
Chess Printing via METAFONT and T_EX

Zalman Rubinstein
University of Haifa

Every chess fan knows the pleasant difference between seeing an interesting chess position or a chess problem printed, and looking at the standard description of the pieces by means of an 8×8 coordinate system a1 to h8. To help bridge this gap we have written a METAFONT-T_EX program which enables one to print chess positions with ease, and to incorporate these positions with an arbitrary T_EX output.

The implementation is based on the idea of dispensing with the creation of a separate chess board but rather in integrating the chess board with the chess pieces, thereby multiplying their number by two. We shall denote the chess pieces by the letters **K**, **Q**, **R**, **B**, **N**, **P**, respectively. The dark square will be designated by the letter **D**. In this notation the king appears in four shapes: **K**, **KD**, **DK**, **DKD**, meaning the white king on a white square, the white king on a dark square, dark king on a white square and finally the dark king on a dark square. Similarly for the queen, rook, bishop, knight and pawn. It follows that the twenty four METAFONT designed pieces along with a single dark square suffice to assemble any chess position.

The shape of the chess pieces is based on simplicity rather than on artistic design at the present with the hope that improvements will be made at our METAFONT seminar this year.

It will suffice here to show the METAFONT code for the white pieces **K**, **Q**, **R**, **B**, **N**, **P** and the dark square **D** since all the other figures are easily deduced by METAFONT geometric and set theoretic operations such as **fill**, **unfill**, **draw**, **undraw**, **cullit**. The basic font was designed at 8pt size scaled 4000 (see illustration 1) on PC equipment. A smaller version scaled 3000 was also prepared (see illustration 2). The first approximation was set up on square paper with each box of size 16×16 squares.

Following is the METAFONT code for the basic figures mentioned earlier.

```
beginchar("KING",8pt#,8pt#,0pt#);
h#:=8pt#; define_pixels(h);
pickup pencircle scaled 0.2pt;
pair w[];
w1=(2.5,0.5); w2=(2.5,1); w3=(0.5,11);
w4=(8,14); w5=(15.5,11); w6=(13.5,1);
w7=(13.5,0.5); w9=(6.5,14);
w10=(9.5,14); w11=(8,3);
```

```
w8=whatever[w6,w5];
w12=whatever[w2,w3];
ypart w12 = ypart w11 = ypart w8;
for i=1 upto 12: z[i]=h/16*w[i]; endfor
draw z9--z10;
draw halfcircle scaled(3*h/16) shifted z4;
draw z12--z8; draw z2--z6;
draw z1--z7; draw z11--z4;
draw z5--z4;
draw z1--z2--z3--z4--z5--z6--z7--cycle;
endchar;
```

```
beginchar("QUEEN",8pt#,8pt#,0pt#);
w1=(3,0.5); w3=(2,1); w4=(1,13);
w5=(4,2); w6=(8,13); w7=(7.5,14.5);
w8=(8,15.5); w9=(8.5,14.5);
w10=(12,2); w11=(15,13); w12=(14,1);
w14=(13,0.5);
ypart w2 = ypart w13 = 1;
w1-w2 = whatever*(w5-w4);
w14-w13 = whatever*(w10-w11);
for i=1 upto 14: z[i]=h/16*w[i]; endfor
draw z1--z2--z3--z4--z5--z6--z7--z8--
z9--z6--z10--z11--z12--z13--z14--cycle;
endchar;
```

```
beginchar("ROOK",8pt#,8pt#,0pt#);
w1=(3,0.5); w2=(3,1.5); w3=(4,1.5);
w4=(4,14); w5=(3,14); w6=(3,15.5);
w7=(5,15.5); w8=(5,14.5); w9=(7,14.5);
w10=(7,15.5); w11=(9,15.5);
w12=(9,14.5); w13=(11,14.5);
w14=(11,15.5); w15=(13,15.5);
w16=(13,14); w17=(12,14); w18=(12,1.5);
w19=(13,1.5); w20=(13,0.5);
for i=1 upto 20: z[i]=h/16*w[i]; endfor
draw z1--z2--z3--z4--z5--z6--z7--z8--
z9--z10--z11--z12--z13--z14--z15--
z16--z17--z18--z19--z20--cycle;
draw z3--z18;
endchar;
```

```
beginchar("BISHOP",8pt#,8pt#,0pt#);
w1=(5.5,0.5); w2=(10.5,0.5);
w3=(10.5,1); w4=(11.5,1);
w5=(11.5,5); w6=(8,14); w7=(4.5,5);
w8=(4.5,1); w14=(8,14.5); w9=(5.5,1);
w10=(7.75,5); w11=(8.25,5);
w12=(8.25,9); w13=(7.75,9);
for i=1 upto 14: z[i]=h/16*w[i]; endfor
draw z10--z11--z12--z13--cycle;
```

```
draw z1--z2--z3--z4...z5{up}...z6 &
  z6...z7{down}...z8--z9--cycle;
draw z9--z3;
draw fullcircle scaled (1*h/16) shifted z14;
endchar;
```

```
beginchar("KNIGHT",8pt#,8pt#,Opt#);
w1=(4.5,0.5); w2=(14.5,0.5);
w3=(14.5,4); w4=(12,10.5); w5=(5,15.5);
w6=(4.8,14); w7=(3.5,9); w8=(2,3.5);
w9=(4.5,4.5); w10=(8.5,5); w11=(5.5,2);
for i=1 upto 11: z[i]=h/16*w[i]; endfor
draw z1--z2 & z2..z3{up}..z4{z5-z2}..z5 &
  z5{z2-z10}..z6{down}..z7{dir 250}..
  (z8+(1.5,3)){down}..(z8+(3,0)){right}..
  z9{dir 30}..{right}z10 & z10{dir 240}..
  z11{z1-z10}..{down}z1;
endchar;
```

```
beginchar("PAWN",8pt#,8pt#,Opt#);
w1=(4,0.5); w2=(12,0.5); w3=(10.5,3);
w4=(11,4); w5=(9,13); w6=(7,13);
w7=(5,4); w8=(5.5,3); w9=(8,15.5);
for i=1 upto 9: z[i]=h/16*w[i]; endfor
draw z1--z2--z3--z8--cycle;
draw z3{right}..z4{up}---z5 &
  z5{dir 45}..z9{left}..z6{dir 135} &
  z6---z7{down}..z8{right};
endchar;
```

```
beginchar("chssqr",8pt#,8pt#,Opt#);
pickup pencircle scaled 0.3pt;
k:=8;
for i=1 upto k:
  draw (w/(2*k)*(2*(k+1-i)-1),w)--
    (0,w/(2*k)*(2*i-1));
endfor
for i=1 upto k:
  draw (w,w/(2*k)*(2*(k+1-i)-1)--
    (w/(2*k)*(2*i-1),0);
endfor
picture W,Z; cullit; Z=currentpicture;
clearit;
fill (0,0)--(0,h)--(w,h)--(w,0)--cycle ;
W=currentpicture;
addto Z also W; cull Z keeping (2,2);
currentpicture:=Z;
endchar;
```

To accommodate the chessfont a short \TeX macro enables printing the initial position in chess as follows:

```
\beginchess
\chessline\DR\DND\DB\DQD\DK\DBD\DN\DRD
\chessline\DPD\DP\DPD\DP\DPD\DP\DPD\DP
\whitechessline
\darkchessline
\whitechessline
\darkchessline
\chessline\P\PD\P\PD\P\PD\P\PD
\chessline\RD\N\BD\Q\KD\B\ND\R
\endchess
```

In a general chess position the white squares can be denoted by $\backslash W$ or by $\backslash \square$ and the black squares by $\backslash D$. $\backslash whitechessline$ describes a horizontal chess line whose leftmost square is white, and similarly for $\backslash darkchessline$.

Obviously \TeX has the capability of producing a macro based on algebraic chess notation with only the pieces on board to be specified. We have not tried to do that.

It is to be noted that the program `chssqr` for the dark square is called in all pieces on dark squares as a subroutine. Because of the geometric design, in order to produce a new version of the twenty four chess pieces, it is only necessary to give the detailed programs of the six basic pieces with the rest following, as described earlier by transformations and set theoretical operations.

We shall conclude this note by listing the \TeX macro code and printing the illustrations mentioned earlier.

```
\font\chess = chesset scaled 4000
\font\chessm= chesset scaled 3000
\def\ifundefined#1{\expandafter
  \ifx\csname#1\endcsname\relax}
\def\beginchess{\relax\beginngroup
  \ifundefined{chess} \message{%
    Undefined font, replaced with cmtt10}
  \let\tt=\tentt
  \else \def\tt{\chess}\fi
  \tt\more }
\def\more{$$\vbox\bgroup
  \offinterlineskip\tabskip=0pt
  \hrule height 1pt
  \halign\bgroup
  \vrule width1pt
  ##&##&##&##&##&##&##&##&##&##&##
  \vrule width1pt\cr}
\def\endchess{\egroup
  \hrule height 1pt
```

This is the initial position in chess!

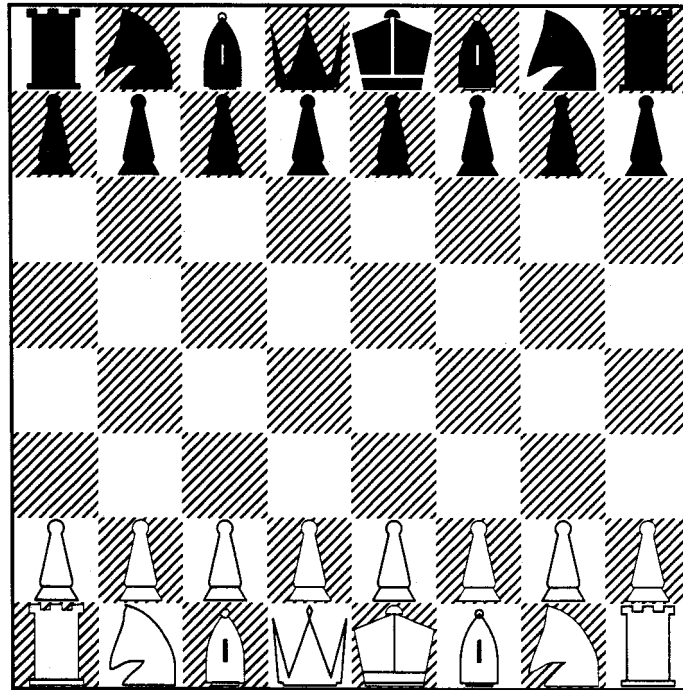


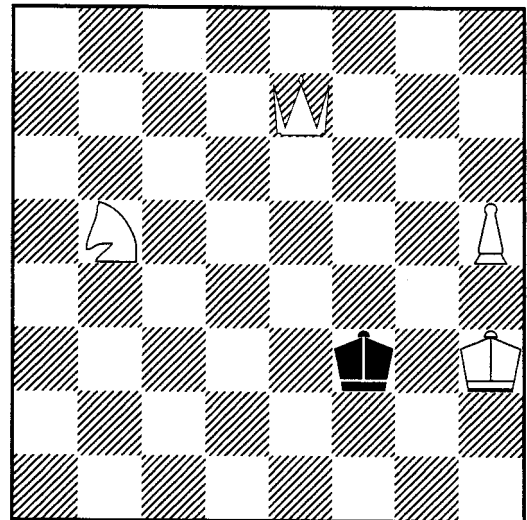
Illustration 1.

```

\egroup $$\endgroup\relax}
\def\whitechessline{&&D&&D&&D&&D&\cr}
\def\darkchessline{&D&&D&&D&&D&&D&\cr}
\def\chessline#1#2#3#4#5#6#7#8%
{&#1&#2&#3&#4&#5&#6&#7&#8&\cr}
\chardef\K="01 \chardef\DK="02
\chardef\KD="03 \chardef\DKD="04
\chardef\Q="05 \chardef\DQ="06
\chardef\QD="07 \chardef\DQD="08
\chardef\R="09 \chardef\DR="0A
\chardef\RD="0B \chardef\DRD="0C
\chardef\B="0D \chardef\DB="0E
\chardef\BD="0F \chardef\DBD="10
\chardef\N="11 \chardef\DN="12
\chardef\ND="13 \chardef\DND="14
\chardef\P="15 \chardef\DP="16
\chardef\PD="17 \chardef\DPD="18
\chardef\D="19 \chardef\W="00

```

◊ Professor Zalman Rubinstein
 University of Haifa
 Department of Mathematics and
 Computer Science
 Mount Carmel
 Haifa 31999 Israel
 Bitnet: rsma407@haifaum



Mate in three. Illustration 2.

My 64K chess computer solved it in twenty seconds.