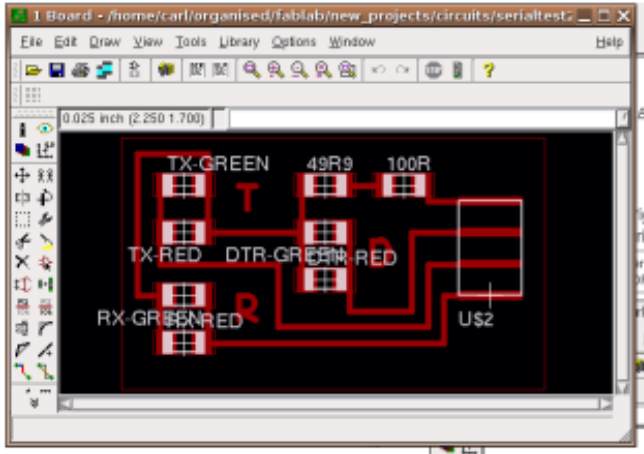
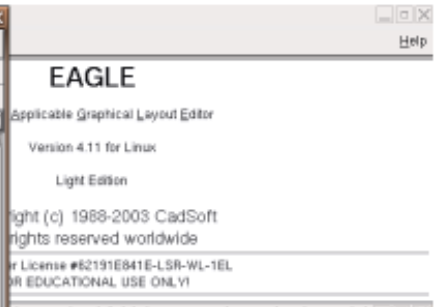
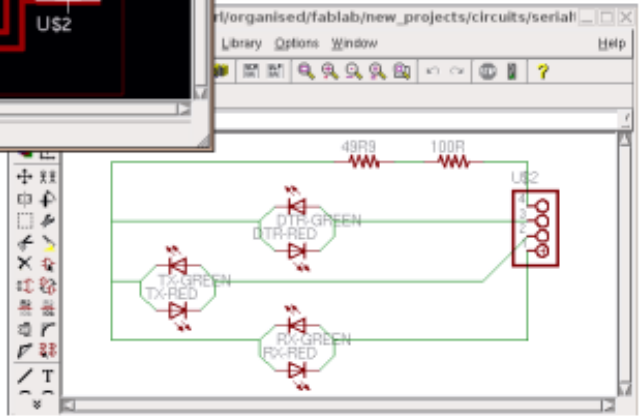
.fab

Turlif Vilbrandt
Carl Scheffler

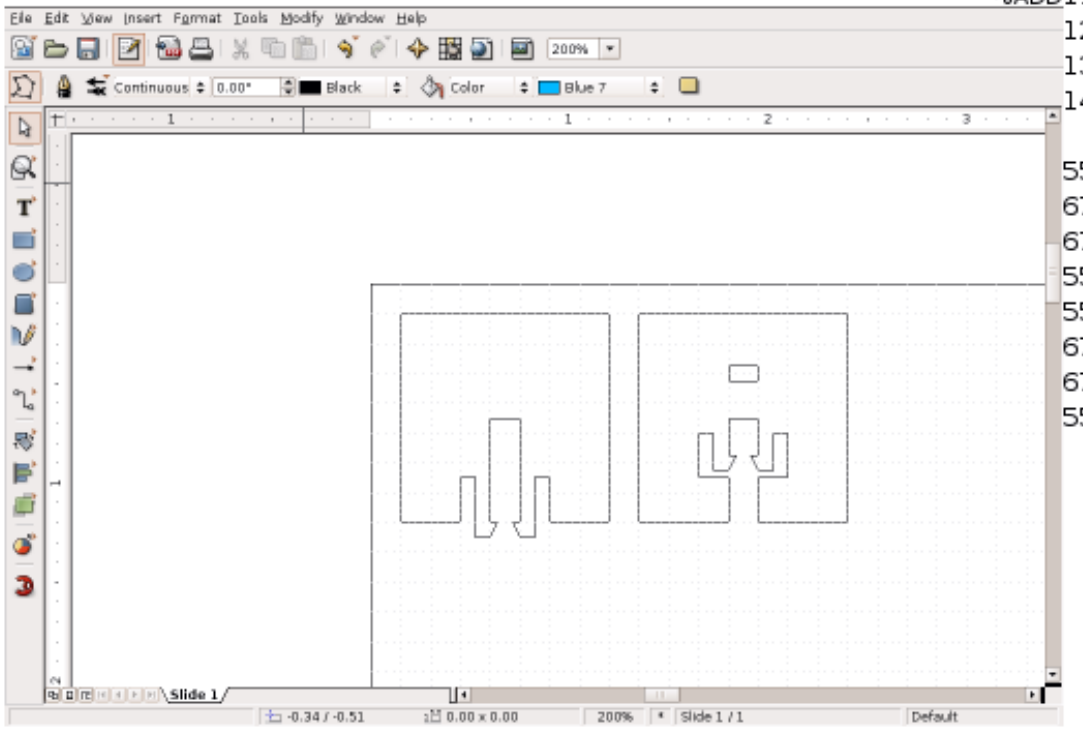
```
   ldi bitcnt, 9+sb; 1+8+sb                    bitloop:
   com txbyte; invert everything                  dec temp
   sec; set start bit                             brne bitl
   putchar0:                                   ret
      brcc putchar1; if carry set           ;
      sbi PORTB, txpin; send a '0'          ; print_delay
      rjmp putchar2; else                   ; delay betwee
   putchar1:                                ;
       cbi PORTB, txpin ; send a '1'        .equ delay = 20
       nop ; even out timing                print_delay:
   putchar2:                                   ldi temp1, d
       rcall bitdelay; one bit delay           charloop1:
       rcall bitdelay                             ldi temp,
       lsr txbyte; get next bit                   charloop0
       dec bitcnt; if not all bits sent              dec te
       brne putchar0; send next bit                  brne c
   ret;                                           dec temp1
;                                                 brne char
; bit delay                                    ret
;                                           ;
.equ b = 17 ; 9600 bps                      ; print
bitdelay:                                   ; prints a null
   ldi temp, b                              ;
```

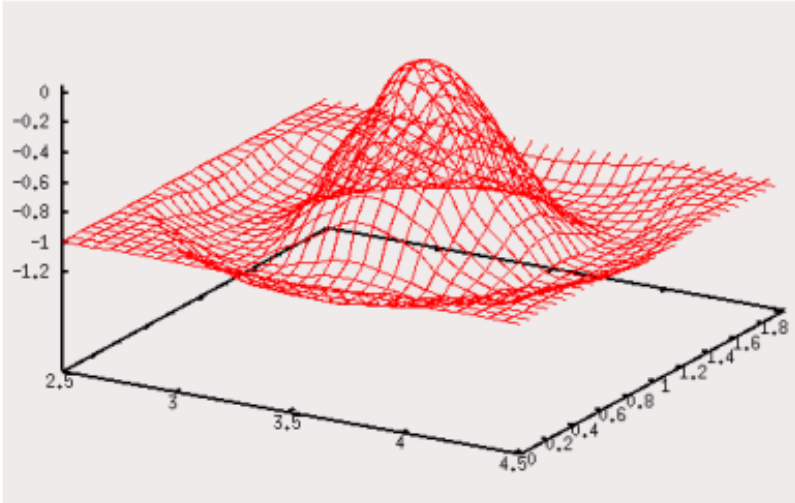






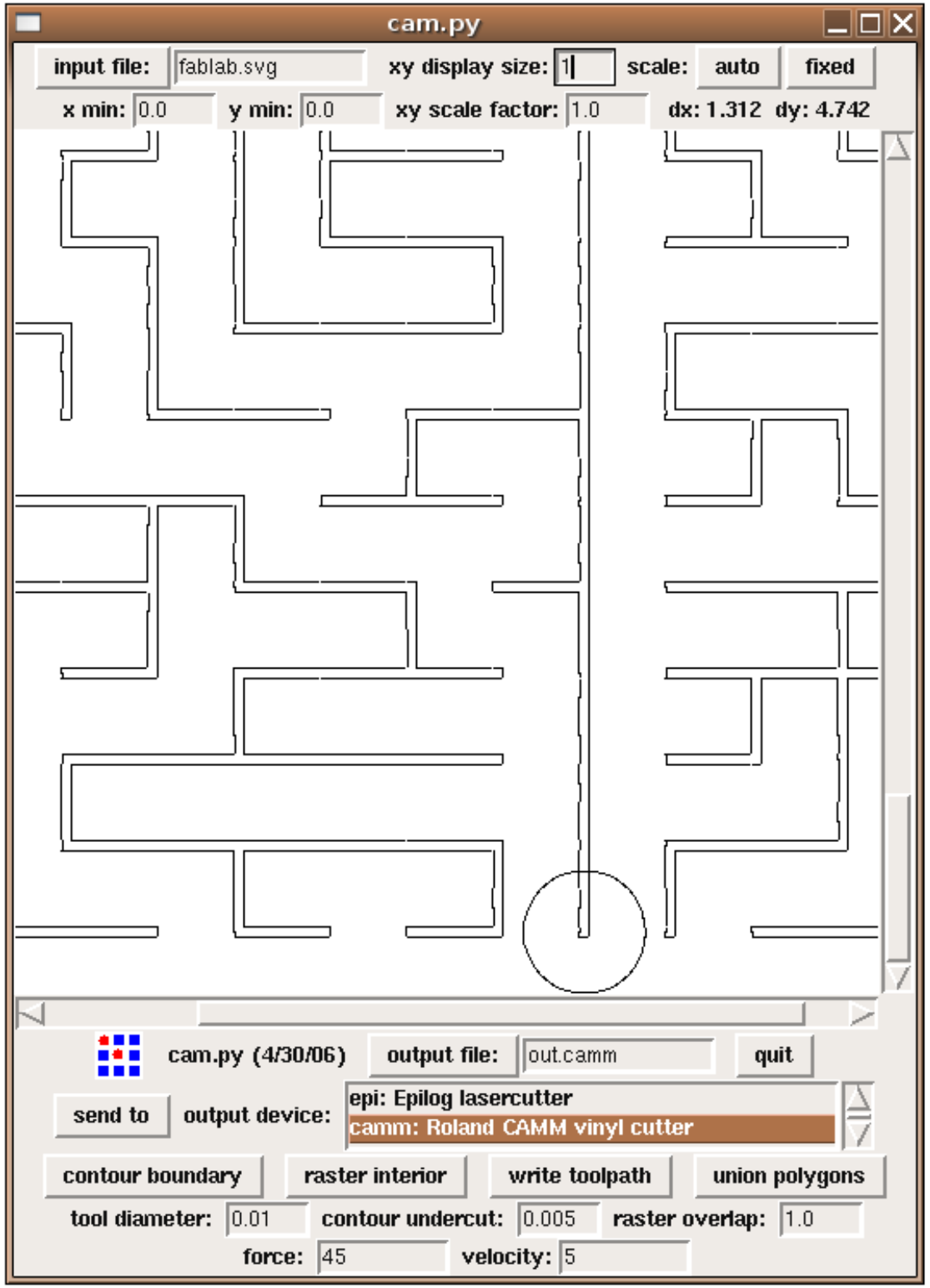

```
G75*
G70*
%OFA0B0*%
%FSLAX24Y24*%
%IPPOS*%
%LPD*%
%AMOC8*
5,1,8,0,0,1.08239X$1,22.5*
%
%ADD10R,0.0630X0.0710*%
%ADD11R,0.2000X0.0250*%
12C,0.0160*%
13C,0.0100*%
14C,0.0000*%

550Y001602D03*
670Y001602D03*
670Y003102D03*
550Y003102D03*
550Y005102D03*
670Y005102D03*
670Y006602D03*
550Y006602D03*
```

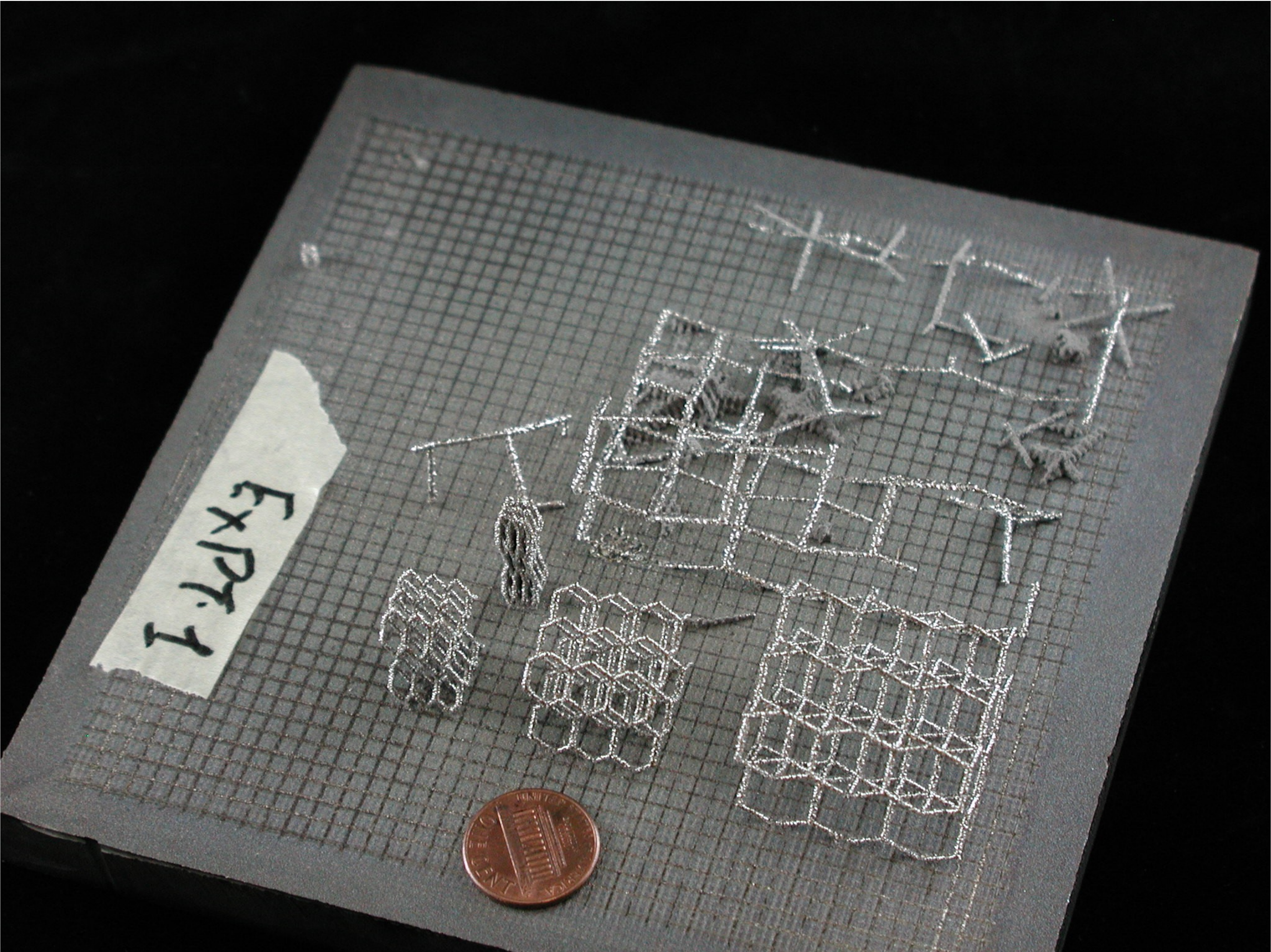


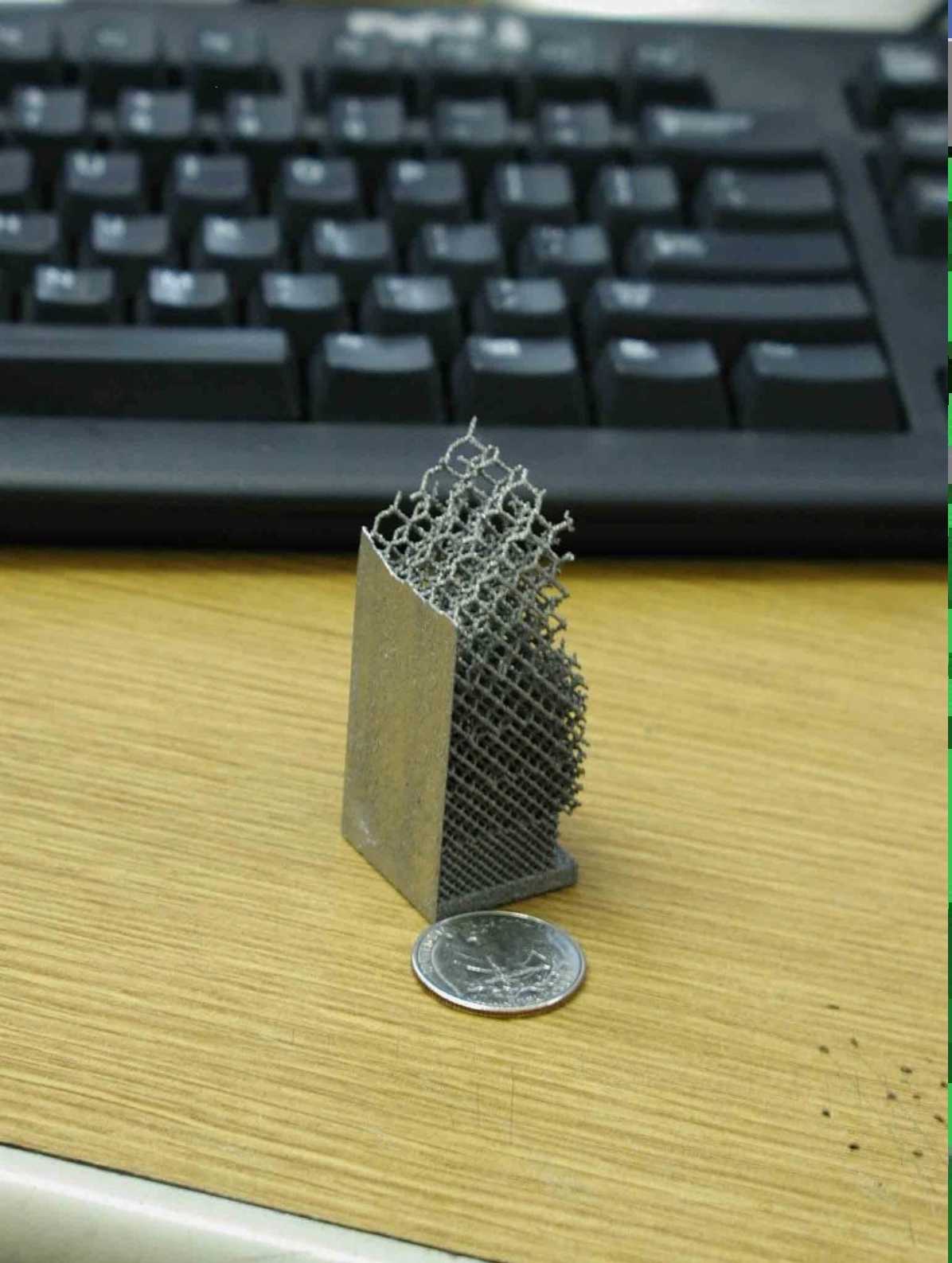


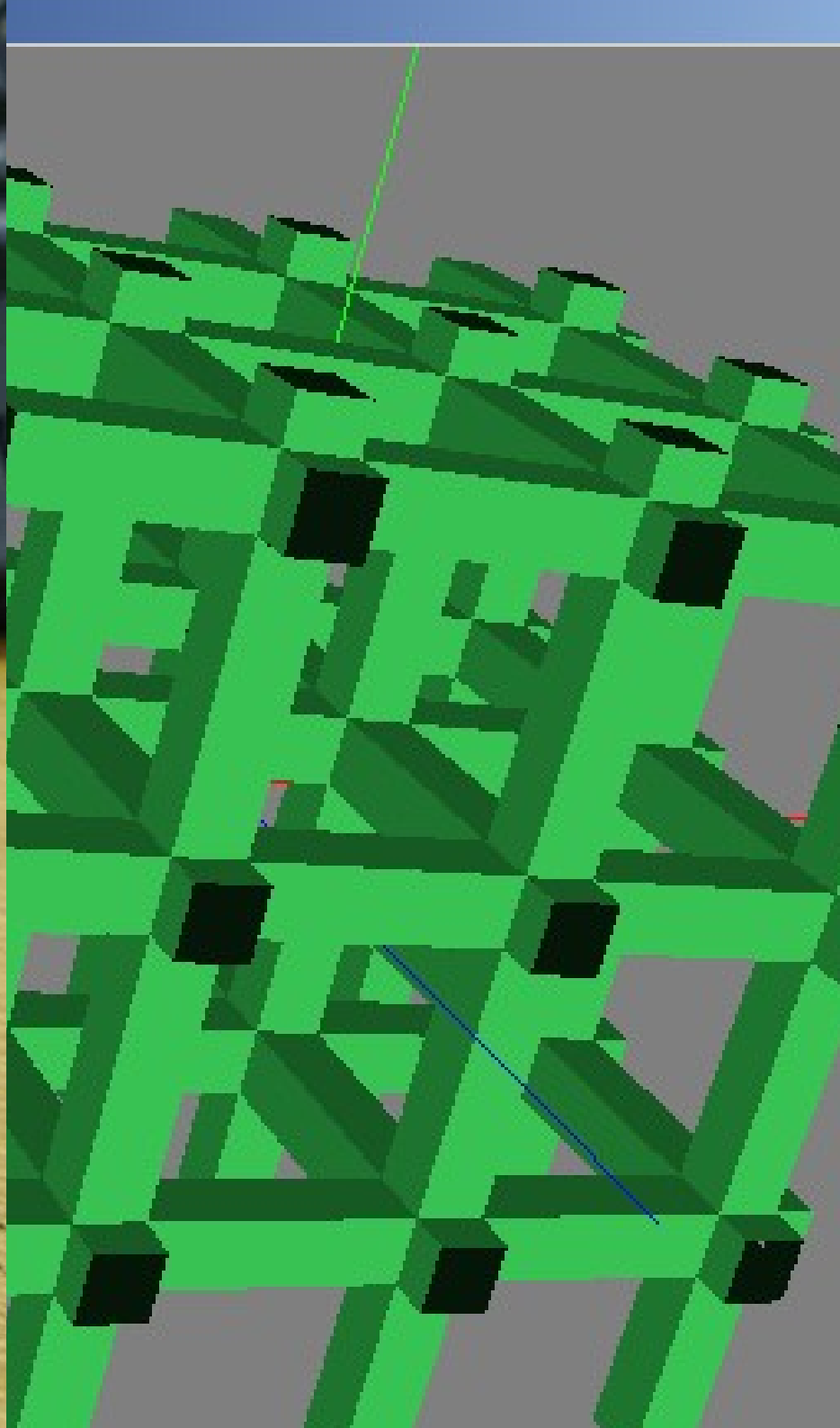
```
<svg width="57mm" height="29mm" viewBox="0 0
 <g style="stroke:rgb(0,0,0);fill:none">
  <polyline points="2667,2667 2667,0 0,0 0,26
                    1238,2667 1143,2667 1143,
                    1715,2857 1715,2095 1905,
  <polyline points="5714,2667 5714,0 3047,0 3
                    4000,1524 4000,2000 4190,
                    4571,1809 4476,1809 4571,
                    4571,2095 4571,2667 5714,
  <polyline points="4572,856 4191,856 4191,66
 </g>
</svg>
```

```
  0                  20
SECTION              7.392460789564439
  2                  30
HEADER               -0.0000504759219531
  9                  9
$ACADVER             $EXTMAX
  1                  10
AC1015               172.2789997812067
  9                  20
$ACADMAINTVER        67.3970677799874
 70                  30
     6               0.0000504702564439
  9                  9
$DWGCODEPAGE
  3
ANSI_1252
  9
$INSBASE
 10
0.0
 20
0.0
 30
0.0
  9
$EXTMIN
 10
123.9713963735132
```




EXPT. 1

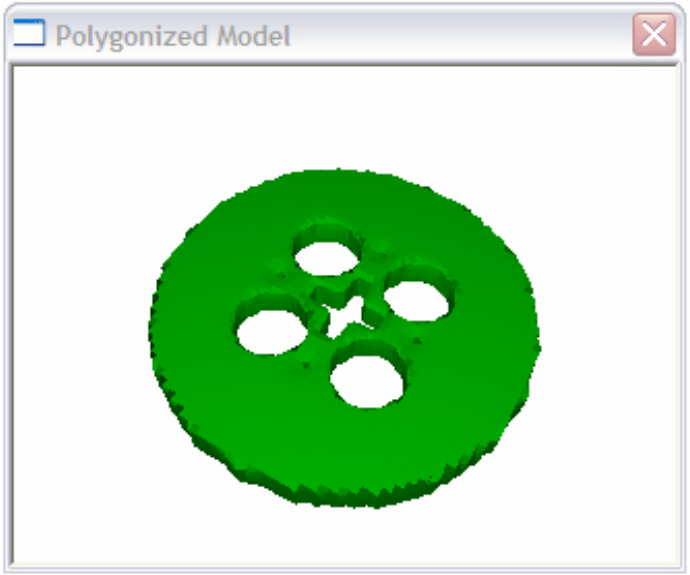




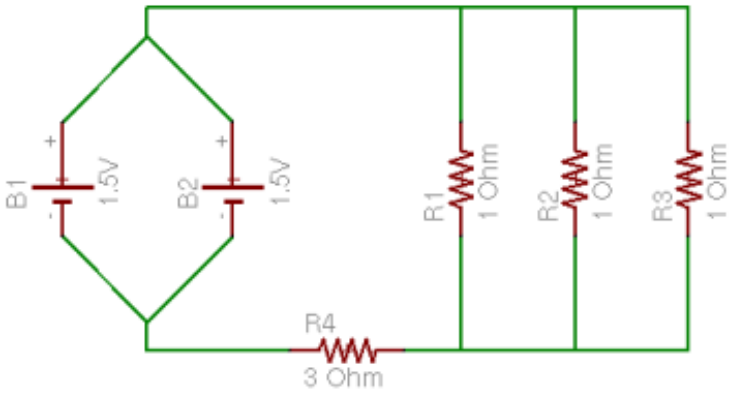
HyperFun for Windows - [amy3.hf]

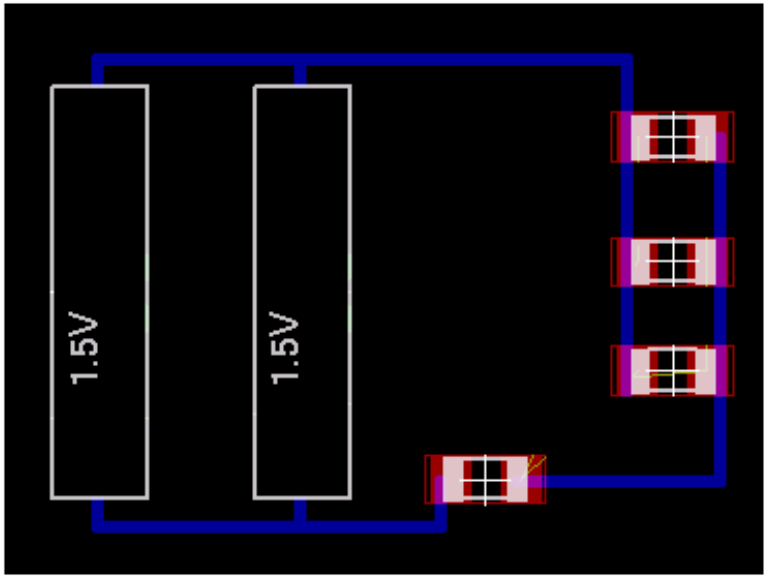
File Edit View Model Run Rendering Options Window Help

```
my_model(x[3], a[1]){

– DEF –
array origin[3];
array horizv[3], vertv[3];
array p[3];

– ORIENTATION –
origin = [0, 0, 0];
– orientation: [x,y,z] = [width, height, length]
–      thickness = height

– THERMODYNAMICAL SYSTEM CONSTRAINTS –
Q_in = total_solar_irradiance(solar_irradiance, collector_area) - Q_out;
m_dot_turbine = m_dot(water_conversion_rate, tube_dia) * recondensation_loss;

– DISK –
disk_thickness=1;
diskOD=sqrt(1/(m_dot_turbine*2*pi)) + vent_ratio;
diskID=sqrt(1/(m_dot_turbine*2*pi)) - vent_ratio;

 y=x[2];
lim1= disk_thickness/2-y;
lim2= disk_thickness/2+y;
cy1 = hfCylinderY(x,origin,diskOD);
disk = lim1 & lim2 & cy1;

– SHAFT –
shaft_width = 2.0;
shaft_thickness = 0.5;
shaft_length = 6.0;

horizv = [  -shaft_width/2.0, -shaft_length/2.0, -shaft_thickness/2.0 ];
vertv = [  -shaft_thickness/2.0, -shaft_length/2.0, -shaft_width/2.0 ];
shaft = hfBlock(x, horizv, 2.0, 6.0, 0.5) | hfBlock(x,vertv,0.5,6.0,2.0);

– VENT PORTS  –
– vent diameter should maximize the space between the diskID and shaft_width
– vent ports should also be max outlet possible
margin = 0.1;
```



Ready





$$R_i.V_2 - R_i.V_1 = R_i.I \times R_i.R, \quad i \in \{1,2,3,4\}$$

$$R_i.V_1 = R_j.V_1, \quad i \neq j, \quad i,j \in \{1,2,3\}$$

$$R_4.I_1 + \sum_{i=1}^{3} R_i.I_2 = 0$$

...

```
<!-- Batteries -->
<set B1 = <battery V=<volt 1.5/>/>/>
<set B2 = <battery V=<volt 1.5/>/>/>

<!-- Resistors -->
<defclass resistor1 _parent=resistor R=<ohm 1/>/>
<set R1 = <resistor1/>/>
<set R2 = <resistor1/>/>
<set R3 = <resistor1/>/>
<set R4 = <resistor R=<ohm 3/>/>/>

<!-- Wire nodes -->
<set node1 = <wire_node "#wires"=5/>/>
<set node2 = <wire_node "#wires"=4/>/>
<set node3 = <wire_node "#wires"=3/>/>

<!-- Define connections -->
<attach B1.nodes.0 node1.0/>
<attach B2.nodes.0 node1.1/>
<attach R1.nodes.0 node1.2/>
<attach R2.nodes.0 node1.3/>
<attach R3.nodes.0 node1.4/>

<attach R1.nodes.1 node2.0/>
<attach R2.nodes.1 node2.1/>
<attach R3.nodes.1 node2.2/>
<attach R4.nodes.0 node2.3/>

<attach R4.nodes.1 node3.0/>
<attach B1.nodes.1 node3.1/>
<attach B2.nodes.1 node3.2/>
```

```
<defclass resistor
   vi_equations=<vector
      "I1.<equals I2/>"
      "V2.<sub V1/>.<equals I1.<times R/>/>"/>
   bounding_box=<bounding_box
      center=<vec3 0 0 0/>
      size=<vec3 2 1 0.5/>/>
   nodes=<vector
      <node
         bounding_box=<bounding_box
            center=<vec3 0.25 0 0/>
            size=<vec3 0.5 1 0.5/>/>
         variables=<vector V1 I1/>/>
      <node
         bounding_box=<bounding_box
            center=<vec3 0.75 0 0/>
            size=<vec3 0.5 1 0.5/>/>
         variables=<vector V2 I2/>/>/>
   symbol=<polyline vertices=<vector
      <vec2 0 0/> <vec2 0.5 0/> <vec2 0.75 1/> <vec2 1.25 -1/> <vec2 1.75 1/> <vec2 2.25 -1/>
      <vec2 2.5 0/> <vec2 3 0/>/>
/>
```