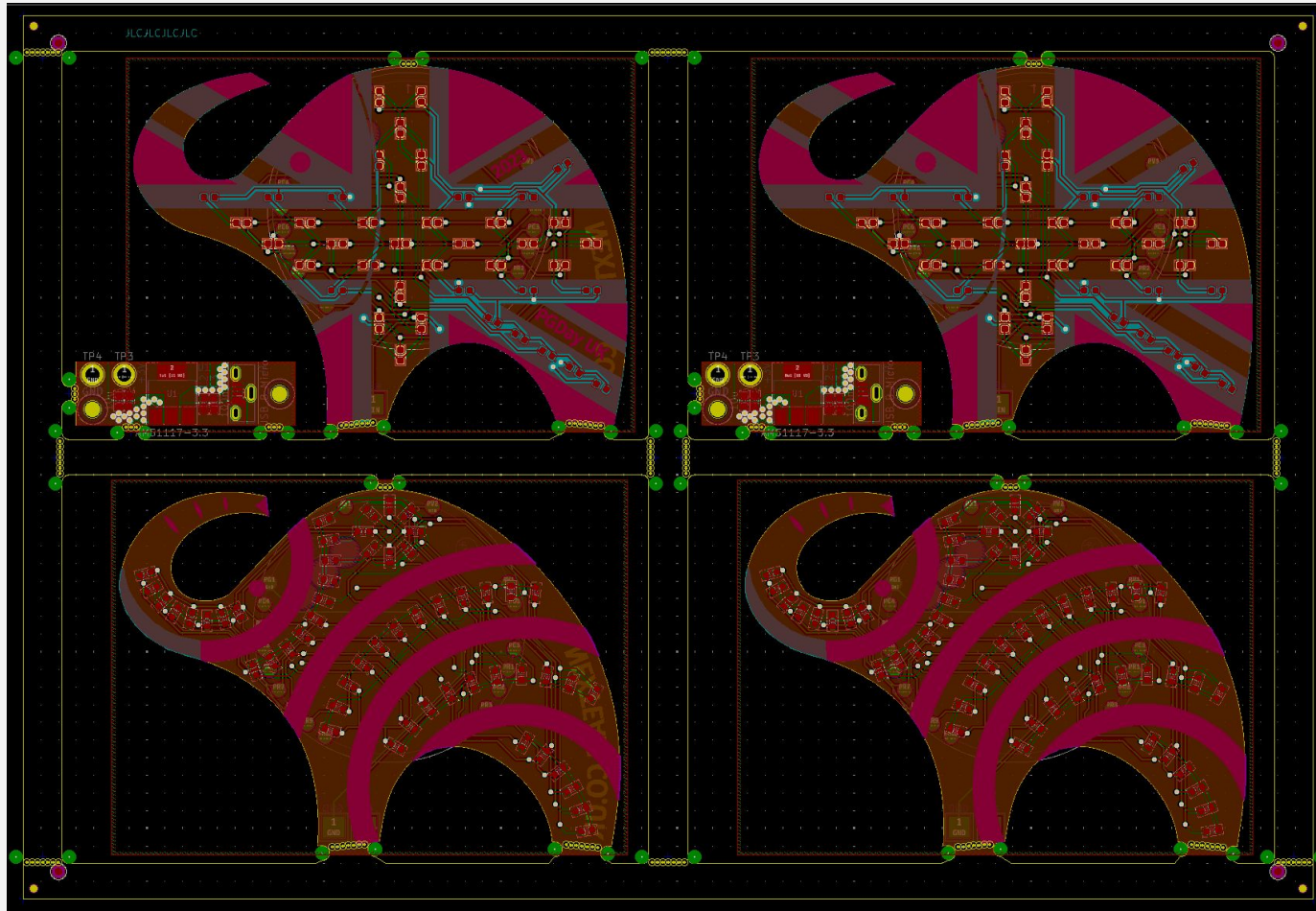
Design Process



Design Process

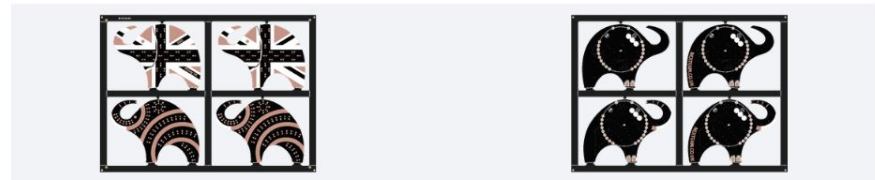


Making



Making

-  Standard PCB/PCBA
-  Advanced PCB/PCBA
-  SMT-Stencil
-  3D/CNC



[← Back to Upload File](#)

Detected 2 layer board of 125x172.5mm(4.92x6.79 inches).

[Gerber Viewer](#)

Base Material FR-4 Flex Aluminum Copper Core Rogers PTFE Teflon

Layers 1 2 4 High Precision PCB 6 8 10 12 14 16 18 20

Dimensions *

PCB Qty

Product Type Industrial/Consumer electronics Aerospace Medical

PCB Specifications

Different Design 1 2 3 4

Delivery Format Single PCB Panel by Customer Panel by JLPCB

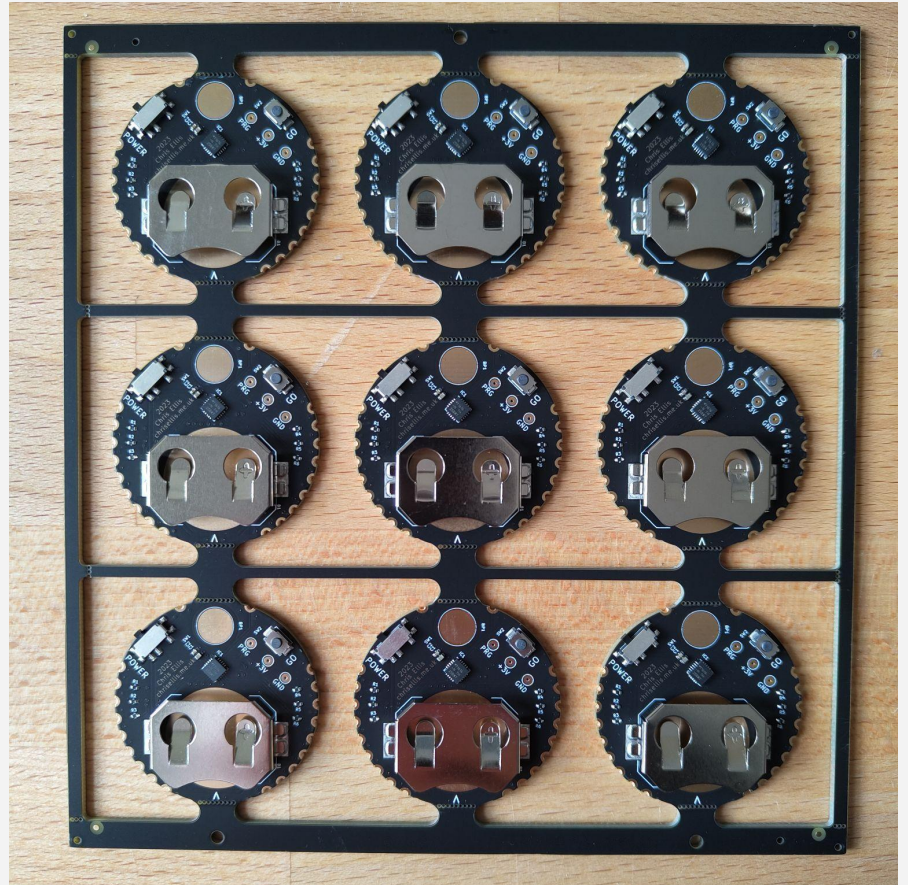
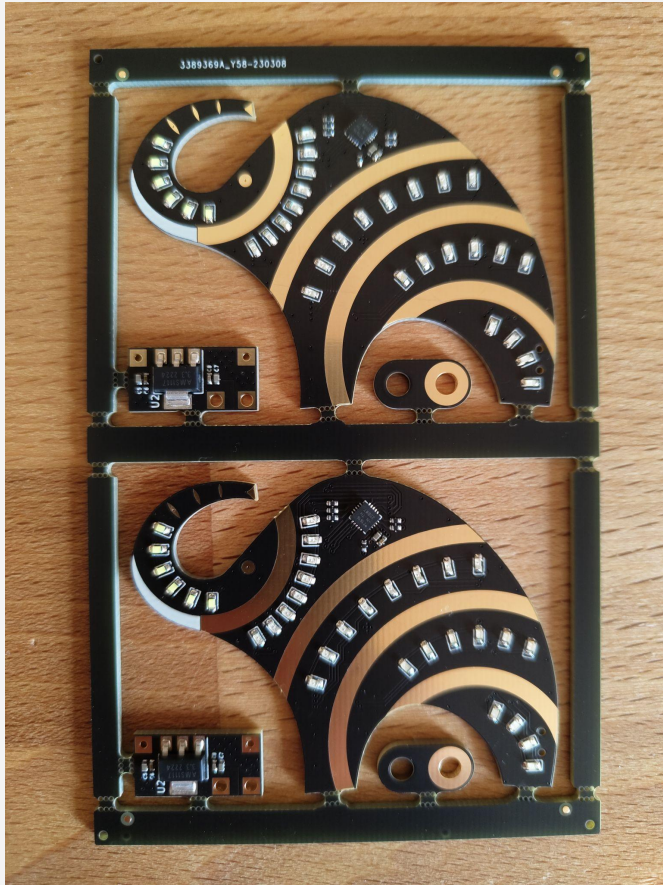
PCB Thickness 0.4 0.6 0.8 1.0 1.2 1.6 2.0

PCB Color Green Purple Red Yellow Blue White Black

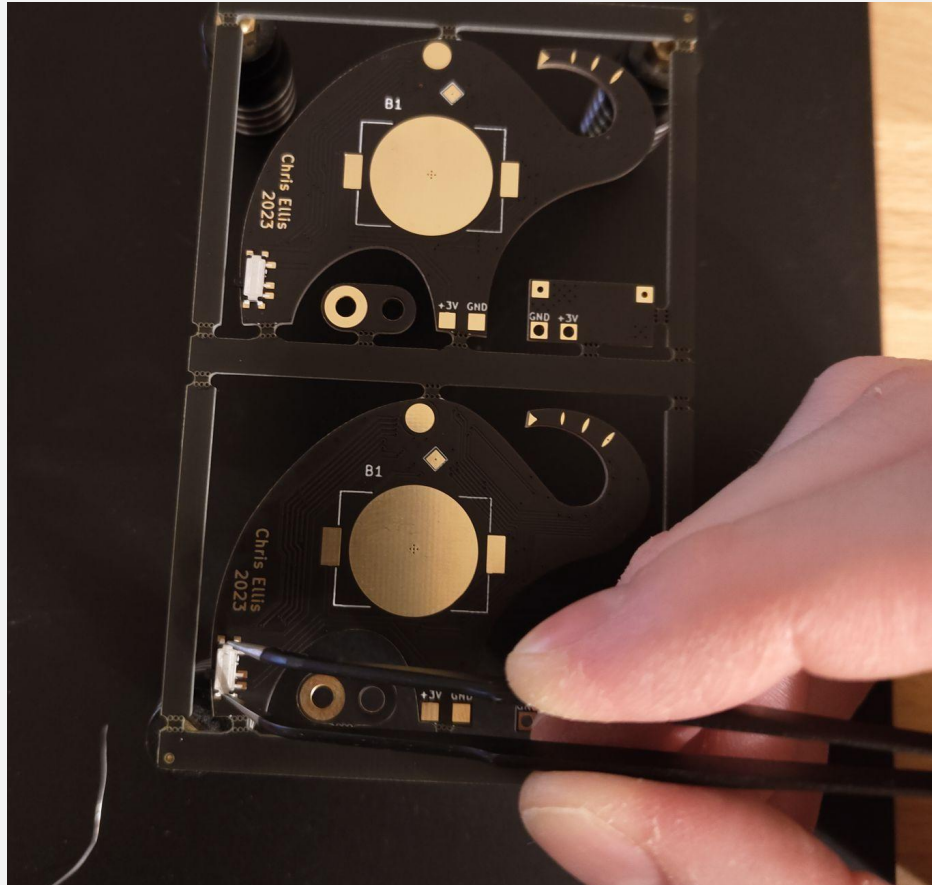
Silkscreen White

Surface Finish HASL(with lead) LeadFree HASL ENIG

Making



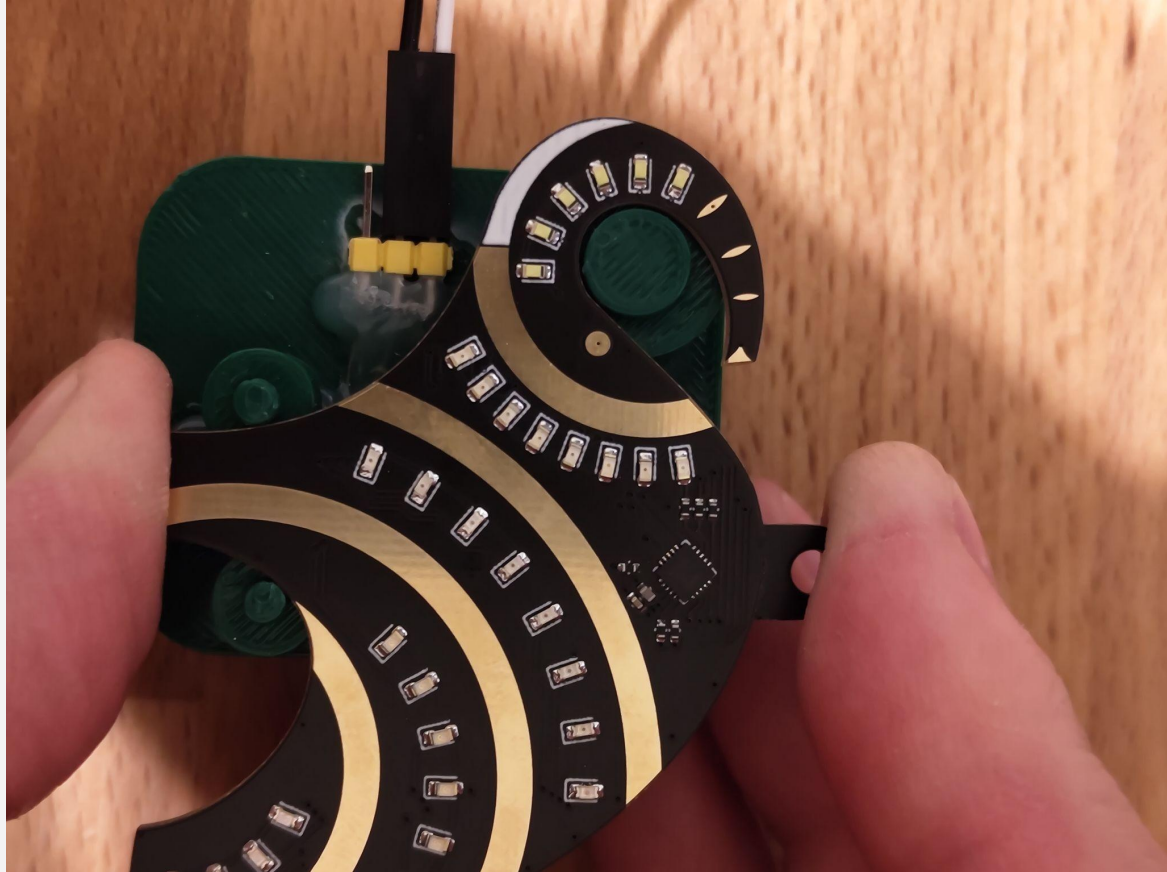
Making



Making



Programming



Software

```
/* subprogram sub_flash_all_yellow: 24 bytes */  
const uint8_t SUBPROGRAM_21_OPS[] = {  
    0x9d, 0x05, 0xd, 0x15,  
    0x1d, 0x25, 0x2d, 0x35,  
    0x3d, 0x45, 0x4d, 0x30,  
    0x99, 0x01, 0x09, 0x11,  
    0x19, 0x21, 0x29, 0x31,  
    0x39, 0x41, 0x49, 0x30  
};
```

Software

Command Byte										Next Bytes	Description	Decode Tree
7	6	5	4	3	2	1	0	OP				
REGISTER	0	0	0	0	0	0	0	PUSH	0 or 1	Push to the argument stack	0 -> 0	
REGISTER	0	0	1	0	0	0	0	POP	0	Pop from the argument stack	0 -> 1	
REGISTER	0	1	0	0	0	0	0	SET	1	Set \$REGISTER to the value of the next byte	0 -> 2	
REGISTER	0	1	1	0	0	0	0	INC	1	Increment \$REGISTER with the next byte	0 -> 3	
REGISTER	1	0	0	0	0	0	0	WAIT	0	Wait for value of \$REGISTER	0 -> 4	
TARGET	1	0	1	0	0	0	0	CALL	1	Call a subprogram, program or built-in	0 -> 5	
REGISTER	1	1	0	0	0	0	0	JUMP	2 or 3	Jump	0 -> 6	
LOOP ID	0	1	1	1	0	0	0	LOOP START	1	Start loop \$ID, next byte is iterations	0 -> 7 -> 0	
LOOP ID	1	1	1	1	0	0	0	LOOP END	0	End loop	0 -> 7 -> 1	
Line No					0	0	1	OFF	0 or 1	Turn matrix line off	1 -> 0	
Line No					1	0	1	ON	0 or 1	Turn matrix line on	1 -> 1	
Line No					0	1	0	PWM_SET	1 or 2	Set matrix line PWM to next byte	2 -> 0	
Line No					1	1	0	PWM_FADE	1 or 2	Fade matrix line PWM to next byte	2 -> 1	
LED No						1	1	FLASH	0 or 1	Flash an LED on and off	3	

Software

```
START_PROGRAM(PROG_UP_DOWN_WHITE_LEDS)
  SET_PERIOD(PERIOD_RANDOM)
  SET_PERIOD(PERIOD_GY)
  START_LOOP(LOOP_A, LOOPS_GX)
    FLASH_LED(WHITE_6)
    FLASH_LED(WHITE_5)
    FLASH_LED(WHITE_4)
    FLASH_LED(WHITE_3)
    FLASH_LED(WHITE_2)
    FLASH_LED(WHITE_1)
    FLASH_LED(WHITE_2)
    FLASH_LED(WHITE_3)
    FLASH_LED(WHITE_4)
    FLASH_LED(WHITE_5)
  END_LOOP(LOOP_A)
  FLASH_LED(WHITE_6)
END_PROGRAM
```


Software

```
subprogram SUB_TRUNK_SPEED_UP
  set $PERIOD 11
  flash WHITE_1
  set $PERIOD 10
  flash WHITE_2
  set $PERIOD 9
  flash WHITE_3
  set $PERIOD 8
  flash WHITE_4
  set $PERIOD 7
  flash WHITE_5
  set $PERIOD 6
  flash WHITE_6
  set $PERIOD 5
  flash WHITE_7
  set $PERIOD 4
  flash WHITE_8
```

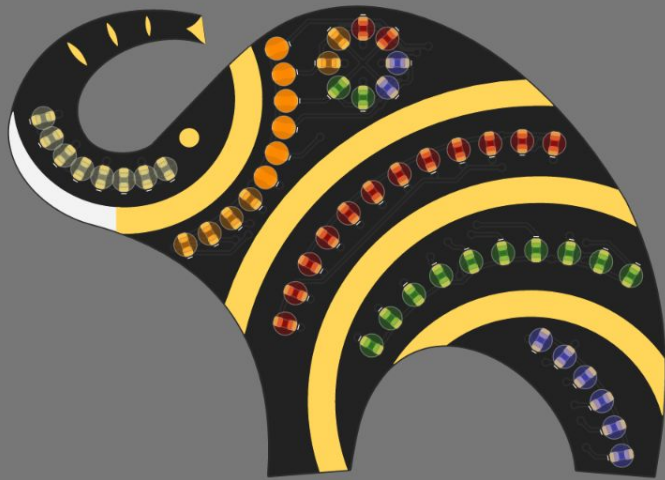
```
subprogram SUB_FILL_WHITE_LEDS
  on COL_WHITE_A
  on COL_WHITE_B
  on ROW_WHITE_8
  wait $PERIOD
  on ROW_WHITE_6
  wait $PERIOD
  on ROW_WHITE_4
  wait $PERIOD
  on ROW_WHITE_2
  wait $PERIOD
  wait $PERIOD
  off COL_WHITE_A
  off COL_WHITE_B
  off ROW_WHITE_2
  off ROW_WHITE_4
  off ROW_WHITE_6
  off ROW_WHITE_8
```

Pattern Editor

LED Pattern Editor

Generate Header File

Elephant V2



```
Flash WHITE_5
Flash WHITE_6
Flash WHITE_7
Flash WHITE_8
```

```
subprogram SUB_DOWN_BLUE_LEDS
Flash BLUE_1
Flash BLUE_2
Flash BLUE_3
Flash BLUE_4
Flash BLUE_5
Flash BLUE_6
```

```
subprogram SUB_FILL_WHITE_LEDS
on COL_WHITE_A
on COL_WHITE_B
on ROW_WHITE_8
wait $PERIOD
on ROW_WHITE_6
wait $PERIOD
on ROW_WHITE_4
wait $PERIOD
on ROW_WHITE_2
wait $PERIOD
wait $PERIOD
off COL_WHITE_A
off COL_WHITE_B
off ROW_WHITE_2
off ROW_WHITE_4
off ROW_WHITE_6
off ROW_WHITE_8
```

```
subprogram SUB_FLASH_ALL_YELLOW
on COL_YELLOW
on ROW_YELLOW_1
on ROW_YELLOW_2
on ROW_YELLOW_3
on ROW_YELLOW_4
on ROW_YELLOW_5
on ROW_YELLOW_6
on ROW_YELLOW_7
on ROW_YELLOW_8
*** END VCL I AM 6
```

LED	Col	Row	Group
D0	C0	R0	Body Top
D1	C0	R1	Body Top
D2	C0	R2	Body Top
D3	C0	R3	Body Top
D4	C0	R4	Body Top
D5	C0	R5	Body Top
D6	C0	R6	Body Top
D7	C0	R7	Body Top
D8	C0	R8	Body Top
D9	C0	R9	Body Top
D10	C1	R0	Leg
D11	C1	R1	Leg
D12	C1	R2	Leg
D13	C1	R3	Leg
D14	C1	R4	Leg
D15	C1	R5	Leg
D18	C1	R8	Body Top
D19	C1	R9	Body Top
D20	C2	R0	Body Mid
D21	C2	R1	Body Mid
D22	C2	R2	Body Mid
D23	C2	R3	Body Mid
D24	C2	R4	Body Mid
D25	C2	R5	Body Mid
D26	C2	R6	Body Mid
D27	C2	R7	Body Mid
D28	C2	R8	Body Mid
D29	C2	R9	Body Mid
D30	C3	R0	Ear
D31	C3	R1	Ear
D32	C3	R2	Ear
D33	C3	R3	Ear
D34	C3	R4	Ear
D35	C3	R5	Ear
D36	C3	R6	Ear
D37	C3	R7	Ear
D38	C3	R8	Ear

Subprogram: sub_fill_yellow_up_leds

Run Halt Step Load Simulate

Registers		Loops		Counter		Stack						0	
Level	0x08	LC	0x00	A	0x00	13	0x00	0x00	0x00	0x00	0x00	0x00	0x00
Period	0x04	LD	0x00	B	0x00	Op	0x00	0x00	0x00	0x00	0x00	0x00	0x00
LA	0x00	LE	0x00	C	0x00	0x30	0x00	0x00	0x00	0x00	0x00	0x00	0x00
LB	0x00	LF	0x00	D	0x00		0x00	0x00	0x00	0x00	0x00	0x00	0x00

[INFO] Loaded program: sub_fill_yellow_up_leds, ready to single step.

[INFO] Assembly succeeded: 57 subprograms, 23 programs

[INFO] Loaded program: sub_fill_yellow_up_leds, ready to single step.

[INFO] Assembly succeeded: 57 subprograms, 23 programs

[INFO] Started program: sub_fill_yellow_up_leds

[INFO] Assembly succeeded: 57 subprograms, 23 programs

[INFO] Simulating device: programs will execute randomly, just as they would on the device.

[INFO] Assembly succeeded: 57 subprograms, 23 programs

[INFO] Assembly succeeded: 57 subprograms, 23 programs, using a total of 2022 bytes

[INFO] Assembly succeeded: 6 subprograms, 1 programs, using a total of 88 bytes

Thanks

<https://intrbiz.com/electronics/led-pcb-art/>