

BDF Font translation

■ Initialization

```
In[1]:= Off[General::spell, General::spell1]
```

■ Function to expand hex numbers into bits

```
In[2]:= HexExpand = Function[{hex}, Sign[BitAnd[ToExpression["16^^" <> hex], #]] & /@  
Table[2^b, {b, 4 StringLength[hex] - 1, 0, -1}]]
```

```
Out[2]= Function[{hex}, (Sign[BitAnd[ToExpression[16^^<> hex], #1]] &) /@  
Table[2^b, {b, 4 StringLength[hex] - 1, 0, -1}]]
```

■ Function to compress bit array into words

```
In[3]:= WordBuild = Function[{bitarray}, Fold[2 #1 + #2 &, 0, bitarray]]
```

```
Out[3]= Function[{bitarray}, Fold[2 #1 + #2 &, 0, bitarray]]
```

Import and process

```
In[4]:= SetDirectory["../src"]
```

```
Out[4]= /home/cmaier/Hackerspaces/sinobit/zpix-pixel-font/src
```

```
In[5]:= FileNames[]
```

```
Out[5]= {Zpix.bdf, Zpix.vfb}
```

```
In[6]:= bdflines = StringSplit/@Import["Zpix.bdf", "Lines"];
```

```
In[7]:= Take[bdflines, 200]
Out[7]= { {STARTFONT, 2.1}, {FONT, -FontForge-Zpix-Normal-R-Normal--12-120-75-75-P-119-ISO10646-1}, {SIZE, 12, 75, 75}, {FONTPADDINGBOX, 19, 13, -5, -2}, {COMMENT, "Generated by, fontforge,, http://fontforge.sourceforge.net"}, {STARTPROPERTIES, 39}, {FOUNDRY, "FontForge"}, {FAMILY_NAME, "Zpix"}, {WEIGHT_NAME, "Normal"}, {SLANT, "R"}, {SETWIDTH_NAME, "Normal"}, {ADD_STYLE_NAME, ""}, {PIXEL_SIZE, 12}, {POINT_SIZE, 120}, {RESOLUTION_X, 75}, {RESOLUTION_Y, 75}, {SPACING, "P"}, {AVERAGE_WIDTH, 119}, {CHARSET_REGISTRY, "ISO10646"}, {CHARSET_ENCODING, "1"}, {FONTNAME_REGISTRY, ""}, {CHARSET_COLLECTIONS, "ASCII, ISOLatin1Encoding, ISO8859-5, JISX0208.1997, GB2312.1980, BIG5, ISO10646-1"}, {FONT_NAME, "Zpix"}, {FACE_NAME, "ZpixEX2"}, {FONT_VERSION, "2.0"}, {FONT_ASCENT, 10}, {FONT_DESCENT, 2}, {UNDERLINE_POSITION, -1}, {UNDERLINE_THICKNESS, 1}, {X_HEIGHT, 6}, {CAP_HEIGHT, 8}, {RAW_ASCENT, 796}, {RAW_DESCENT, 203}, {NORM_SPACE, 5}, {RELATIVE_WEIGHT, 40}, {RELATIVE_SETWIDTH, 50}, {SUPERSCRIPT_X, 0}, {SUPERSCRIPT_Y, 6}, {SUPERSCRIPT_SIZE, 6}, {SUBSCRIPT_X, 0}, {SUBSCRIPT_Y, 0}, {SUBSCRIPT_SIZE, 6}, {FIGURE_WIDTH, 6}, {AVG_LOWERCASE_WIDTH, 89}, {AVG_UPPERCASE_WIDTH, 95}, {ENDPROPERTIES}, {CHARS, 22043}, {STARTCHAR, space}, {ENCODING, 32}, {SWIDTH, 333, 0}, {DWIDTH, 4, 0}, {BBX, 1, 1, 13, -2}, {BITMAP}, {00}, {ENDCHAR}, {STARTCHAR, exclam}, {ENCODING, 33}, {SWIDTH, 300, 0}, {DWIDTH, 6, 0}, {BBX, 1, 9, 2, 0}, {BITMAP}, {80}, {80}, {80}, {80}, {80}, {80}, {80}, {00}, {80}, {80}, {ENDCHAR}, {STARTCHAR, quotedbl}, {ENCODING, 34}, {SWIDTH, 378, 0}, {DWIDTH, 6, 0}, {BBX, 3, 2, 1, 7}, {BITMAP}, {A0}, {A0}, {ENDCHAR}, {STARTCHAR, numbersign}, {ENCODING, 35}, {SWIDTH, 601, 0}, {DWIDTH, 6, 0}, {BBX, 5, 9, 0, 0}, {BITMAP}, {50}, {50}, {F8}, {50}, {50}, {50}, {F8}, {50}, {50}, {ENDCHAR}, {STARTCHAR, dollar}, {ENCODING, 36}, {SWIDTH, 628, 0}, {DWIDTH, 6, 0}, {BBX, 5, 11, 0, -1}, {BITMAP}, {20}, {70}, {A8}, {A0}, {60}, {20}, {30}, {28}, {A8}, {70}, {20}, {ENDCHAR}, {STARTCHAR, percent}, {ENCODING, 37}, {SWIDTH, 968, 0}, {DWIDTH, 6, 0}, {BBX, 6, 8, 0, 0}, {BITMAP}, {48}, {A8}, {B0}, {50}, {28}, {34}, {54}, {48}, {ENDCHAR}, {STARTCHAR, ampersand}, {ENCODING, 38}, {SWIDTH, 769, 0}, {DWIDTH, 6, 0}, {BBX, 5, 9, 0, 0}, {BITMAP}, {60}, {90}, {90}, {60}, {40}, {A8}, {A8}, {90}, {68}, {ENDCHAR}, {STARTCHAR, quotesingle}, {ENCODING, 39}, {SWIDTH, 203, 0}, {DWIDTH, 6, 0}, {BBX, 1, 2, 2, 7}, {BITMAP}, {80}, {80}, {ENDCHAR}, {STARTCHAR, parenleft}, {ENCODING, 40}, {SWIDTH, 359, 0}, {DWIDTH, 6, 0}, {BBX, 3, 11, 2, -1}, {BITMAP}, {20}, {40}, {40}, {80}, {80}, {80}, {80}, {40}, {40}, {20}, {ENDCHAR}, {STARTCHAR, parenright}, {ENCODING, 41}, {SWIDTH, 359, 0}, {DWIDTH, 6, 0}, {BBX, 3, 11, 1, -1}, {BITMAP}, {80}, {40}, {40}, {20}, {20}, {20}, {20}, {40}, {40}, {80}, {ENDCHAR}, {STARTCHAR, asterisk}, {ENCODING, 42}, {SWIDTH, 437, 0}, {DWIDTH, 6, 0}, {BBX, 5, 5, 0, 2}, {BITMAP}, {20}, {20}, {F8}, {20}}
```

■ Split font file in character descriptions

```
In[8]:= Transpose[Flatten[Position[bdflines, #]] & /@ {{"STARTCHAR", _}, {"ENDCHAR"}}];
Dimensions[charlines = Take[bdflines, #] & /@ %]

Out[9]= {22 043}
```

```
In[10]:= charlines[[1]]
Out[10]= {{STARTCHAR, space}, {ENCODING, 32}, {SWIDTH, 333, 0},
           {DWIDTH, 4, 0}, {BBX, 1, 1, 13, -2}, {BITMAP}, {00}, {ENDCHAR}}
```

```
In[11]:= charlines[[2]]
Out[11]= {{STARTCHAR, exclam}, {ENCODING, 33}, {SWIDTH, 300, 0},
           {DWIDTH, 6, 0}, {BBX, 1, 9, 2, 0}, {BITMAP}, {80}, {80},
           {80}, {80}, {80}, {80}, {00}, {80}, {80}, {ENDCHAR}}
```

■ Extract character codes

```
In[12]:= Dimensions[
charcodes = Last[Flatten[Select[#, First[#] == "ENCODING" &]]] & /@ charlines]
Out[12]= {22 043}
```

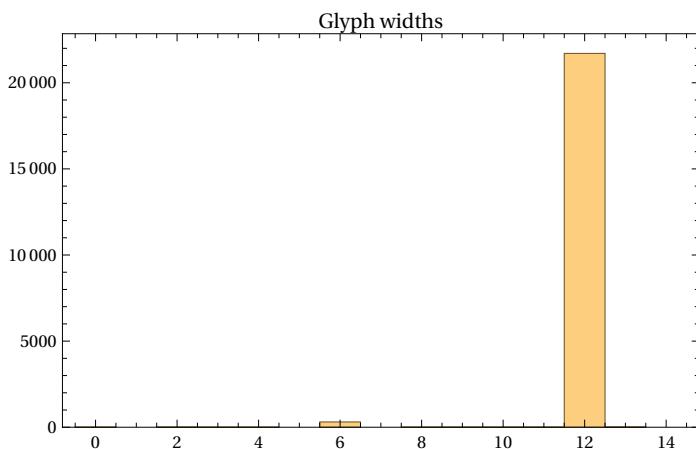
■ Extract character widths

```
In[13]:= Dimensions[widths = Flatten[Select[#, First[#] == "DWIDTH" &]]] & /@ charlines]
Out[13]= {22 043}
```

■ Character width statistics

```
In[14]:= TableForm[Sort[Tally[ToExpression/@widths], #1[[1]] < #2[[1]] &]]
Out[14]//TableForm=
0      2
2      6
3      2
4      1
6      305
8      2
9      3
10     3
11     3
12     21 706
13     10
```

```
In[15]:= Print[Histogram[ToExpression/@widths, {-\frac{1}{2}, \frac{29}{2}, 1},
Frame → True, PlotRange → All, PlotLabel → "Glyph widths"]];
```



■ Indices of characters with widths ≠ 12

```
In[16]:= nonstdwidth = Flatten[Position[ToExpression/@widths, w_ /; w ≠ 12]]  
Out[16]= {1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24,  
25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45,  
46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65,  
66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85,  
86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96, 97, 98, 100, 101, 104, 105, 106, 107,  
108, 109, 110, 113, 114, 115, 117, 119, 120, 121, 122, 123, 124, 125, 126, 127,  
128, 129, 130, 131, 132, 133, 134, 135, 136, 137, 138, 139, 140, 141, 142, 143,  
144, 145, 146, 147, 148, 149, 151, 152, 153, 154, 155, 156, 157, 158, 159, 160,  
161, 162, 163, 164, 165, 166, 167, 168, 169, 170, 171, 172, 173, 174, 175, 176,  
177, 178, 179, 180, 181, 183, 184, 185, 186, 187, 188, 189, 190, 191, 192, 193,  
194, 195, 196, 197, 198, 199, 200, 201, 202, 203, 204, 205, 206, 207, 208, 209,  
210, 211, 212, 325, 326, 329, 332, 333, 336, 339, 345, 346, 347, 349, 350, 351,  
352, 353, 354, 355, 356, 357, 358, 359, 360, 361, 362, 364, 365, 366, 369, 370,  
372, 373, 374, 375, 376, 387, 388, 403, 404, 405, 407, 412, 416, 418, 419, 420,  
437, 441, 442, 451, 452, 667, 668, 1354, 21834, 21835, 21836, 21837, 21838,  
21839, 21840, 21841, 21842, 21843, 21855, 21856, 21857, 21858, 21859, 21860,  
21861, 21891, 21974, 21975, 21976, 21977, 21978, 21979, 21980, 21981, 21982,  
21983, 21984, 21985, 21986, 21987, 21988, 21989, 21990, 21991, 21992, 21993,  
21994, 21995, 21996, 21997, 21998, 21999, 22000, 22001, 22002, 22003, 22004,  
22005, 22006, 22007, 22008, 22009, 22010, 22011, 22012, 22013, 22014, 22015,  
22016, 22017, 22018, 22019, 22020, 22021, 22022, 22023, 22024, 22025, 22026,  
22027, 22028, 22029, 22030, 22031, 22032, 22033, 22034, 22035, 22036}
```

■ Extract bounding box dimensions

```
In[17]:= Dimensions[boundingboxes = Flatten[Select[#, First[#] == "BBX" &]] & /@ charlines]  
Out[17]= {22 043, 5}
```

■ Extract bounding box sizes

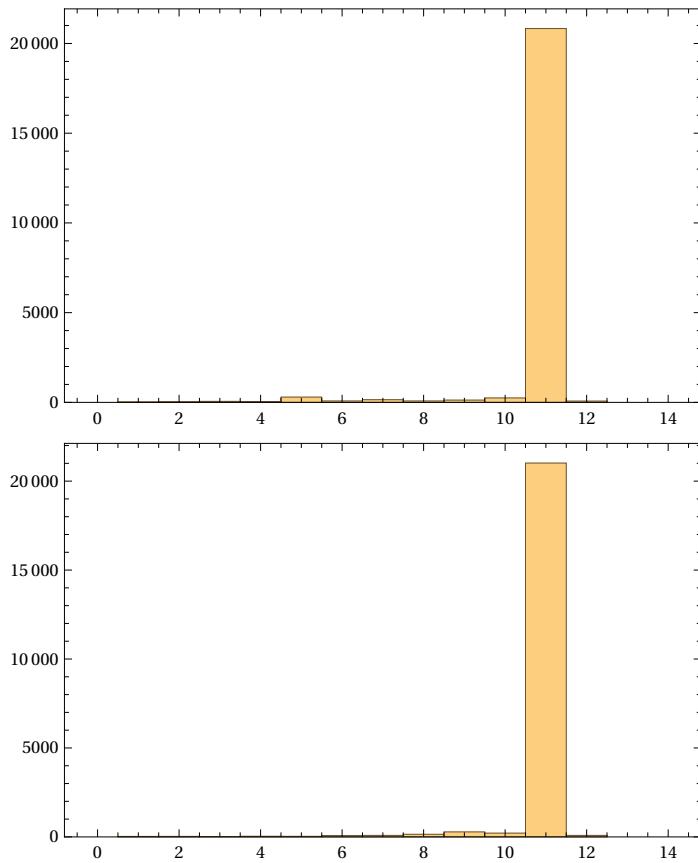
```
In[18]:= Dimensions[sizes = Take[#, {2, 3}] & /@ ToExpression[boundingboxes]]  
Out[18]= {22 043, 2}
```

■ Maximum bounding box size

```
In[19]:= maxsize = MapThread[Apply, {{Max, Max}, Transpose[sizes]}]  
Out[19]= {12, 12}
```

■ Bounding box size statistics

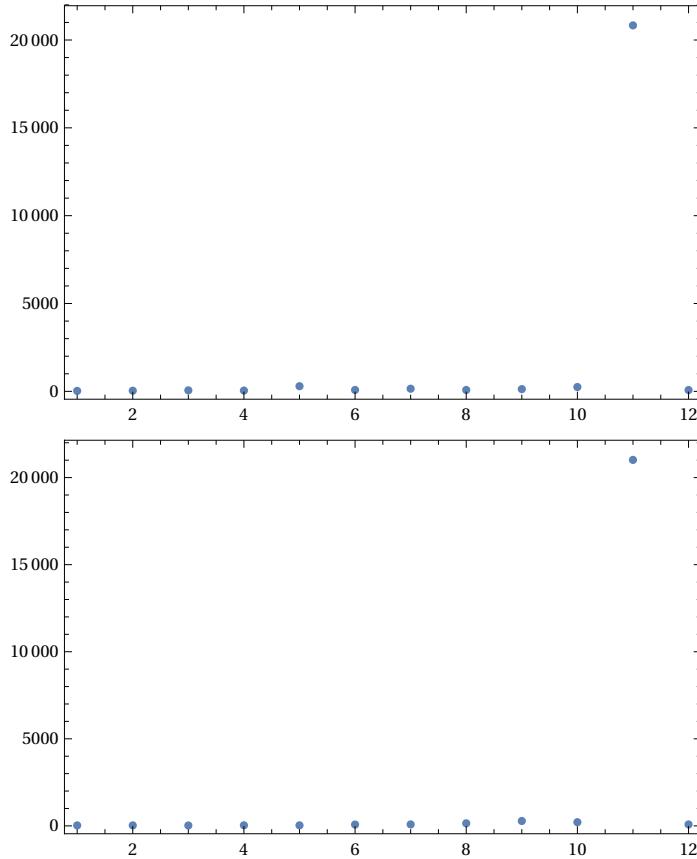
```
In[20]:= Print[Histogram[#, {-1/2, 29/2, 1}], Frame → True, PlotRange → All]] & /@ Transpose[sizes];
```



```
In[21]:= TableForm[Sort[Tally[#, #1[[1]] < #2[[1]] &] &/@Transpose[sizes]]
Print[ListPlot[#, Frame -> True, PlotRange -> All]] &/@%;
```

Out[21]/TableForm=

1	2	3	4	5	6	7	8	9	10	11	12
25	36	56	45	294	79	150	79	127	247	20832	73
1	2	3	4	5	6	7	8	9	10	11	12
27	28	23	35	31	76	84	147	281	212	21015	84



■ Extract bounding box positions

```
In[23]:= bounds = Flatten[{Take[#, -2], Take[#, -2] + Take[#, {2, 3}]}] & /@
  ToExpression[boundingboxes];
Dimensions[bounds]
```

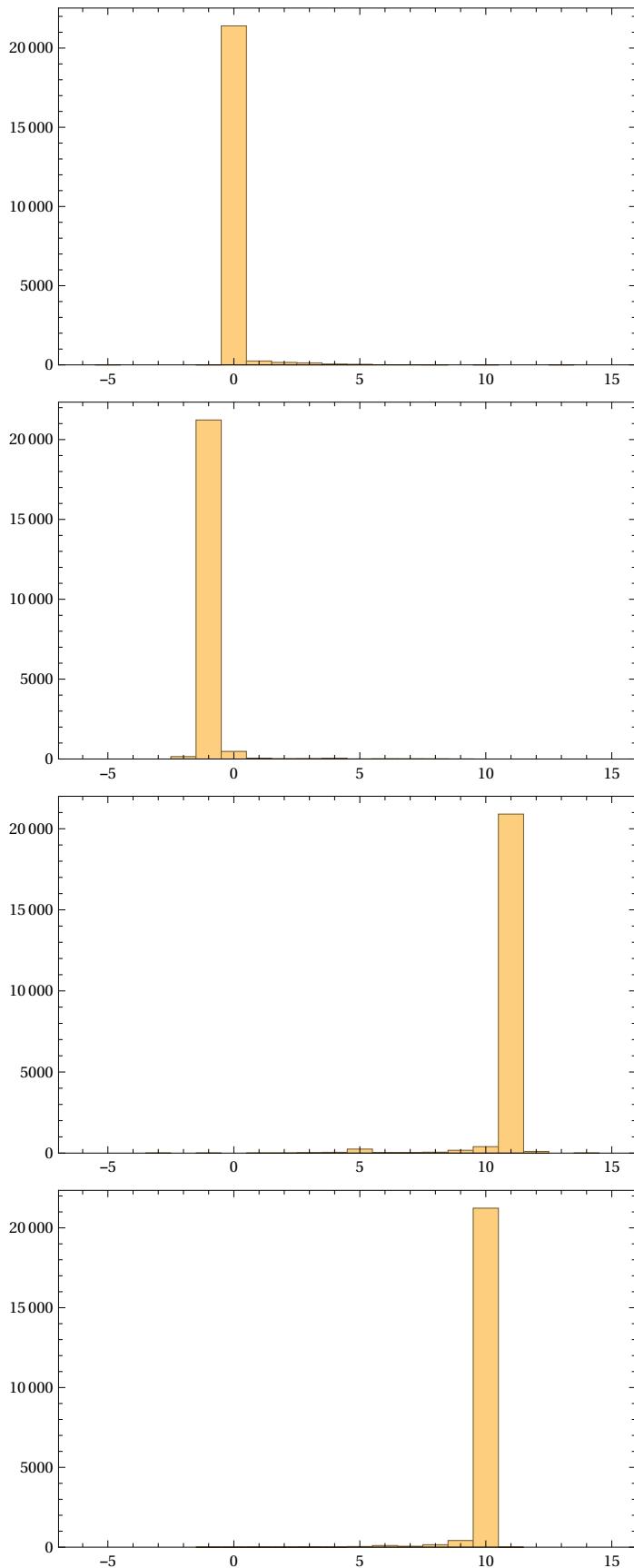
```
Out[24]= {22 043, 4}
```

■ Envelope of all bounding boxes

```
In[25]:= bigbox = MapThread[Apply, {{Min, Min, Max, Max}, Transpose[bounds]}]
Out[25]= {-5, -2, 14, 11}
```

■ Bounding box statistics

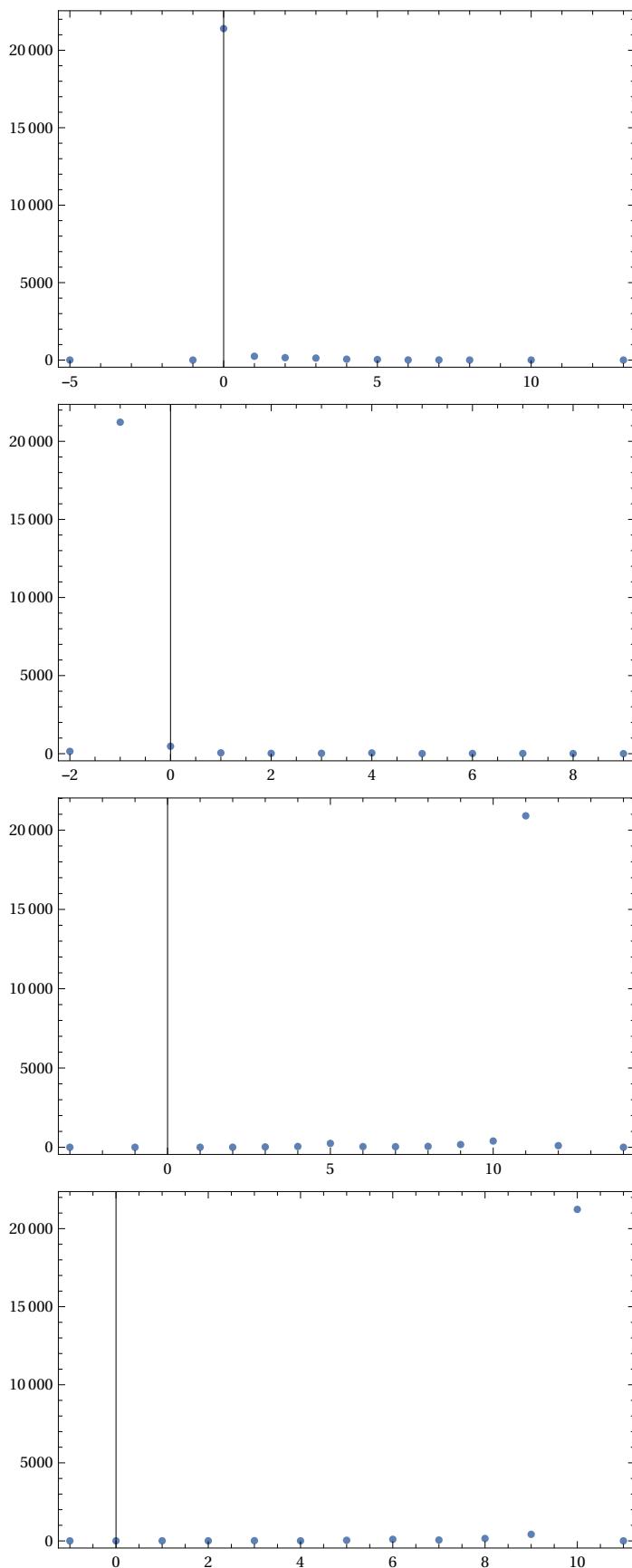
```
In[26]:= Print[Histogram[#, {- $\frac{13}{2}$ ,  $\frac{31}{2}$ , 1}, Frame → True, PlotRange → All]] & /@
  Transpose[bounds];
```



```
In[27]:= TableForm[Sort[Tally[#, #1[[1]] < #2[[1]] &] &/@Transpose[bounds]]]
Print[ListPlot[#, Frame -> True, PlotRange -> All]] &/@%;
```

Out[27]:= TableForm=

-5	-1	0	1	2	3	4	5	6	7	8	10
2	1	21400	246	155	128	55	35	9	8	2	1
-2	-1	0	1	2	3	4	5	6	7	8	9
149	21225	475	54	20	26	47	6	15	13	10	3
-3	-1	1	2	3	4	5	6	7	8	9	10
1	1	3	4	24	50	247	43	38	56	174	395
-1	0	1	2	3	4	5	6	7	8	9	10
3	3	7	4	13	5	41	102	56	155	419	21232



■ Indices of bounding box position outliers

```
In[30]:= outliers = Flatten[Position[bounds, {l_, b_, r_, t_} /; l < 0 || b < -1 || r > 11 || t > 11]]
```

```
Out[30]= {1, 13, 64, 93, 101, 116, 134, 166, 236, 237, 240, 241, 247, 250, 254, 255, 256, 263, 265, 281, 284, 295, 307, 310, 311, 313, 316, 327, 328, 329, 517, 518, 519, 520, 521, 522, 523, 524, 525, 526, 527, 528, 529, 530, 531, 532, 533, 534, 535, 536, 537, 538, 539, 540, 545, 546, 547, 548, 549, 550, 551, 552, 553, 554, 555, 556, 557, 558, 559, 560, 561, 562, 563, 564, 565, 566, 567, 568, 569, 570, 571, 572, 573, 574, 575, 576, 577, 578, 579, 580, 581, 582, 583, 584, 585, 586, 587, 588, 589, 590, 591, 592, 593, 594, 595, 596, 597, 598, 599, 600, 601, 602, 603, 607, 608, 609, 610, 611, 612, 613, 614, 615, 616, 617, 618, 619, 620, 621, 622, 623, 625, 626, 627, 628, 629, 630, 631, 632, 633, 634, 635, 636, 637, 638, 639, 640, 641, 642, 643, 646, 658, 659, 661, 668, 675, 1033, 2300, 3362, 5390, 6077, 6760, 6787, 8423, 9419, 13735, 13789, 14038, 15261, 15906, 16400, 17659, 17900, 20505, 21427, 21576, 21590, 21706, 21829, 21831, 21840, 21858, 21877, 21891, 21942, 21950, 21953, 21959, 21960, 21968, 21971, 22040}
```

■ Extend glyph bitmaps to full font bounding box dimensions

```
In[31]:= fills = # - bigbox & /@ bounds;
```

```
In[32]:= Take[fills, 16]
```

```
Out[32]= {{18, 0, 0, -12}, {7, 2, -11, -2}, {6, 9, -10, -2}, {5, 2, -9, -2}, {5, 1, -9, -1}, {5, 2, -8, -3}, {5, 2, -9, -2}, {7, 9, -11, -2}, {7, 1, -9, -1}, {6, 1, -10, -1}, {5, 4, -9, -4}, {5, 4, -9, -4}, {5, 0, -12, -10}, {5, 6, -9, -6}, {7, 2, -11, -9}, {6, 2, -10, -2}}
```

```
In[33]:= Dimensions[bitmaps = Function[{line},
```

```
Block[{bounds = ToExpression[Drop[Flatten[Select[line, First[#] == "BBX" &]], 1]]},
```

```
1 - Take[HexExpand[#], bounds[[1]]] & /@ Flatten[Take[line,
```

```
{1, -1} + Flatten[Position[line, #] & /@ {"BITMAP", "ENDCHAR"}]]]]]
```

```
]
```

```
]/@ charlines]
```

```
Out[33]= {22 043}
```

```

In[34]:= Take[bitmaps, 16]

Out[34]= {{{1}}, {{0}, {0}, {0}, {0}, {0}, {1}, {0}, {0}}, {{0, 1, 0}, {0, 1, 0}}, {{1, 0, 1, 0, 1}, {1, 0, 1, 0, 1}, {0, 0, 0, 0, 0}, {1, 0, 1, 0, 1}, {1, 0, 1, 0, 1}, {1, 0, 1, 0, 1}, {0, 0, 0, 0, 0}, {1, 0, 1, 0, 1}, {1, 0, 1, 0, 1}}, {{1, 1, 0, 1, 1}, {1, 0, 0, 0, 1}, {0, 1, 0, 1, 0}, {0, 1, 0, 1, 1}, {1, 0, 0, 1, 1}, {1, 1, 0, 1, 1}, {1, 0, 0, 0, 1}, {0, 1, 0, 1, 0}, {0, 1, 0, 1, 1}, {1, 0, 0, 1, 1}, {1, 1, 0, 1, 1}, {1, 0, 0, 0, 1}, {1, 1, 0, 0, 1}, {1, 1, 0, 1, 0}, {0, 1, 0, 1, 0}, {1, 0, 0, 0, 1}, {1, 1, 0, 1, 1}}, {{1, 0, 1, 1, 0, 1}, {0, 1, 0, 1, 0, 1}, {0, 1, 0, 0, 1, 1}, {1, 0, 1, 0, 1, 1}, {1, 1, 0, 1, 0, 1}, {1, 0, 1, 0, 1, 0}, {1, 0, 1, 1, 0, 1}, {0, 1, 1, 0, 0, 1}, {1, 0, 1, 0, 1, 0}, {1, 0, 1, 1, 0, 1}, {1, 0, 0, 1, 1}, {0, 1, 1, 0, 1}, {0, 1, 0, 1, 1}, {1, 0, 0, 1, 1}, {1, 0, 1, 1, 1}, {0, 1, 0, 1, 0}, {0, 1, 1, 0, 1}, {1, 0, 0, 1, 0}, {{0}, {0}}, {{1, 1, 0}, {1, 0, 1}, {0, 1, 1}, {0, 1, 1}, {0, 1, 1}, {0, 1, 1}, {0, 1, 1}, {0, 1, 1}, {1, 0, 1, 1, 0, 1}, {1, 1, 0, 0, 1, 0}, {1, 0, 1, 0, 1, 0}, {1, 0, 1, 1, 0, 1}, {1, 0, 0, 1, 1, 0}, {0, 1, 0, 1, 0, 0}, {0, 1, 1, 0, 1, 0}, {1, 0, 0, 1, 0, 0}, {0, 1, 0, 1, 1, 0}, {1, 0, 0, 1, 1, 0}, {1, 1, 0, 1, 0, 0}, {1, 0, 1, 0, 1, 0}, {1, 0, 1, 1, 0, 0}, {1, 1, 0, 1, 1, 0}, {1, 1, 0, 1, 0, 1}, {1, 0, 1, 0, 1, 1}, {1, 0, 1, 1, 0, 1}, {1, 1, 0, 1, 1, 0}, {1, 1, 0, 1, 0, 1}, {1, 0, 1, 0, 1, 1}, {1, 0, 1, 1, 0, 1}, {1, 1, 0, 1, 1, 0}, {1, 1, 0, 1, 0, 1}, {1, 0, 1, 0, 1, 1}, {1, 0, 1, 1, 0, 1}, {1, 1, 0, 1, 1, 0}, {1, 1, 0, 1, 0, 1}}}

In[35]:= BitmapExtend = Function[{bitmap, fill},
  Block[{bmp, xl = Table[1, {0, fill[[1]]}], xr = Table[1, {0, -fill[[3]]}], yt, yb},
    bmp = Join[xl, #, xr] & /@ bitmap;
    yb = Table[Table[1, {0, Length[First[bmp]]}], {0, fill[[2]]}];
    yt = Table[Table[1, {0, Length[First[bmp]]}], {0, -fill[[4]]}];
    Join[yt, bmp, yb]
  ]
]

Out[35]= Function[{bitmap, fill},
  Block[{bmp, xl = Table[1, {0, fill[[1]]}], xr = Table[1, {0, -fill[[3]]}], yt, yb},
    bmp = (Join[xl, #1, xr] &) /@ bitmap;
    yb = Table[Table[1, {0, Length[First[bmp]]}], {0, fill[[2]]}];
    yt = Table[Table[1, {0, Length[First[bmp]]}], {0, -fill[[4]]}];
    Join[yt, bmp, yb]]]

In[36]:= Dimensions[extbitmaps = MapThread[BitmapExtend, {bitmaps, fills}]]]

Out[36]= {22 043, 13, 19}

In[37]:= bigbox

Out[37]= {-5, -2, 14, 11}

  ■ Truncate glyphs to 12×12 bitmap

In[38]:= BitmapTruncate = Function[bitmap, Take[#, {6, 17}] & /@ Take[bitmap, {2, 13}]]]

Out[38]= Function[bitmap, (Take[#, {6, 17}] &) /@ Take[bitmap, {2, 13}]]]

In[39]:= Dimensions[trbitmaps = BitmapTruncate /@ extbitmaps]

Out[39]= {22 043, 12, 12}

```

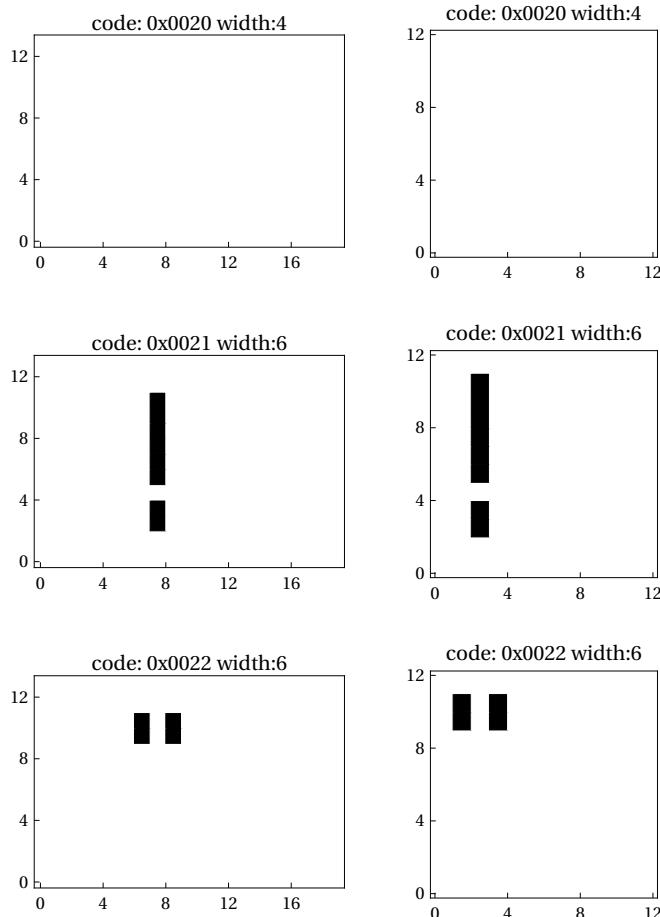
Display glyphs

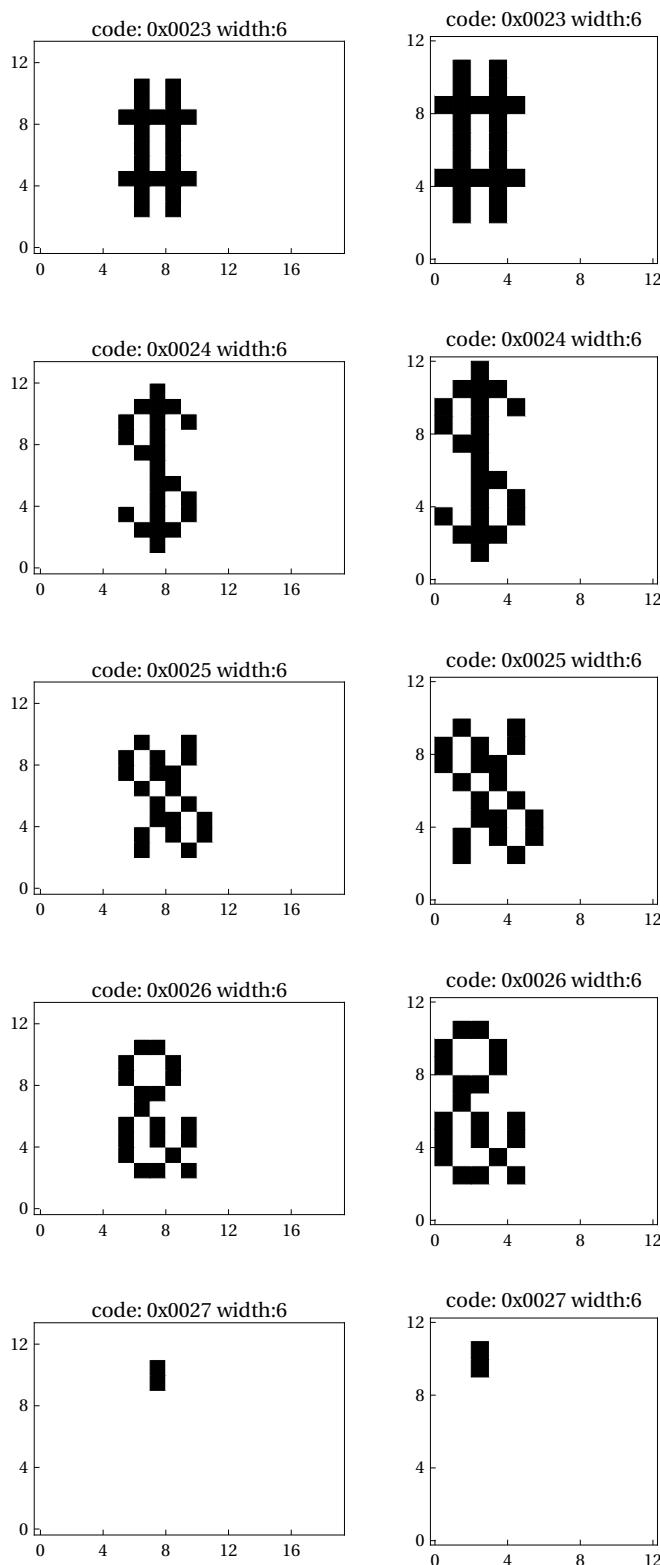
■ Non standard width glyphs

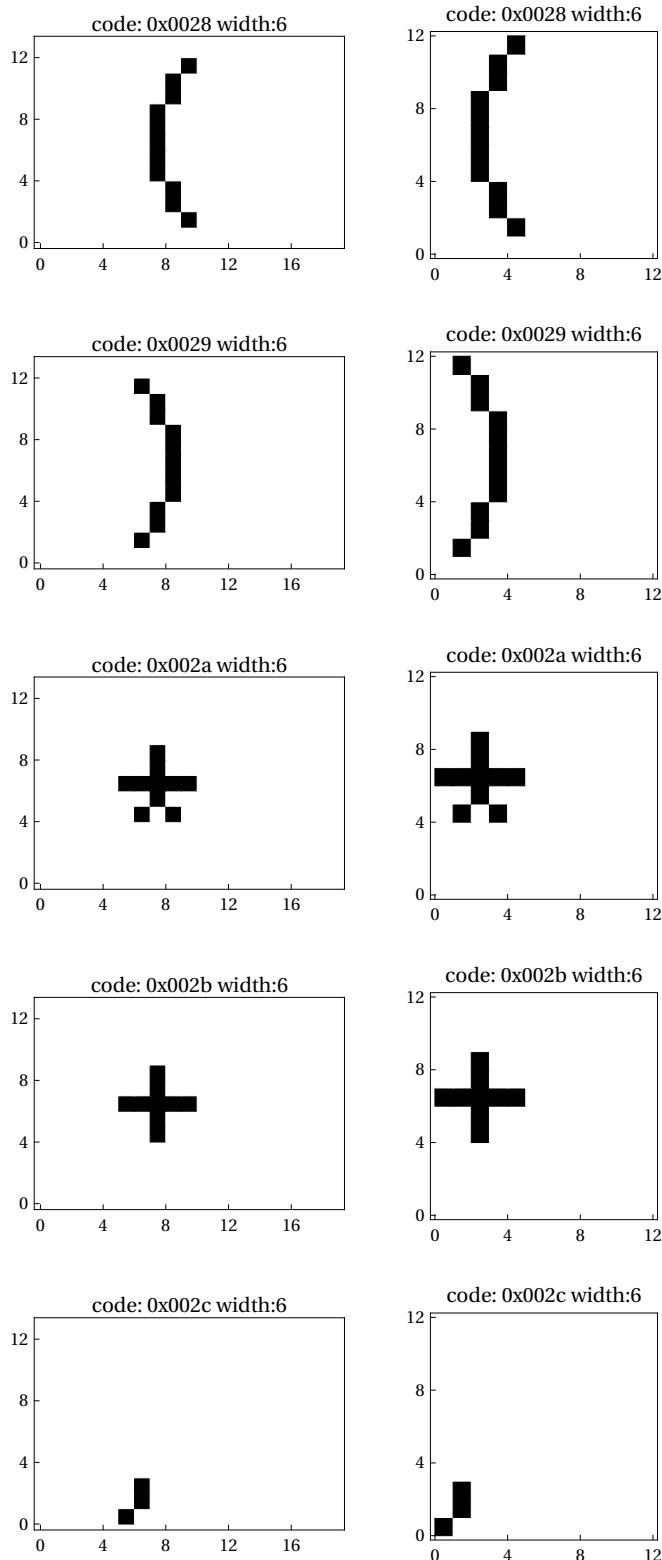
```
In[40]:= nonstdwidthgraphics =
  MapThread[Graphics[Raster[Reverse[#1]], Frame → True, AspectRatio →
    Divide @@ Dimensions[#1], FrameTicks → ({#, #, {}, {}} & [Range[0, 16, 4]]),
    PlotLabel → ("code: " <> IntegerString[ToExpression[#2], 16, 4] <> " width:" <>
      ToString[#3])] &, #[[nonstdwidth]] & /@ {extbitmaps, charcodes, widths}];

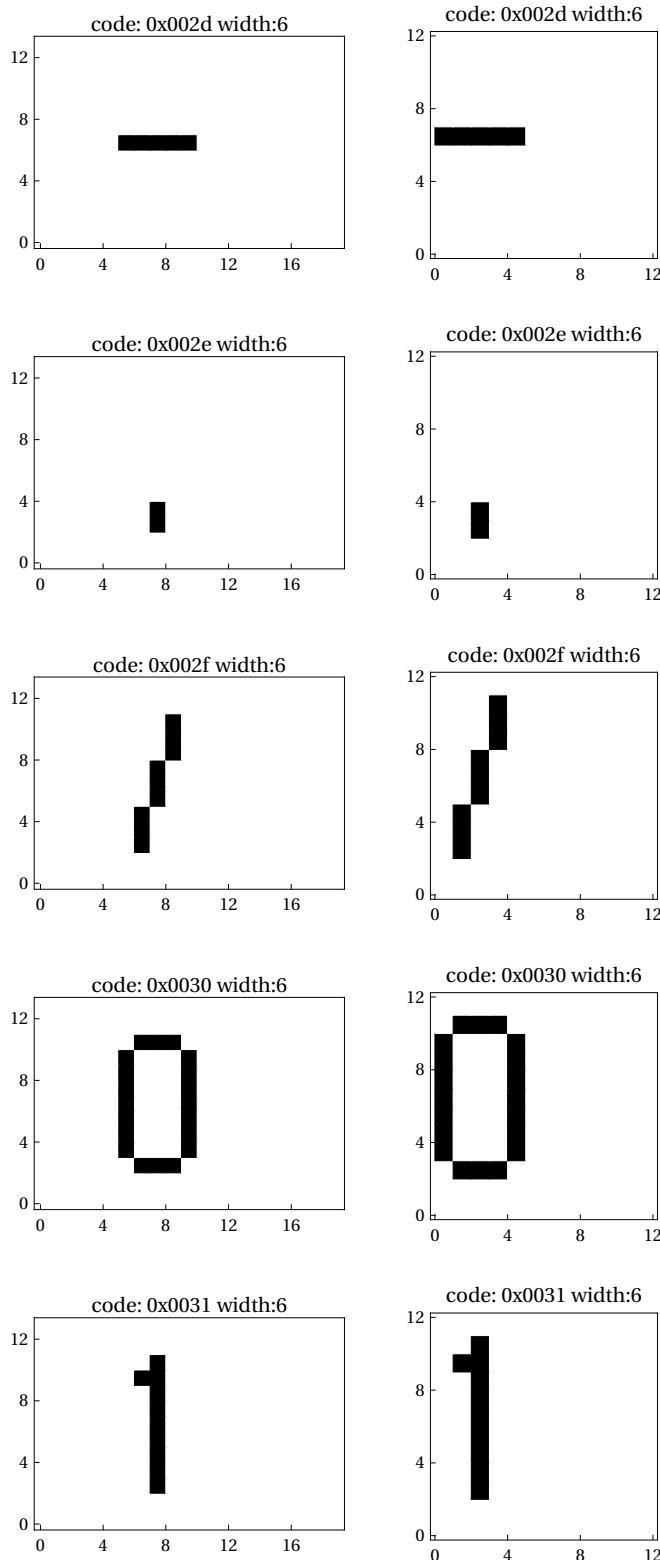
In[41]:= nonstdwidthtruncgraphics =
  MapThread[Graphics[Raster[Reverse[#1]], Frame → True, AspectRatio →
    Divide @@ Dimensions[#1], FrameTicks → ({#, #, {}, {}} & [Range[0, 16, 4]]),
    PlotLabel → ("code: " <> IntegerString[ToExpression[#2], 16, 4] <> " width:" <>
      ToString[#3])] &, #[[nonstdwidth]] & /@ {trbitmaps, charcodes, widths}];

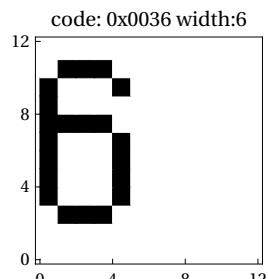
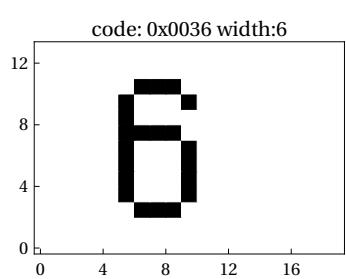
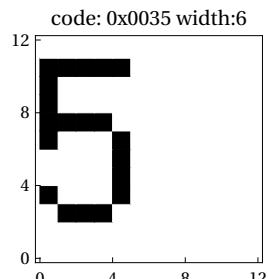
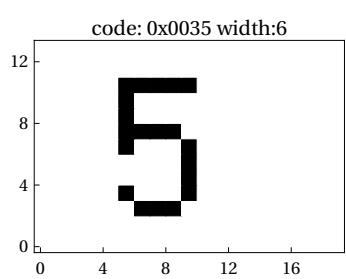
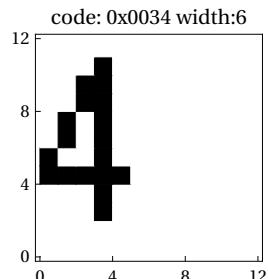
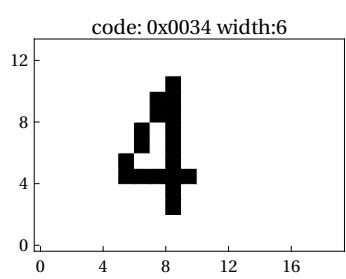
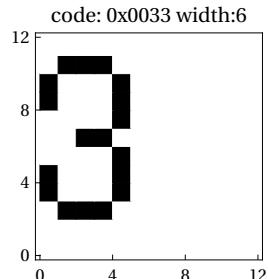
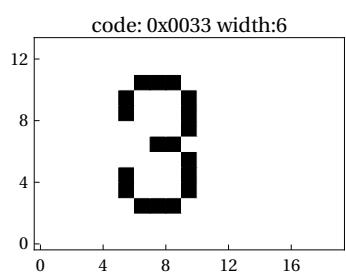
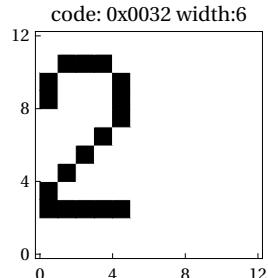
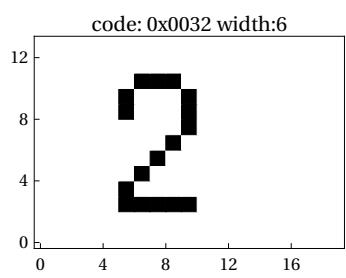
In[42]:= MapThread[Print[GraphicsGrid[{{##}}]]] &,
{nonstdwidthgraphics, nonstdwidthtruncgraphics};
```

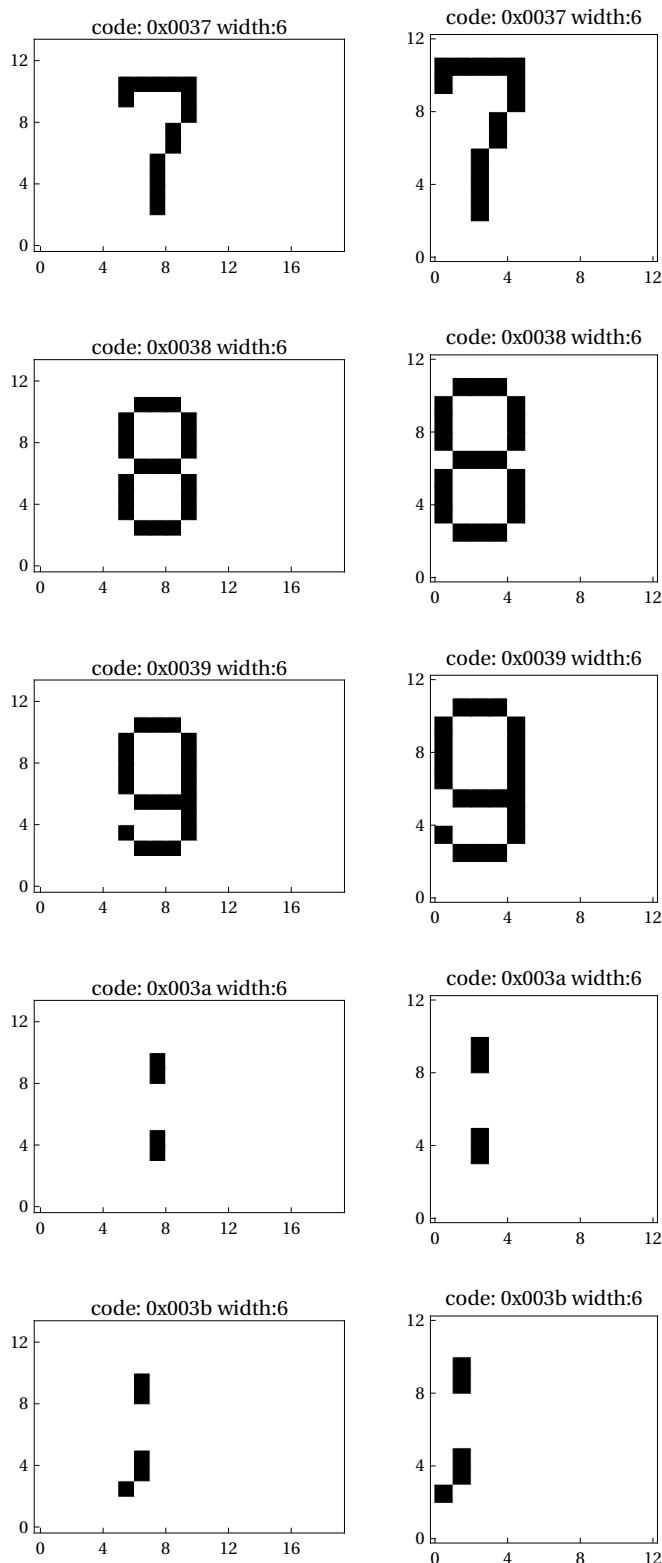


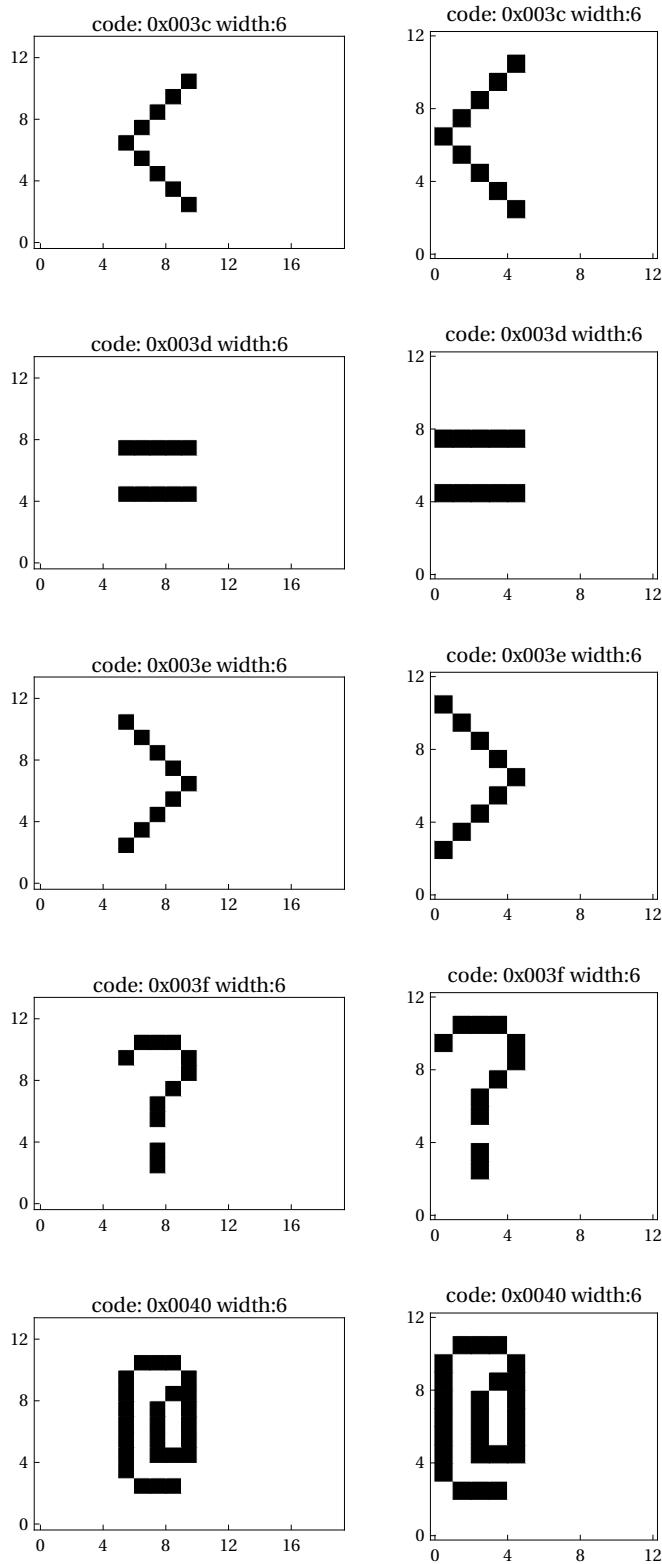


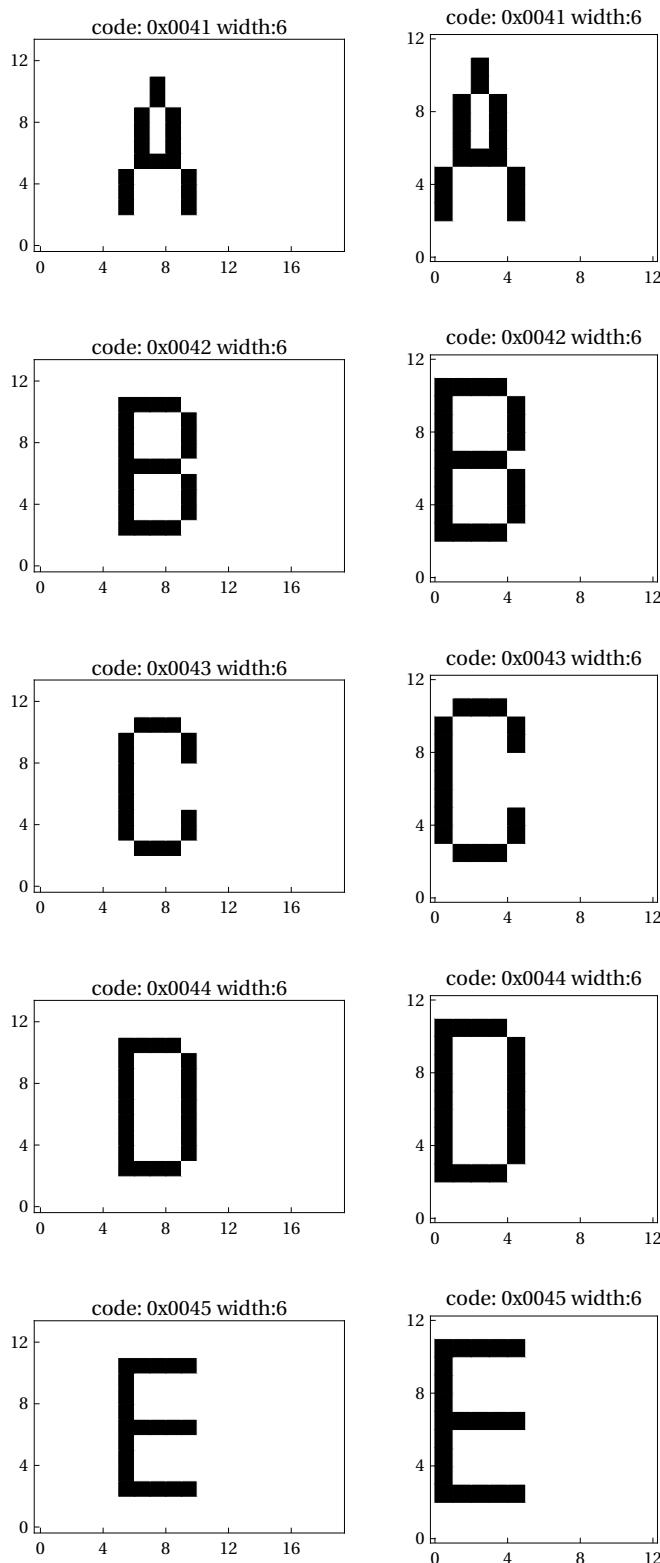


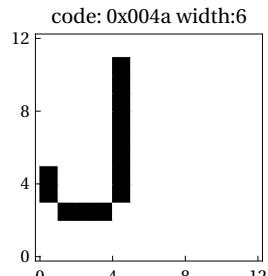
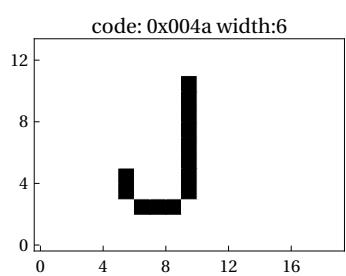
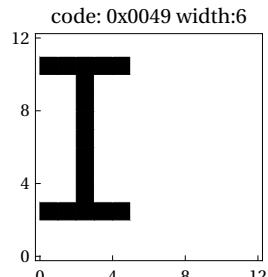
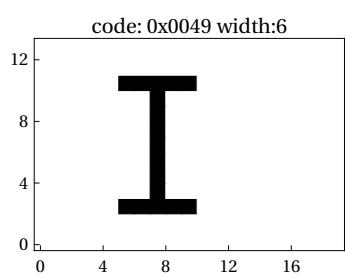
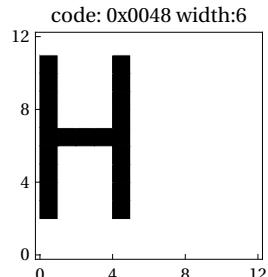
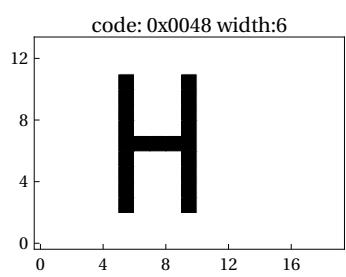
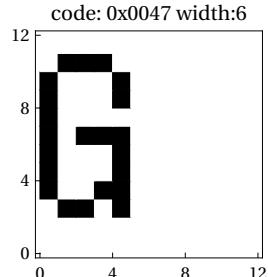
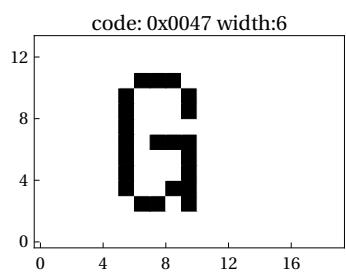
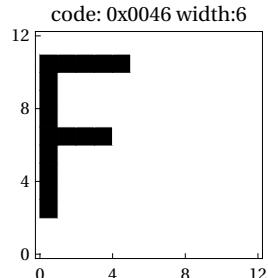
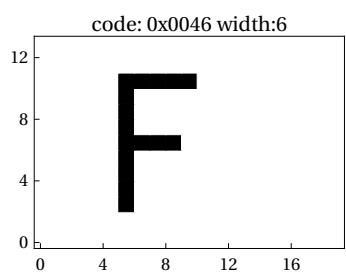


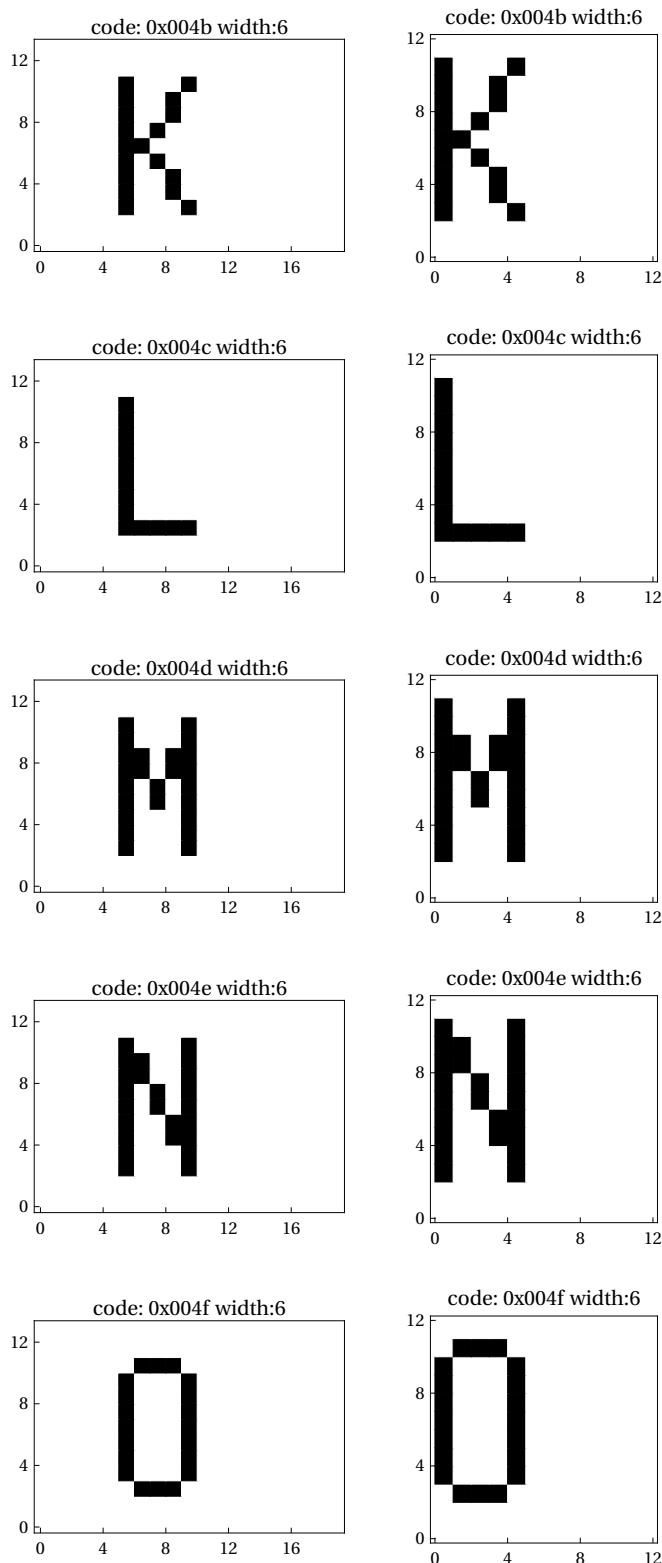


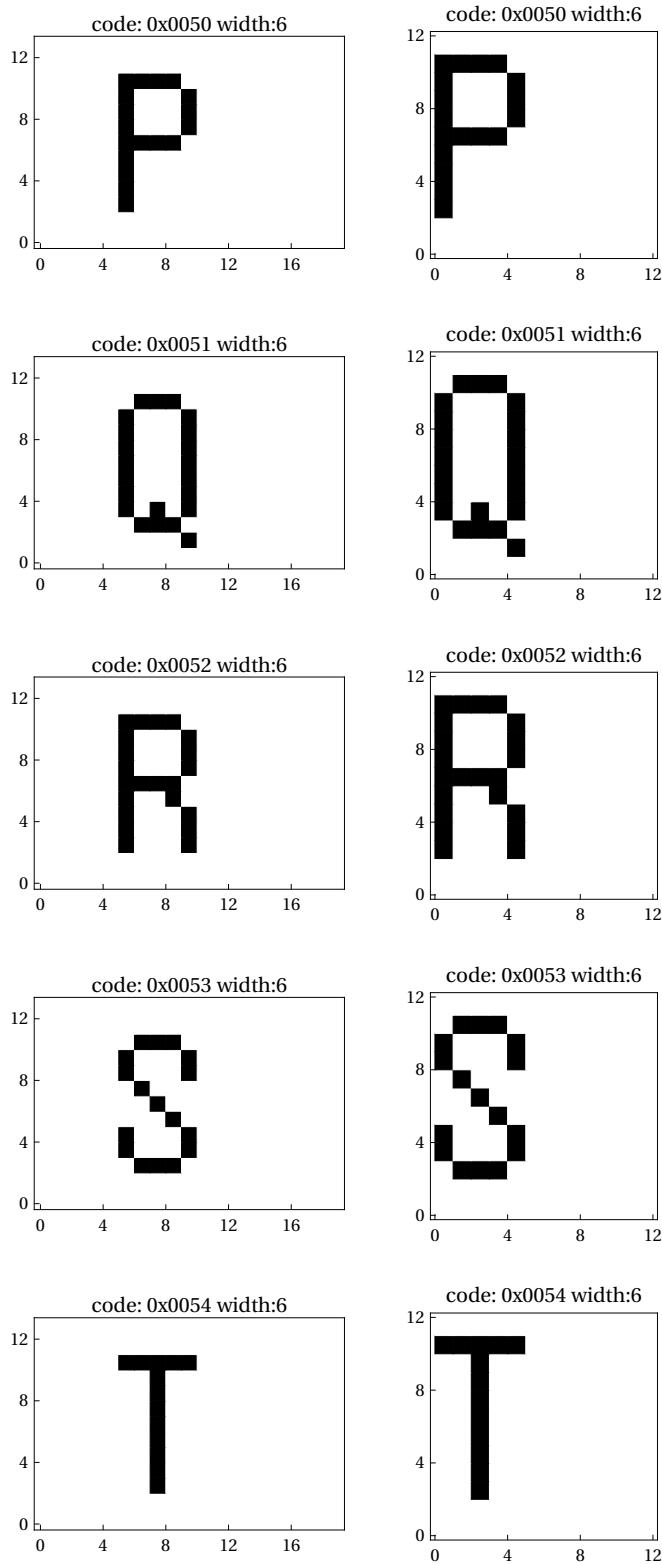


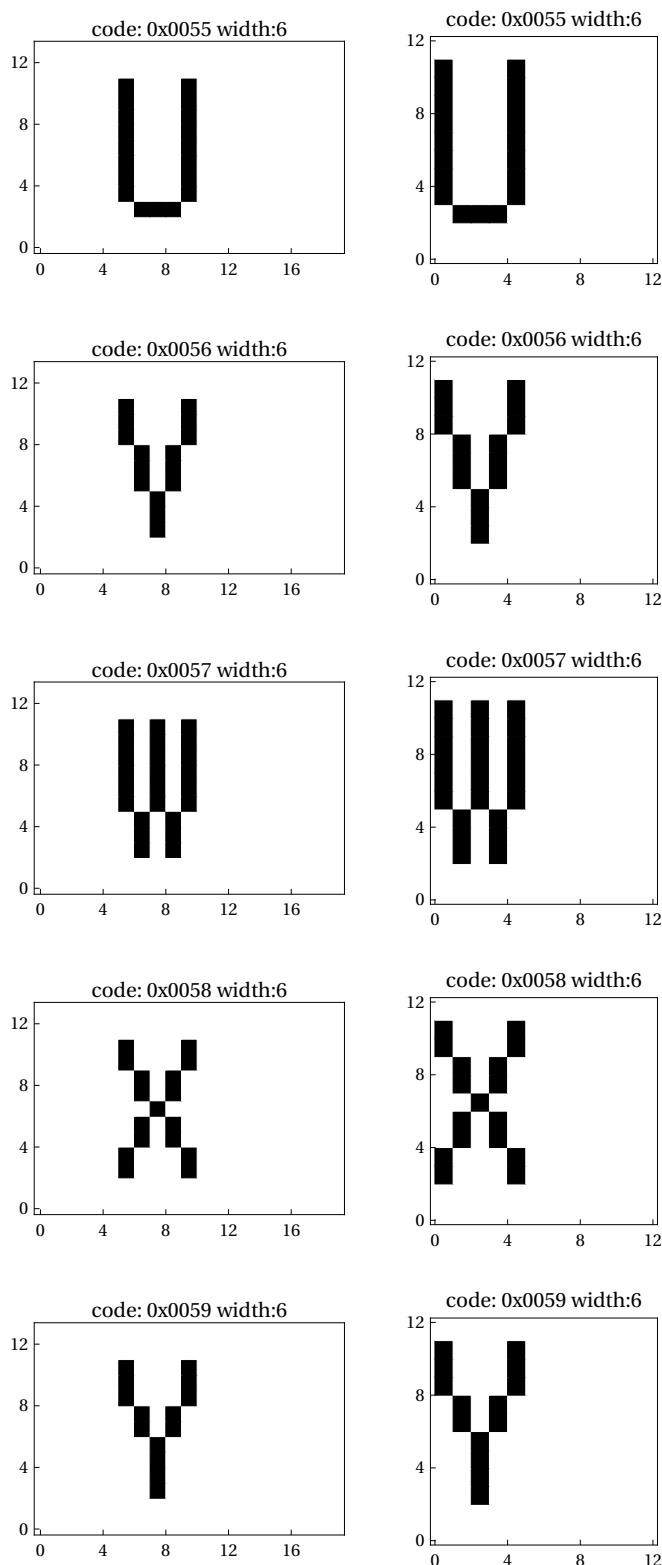


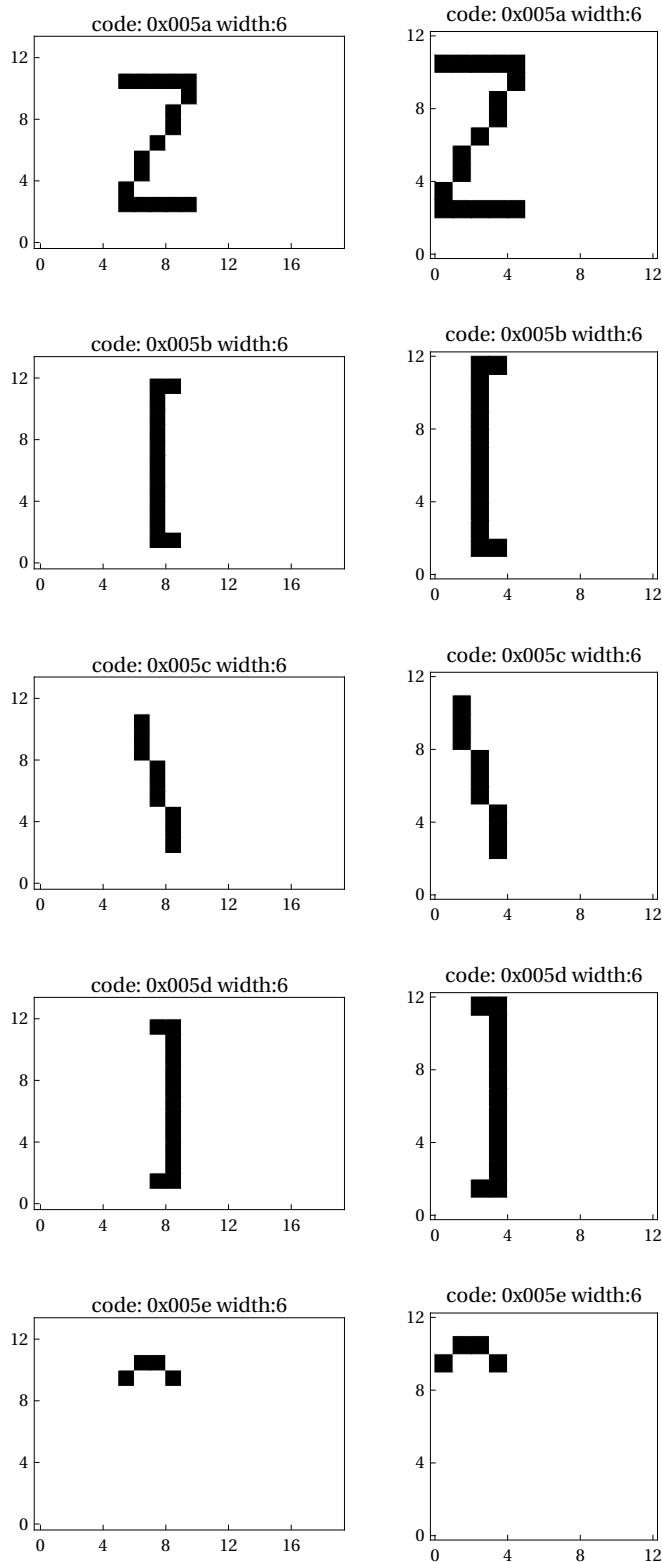


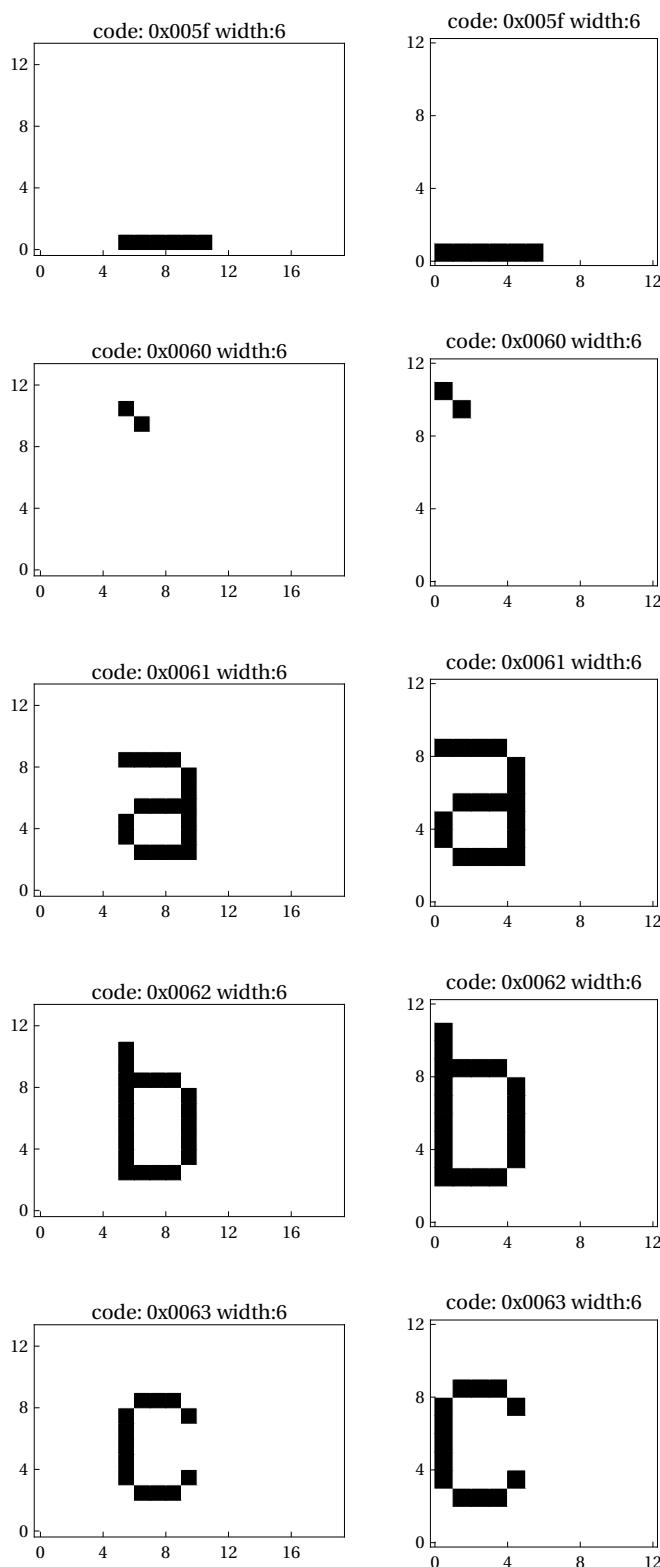


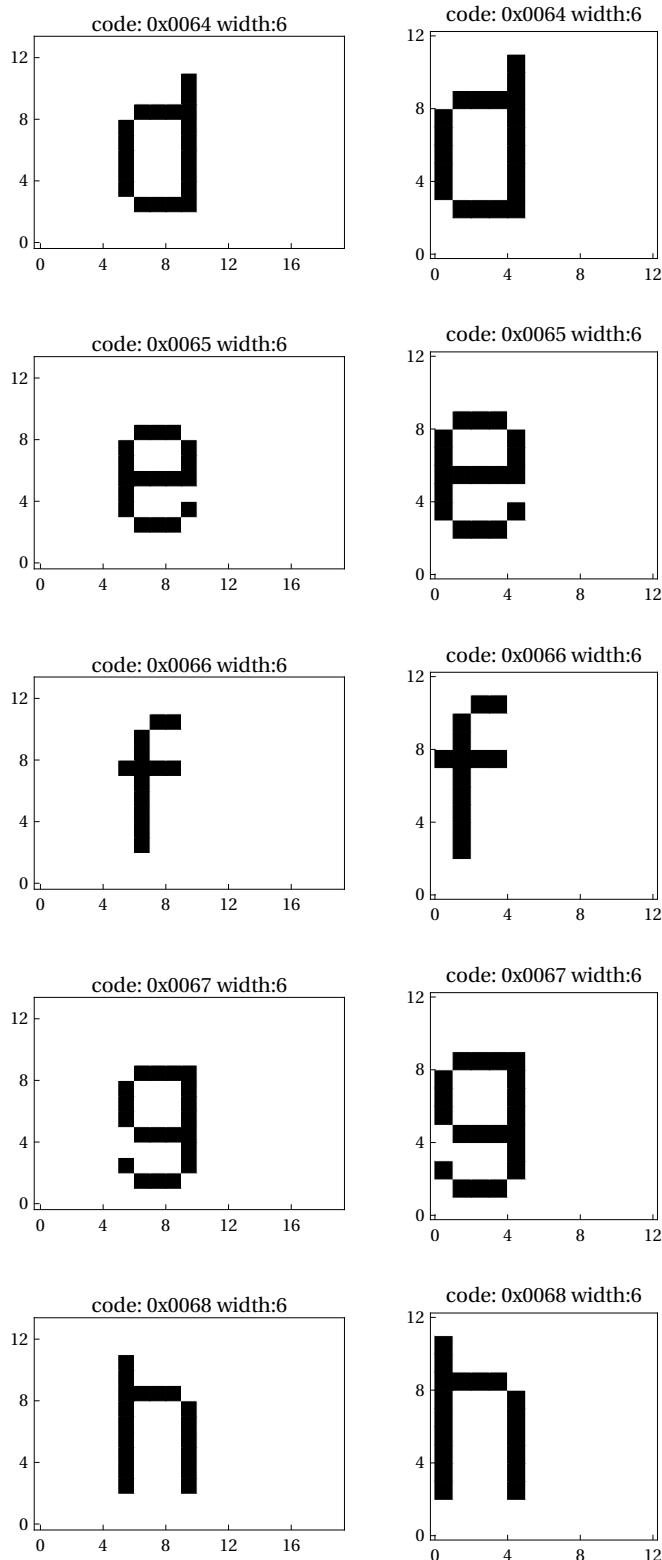


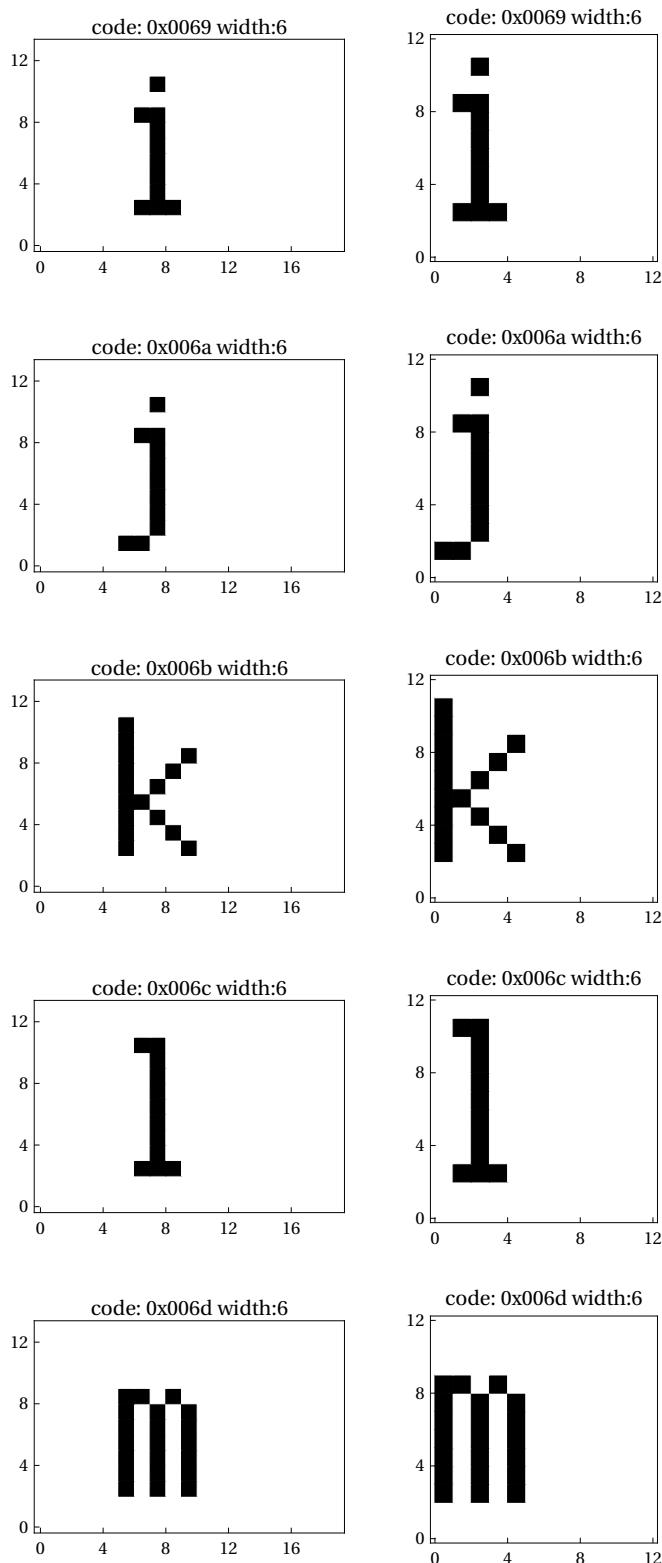


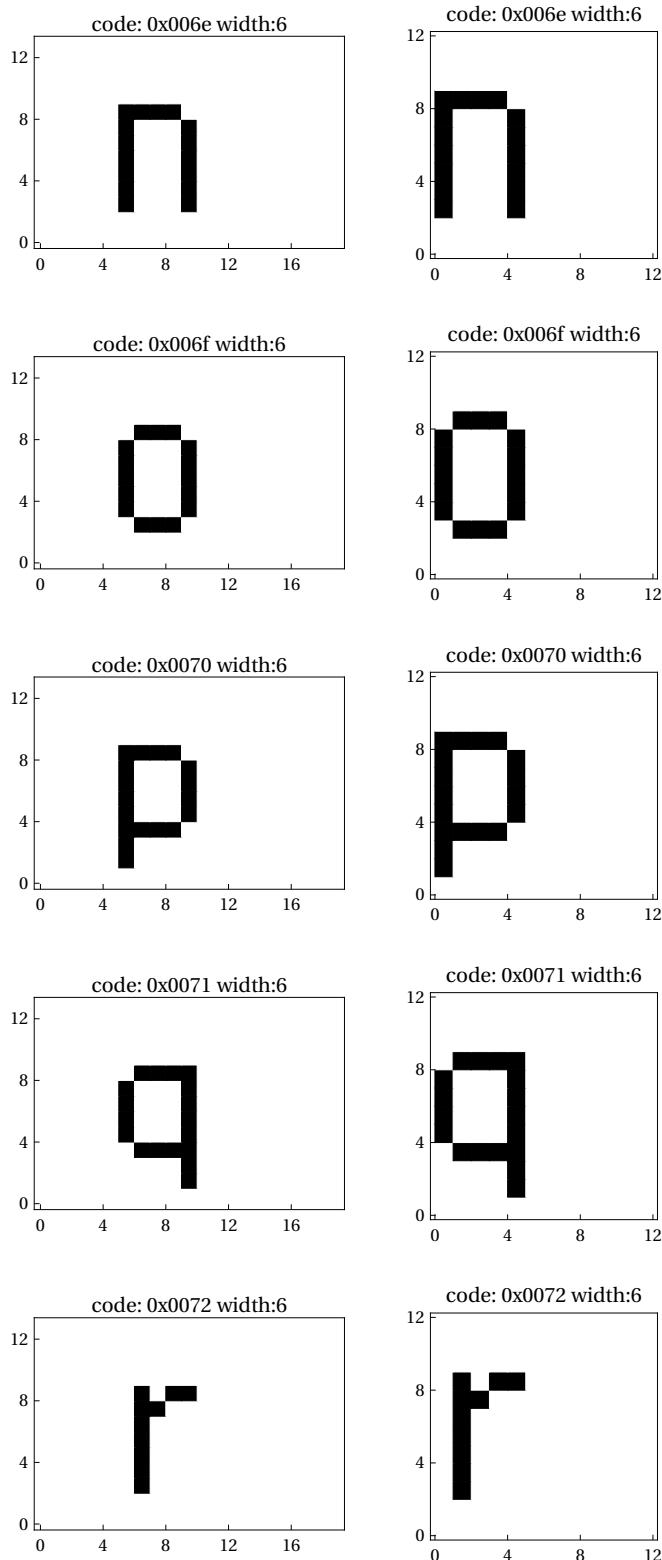


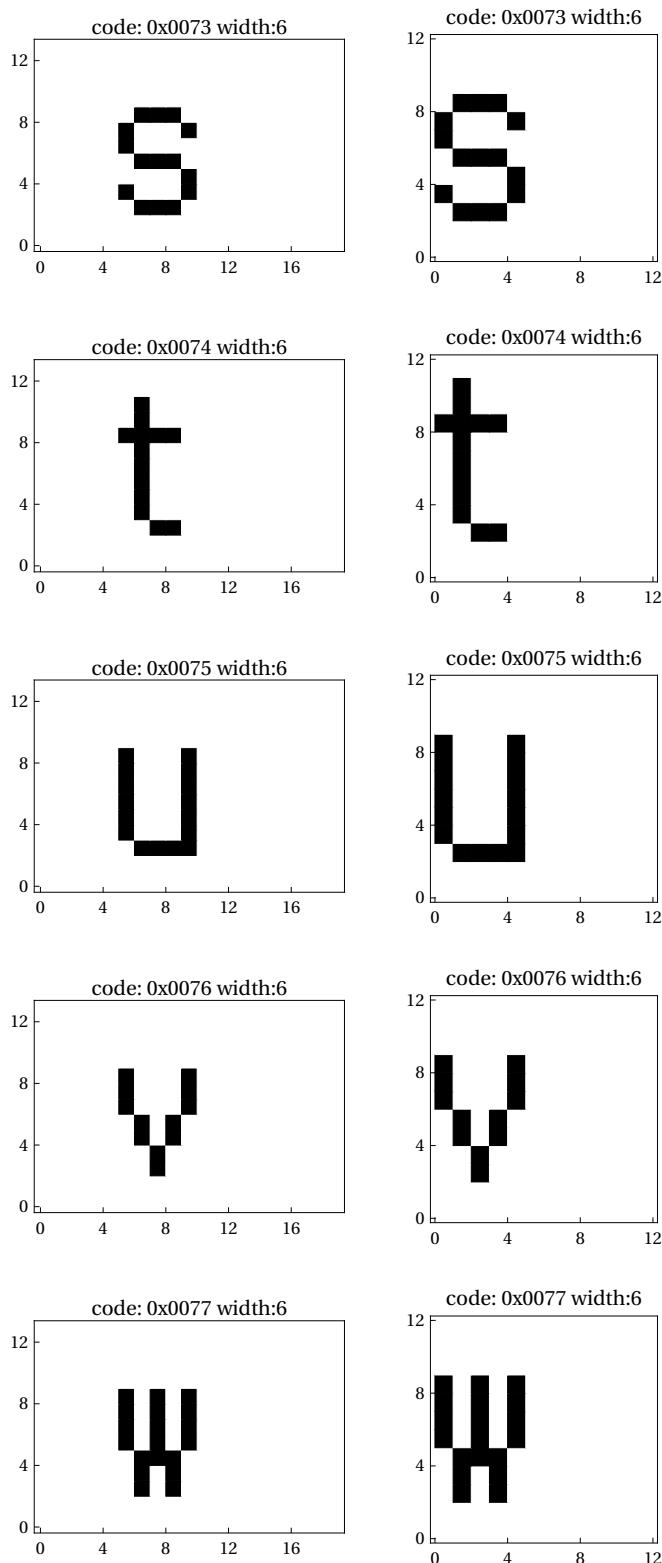


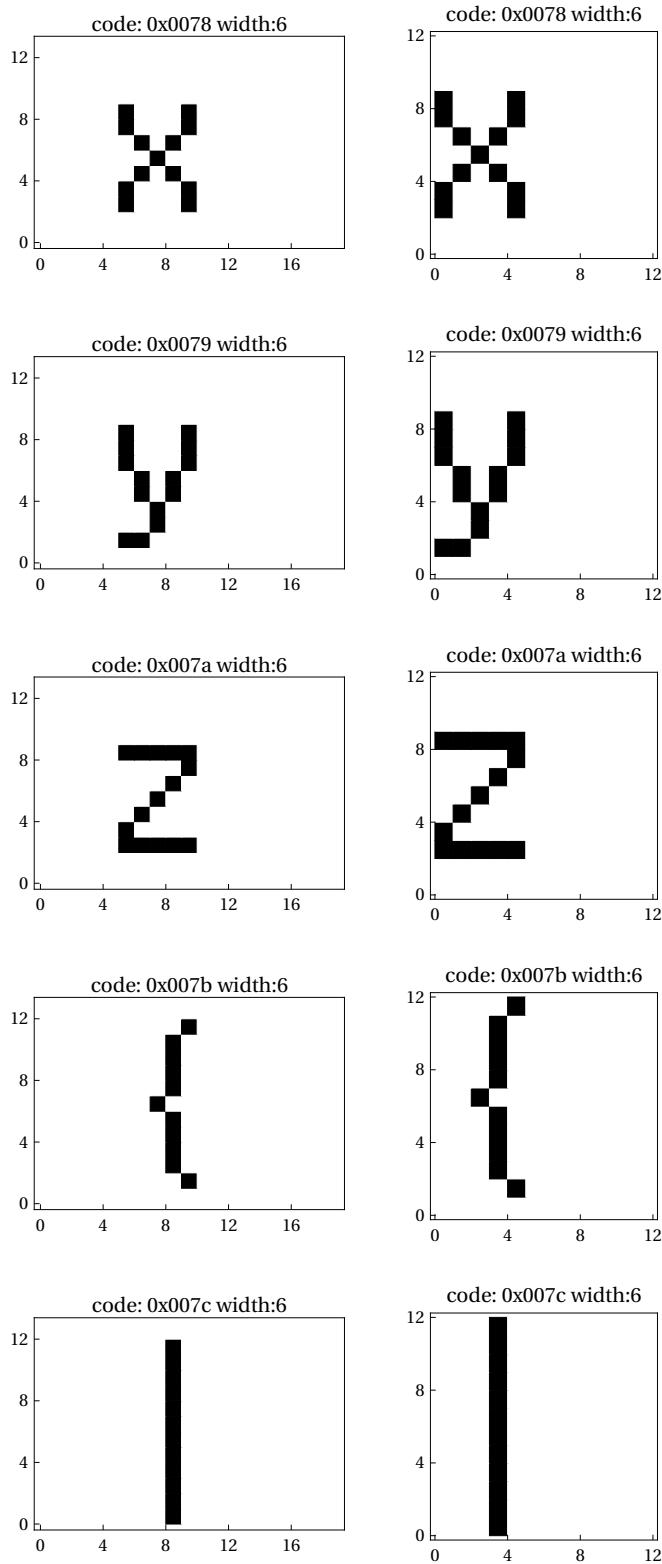


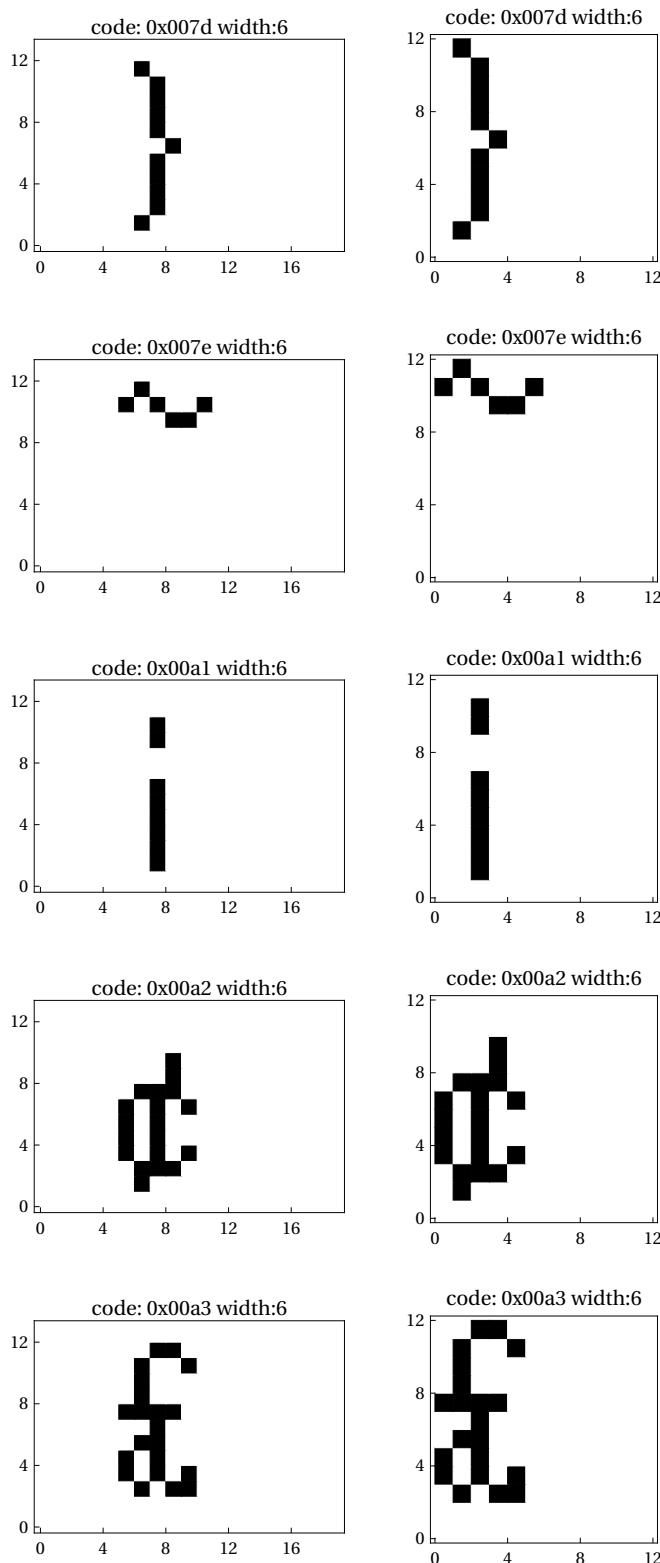


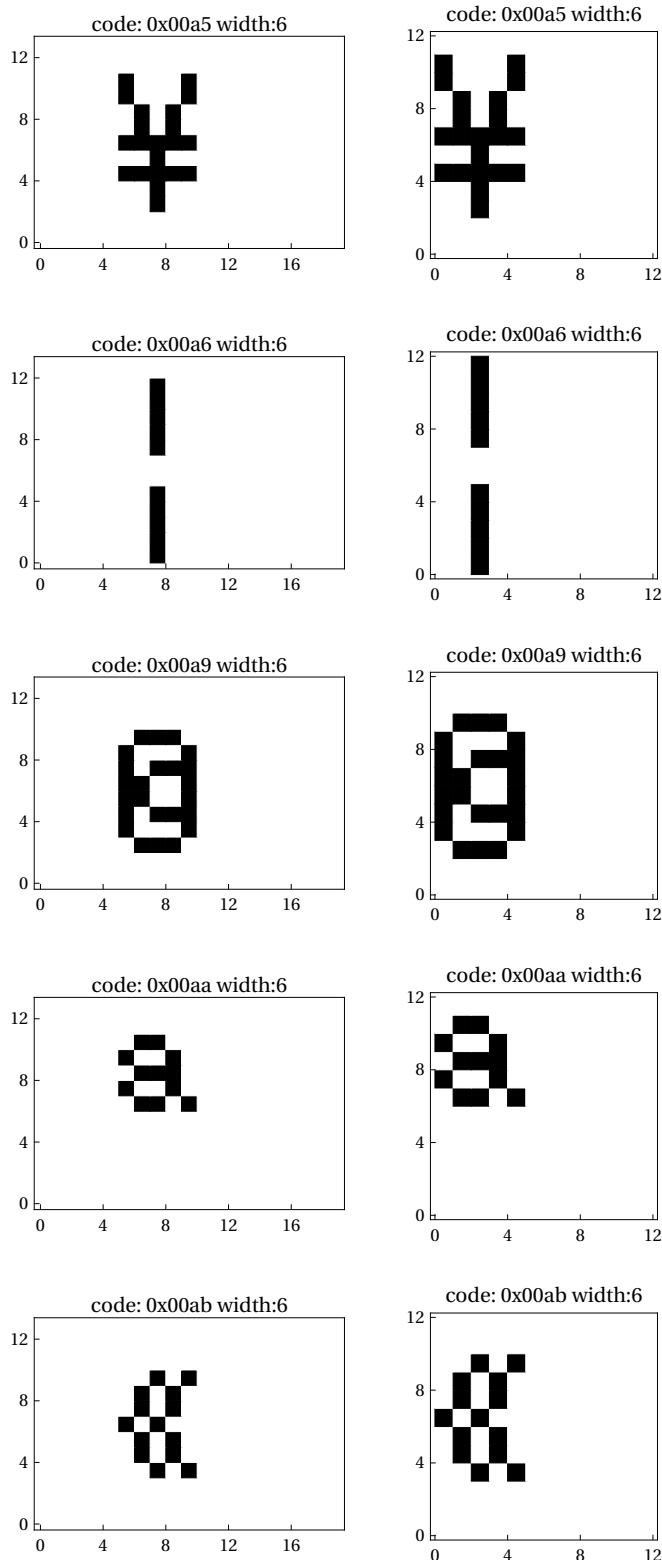


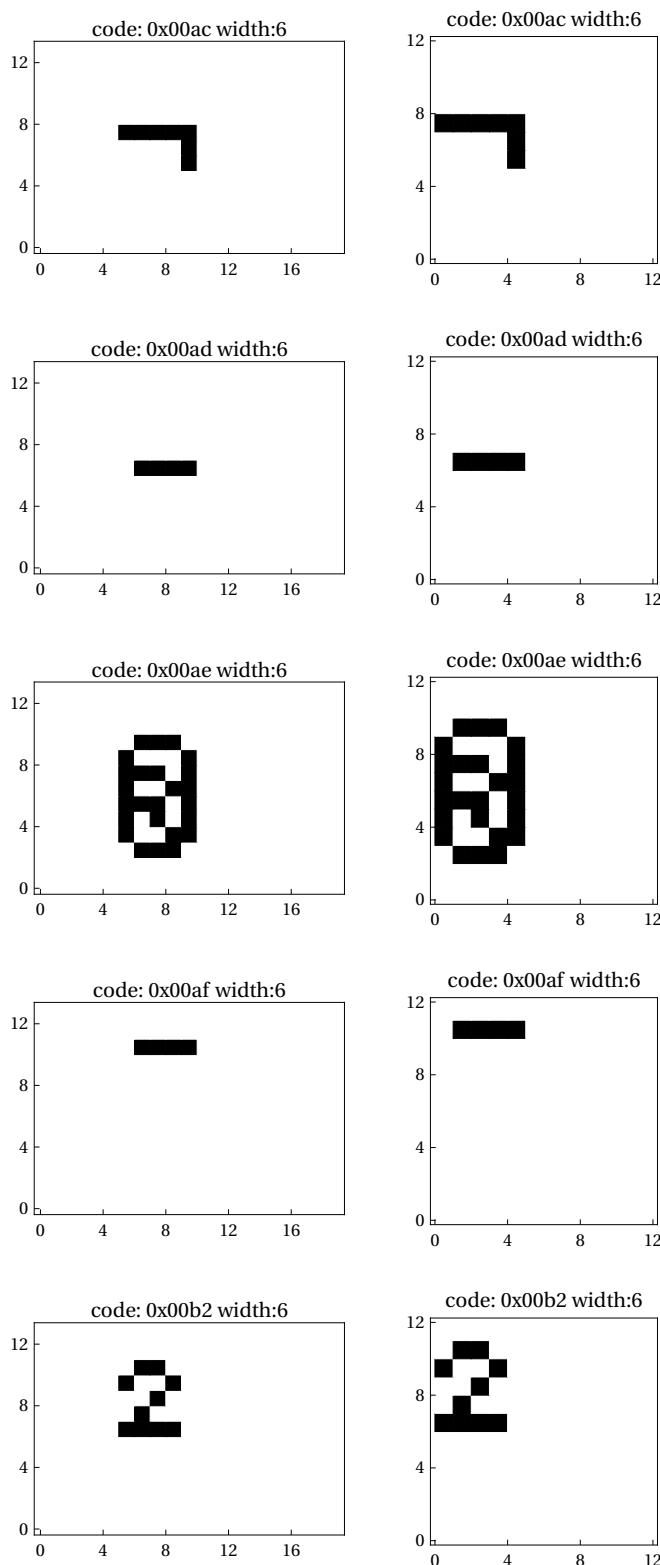


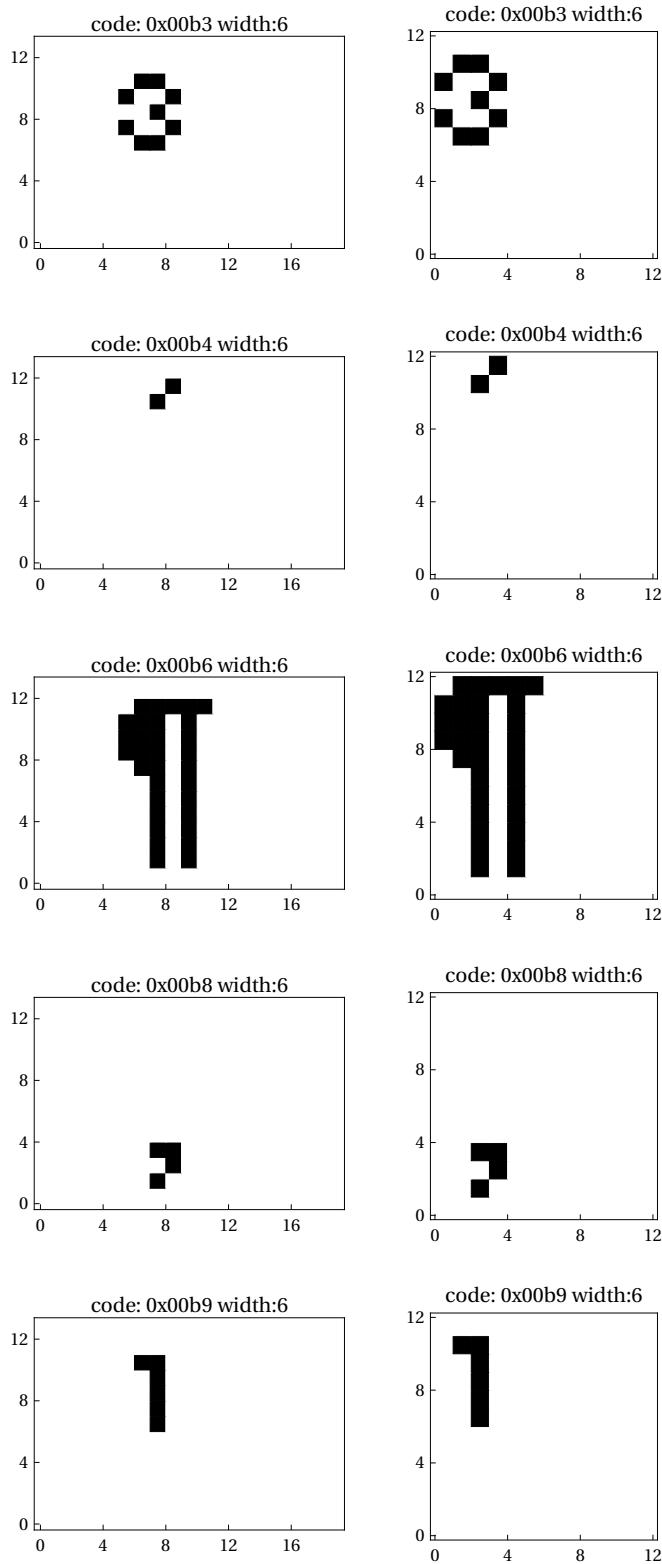


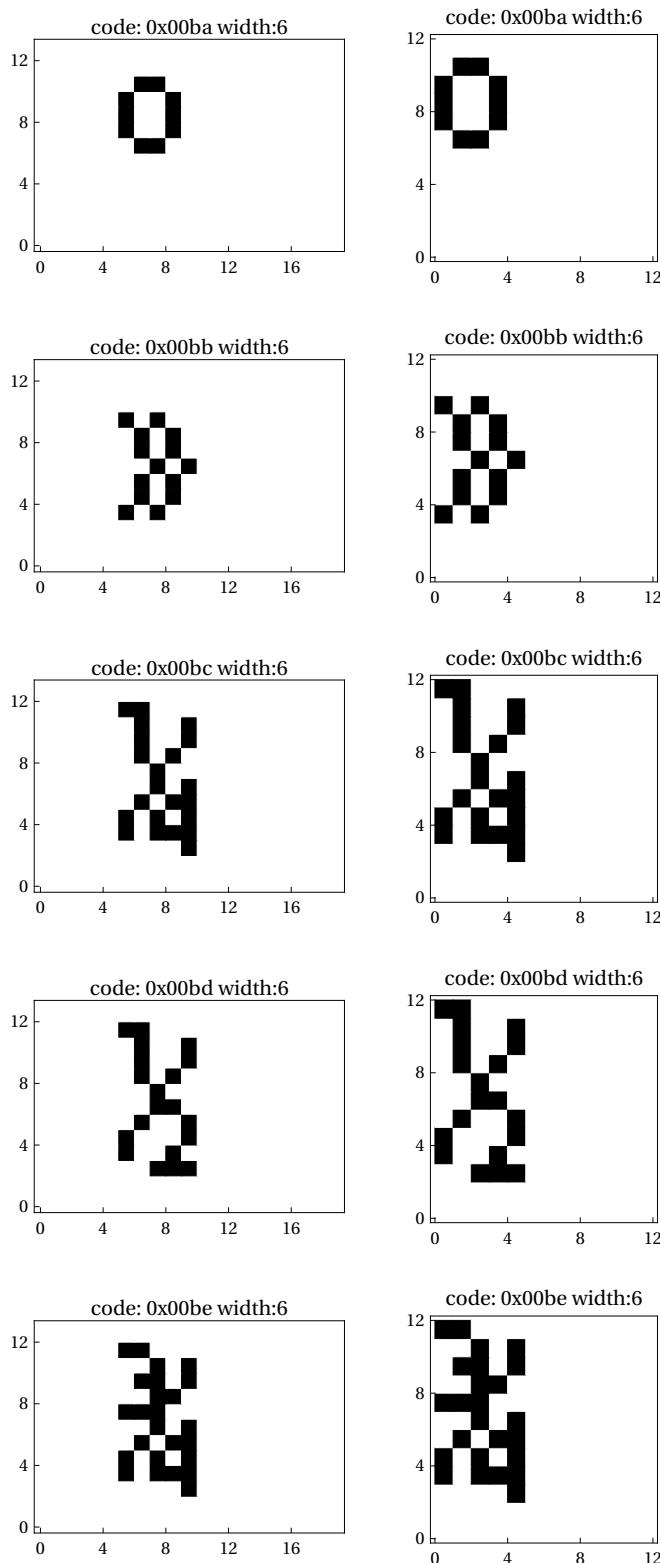


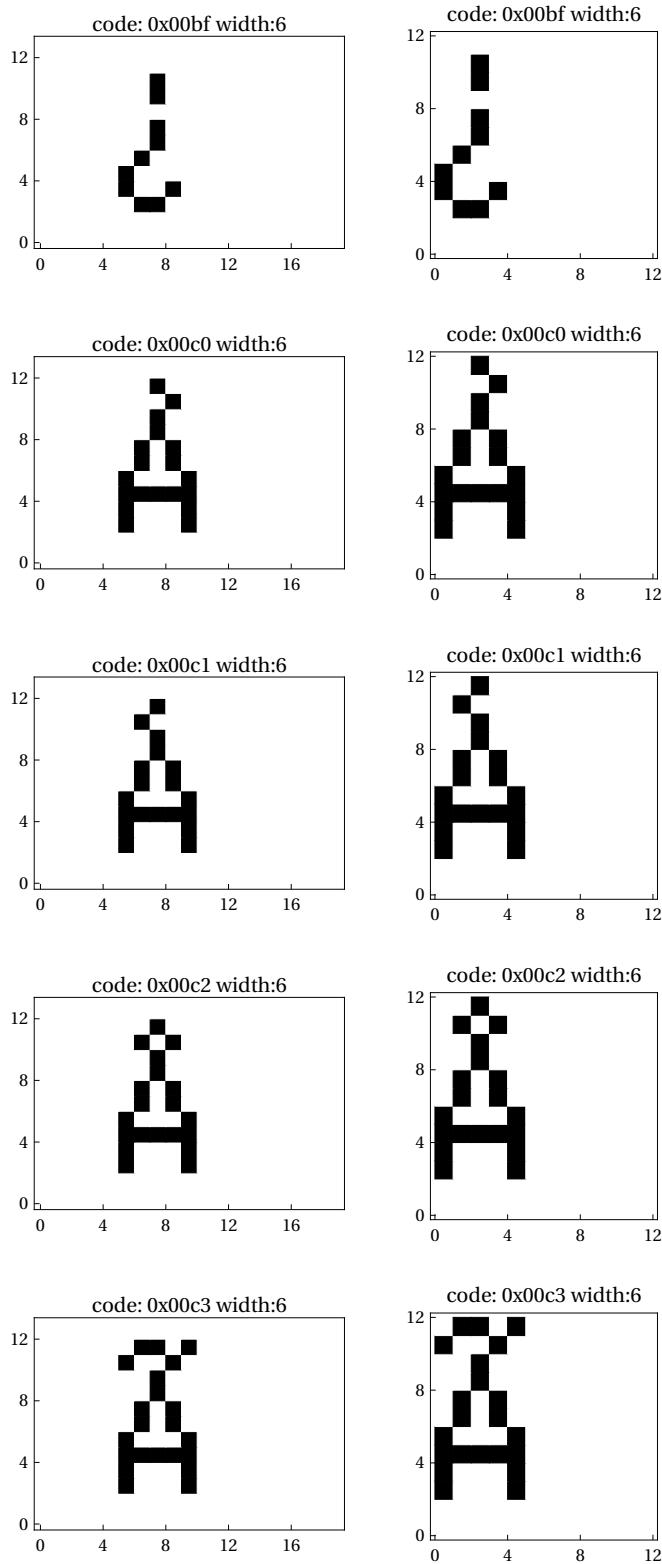


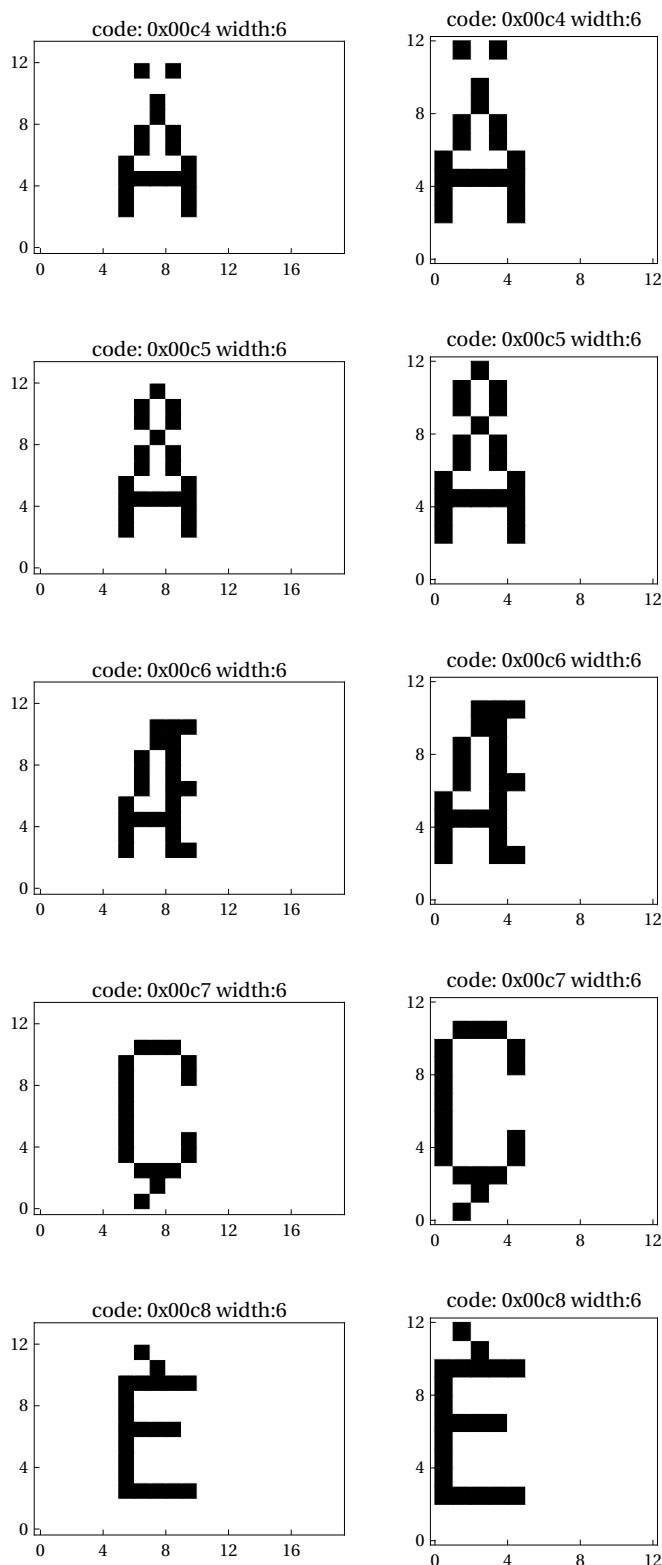


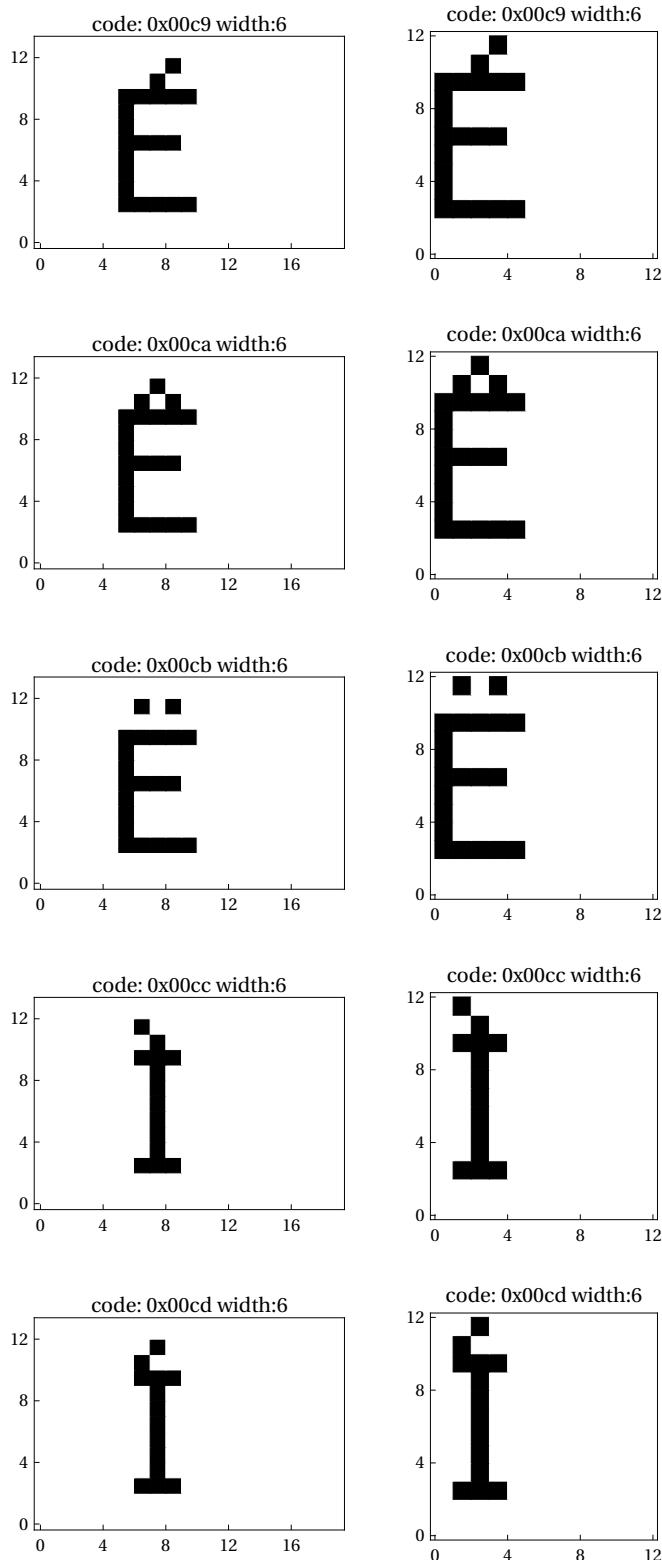


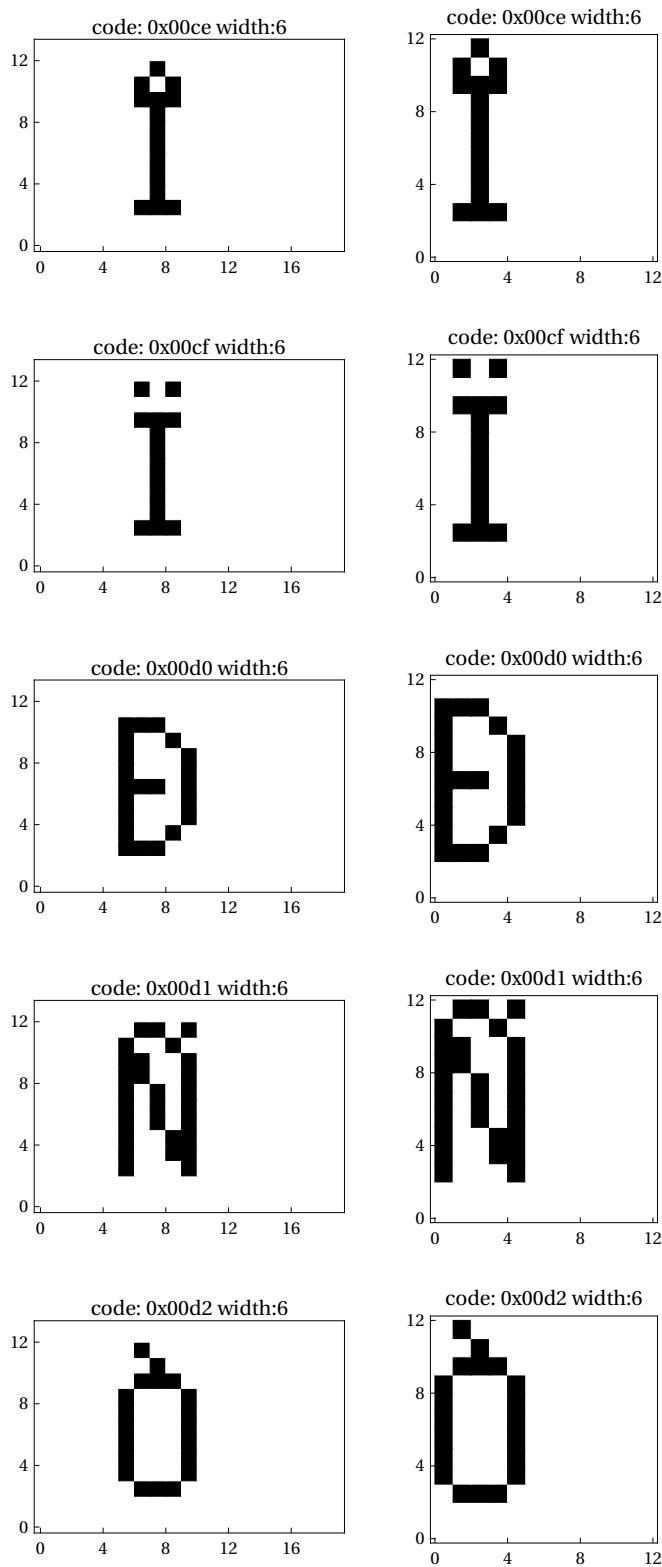


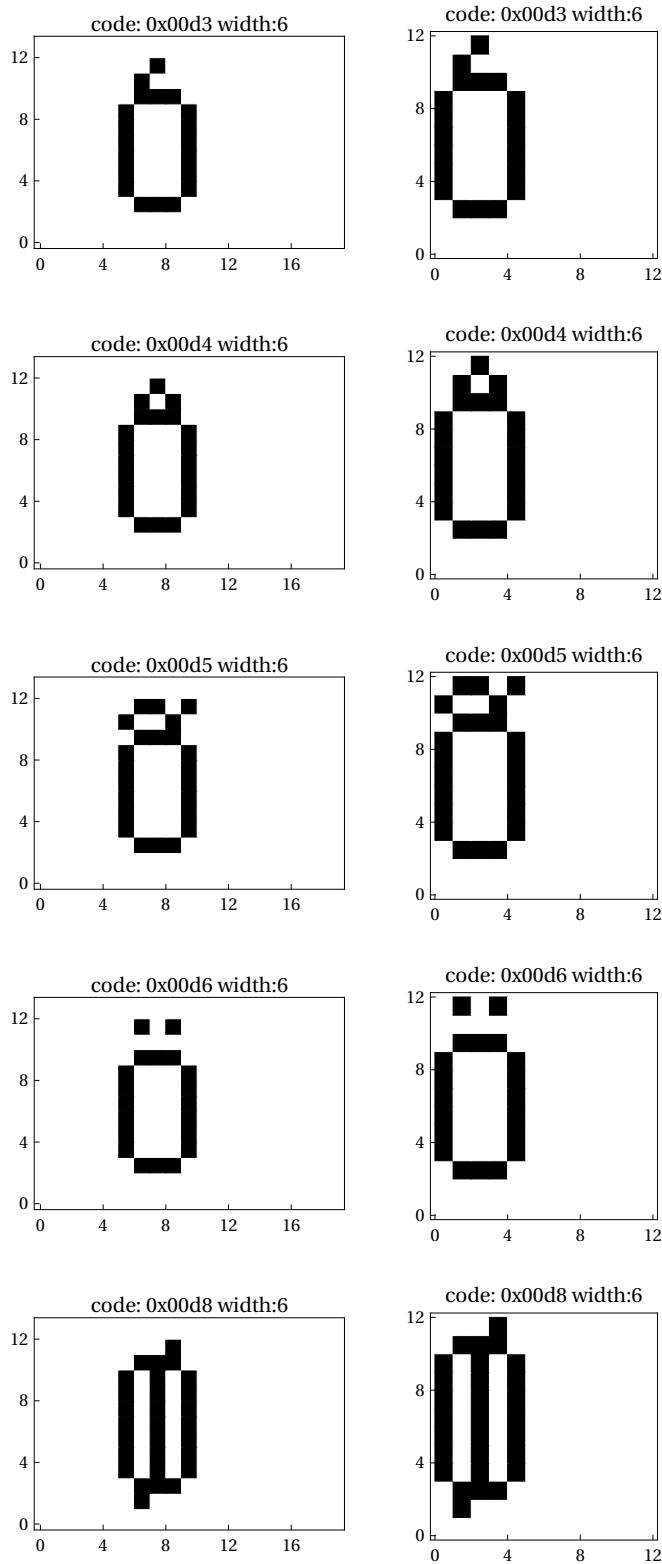


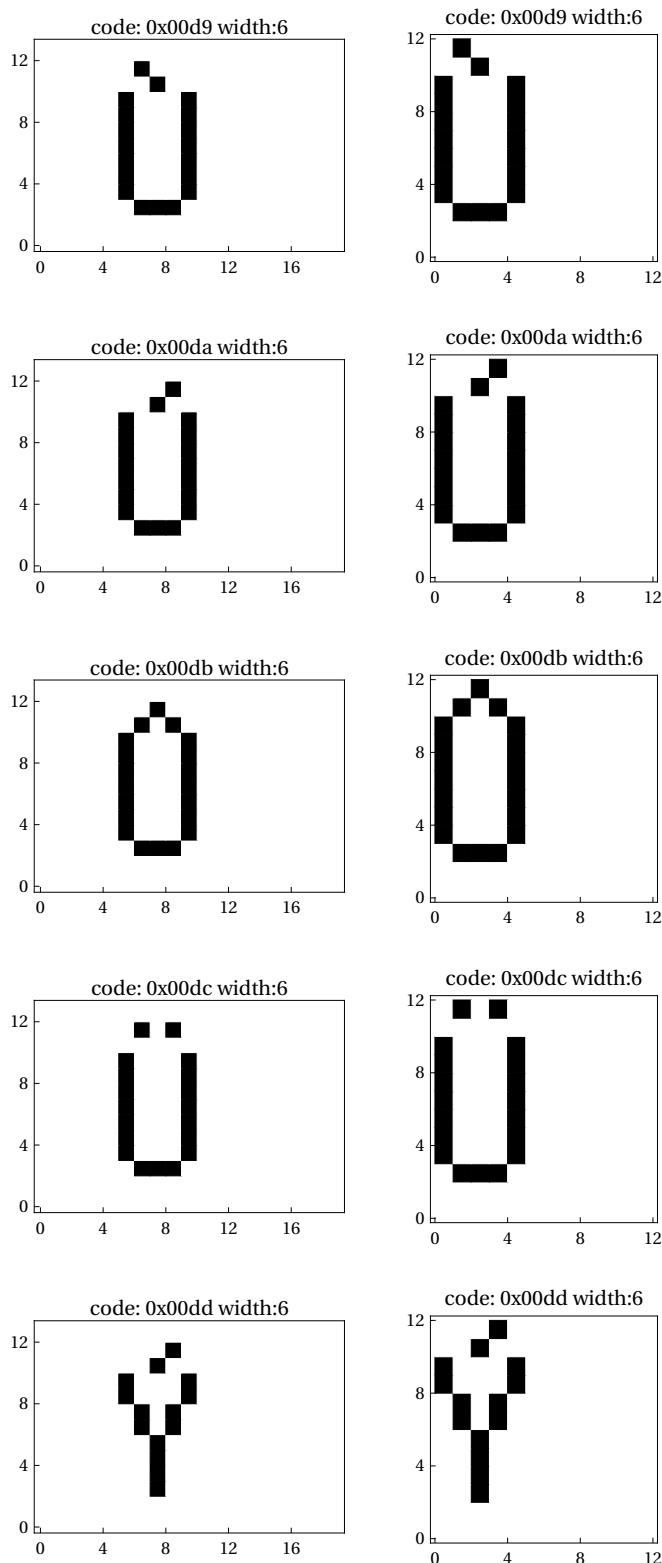


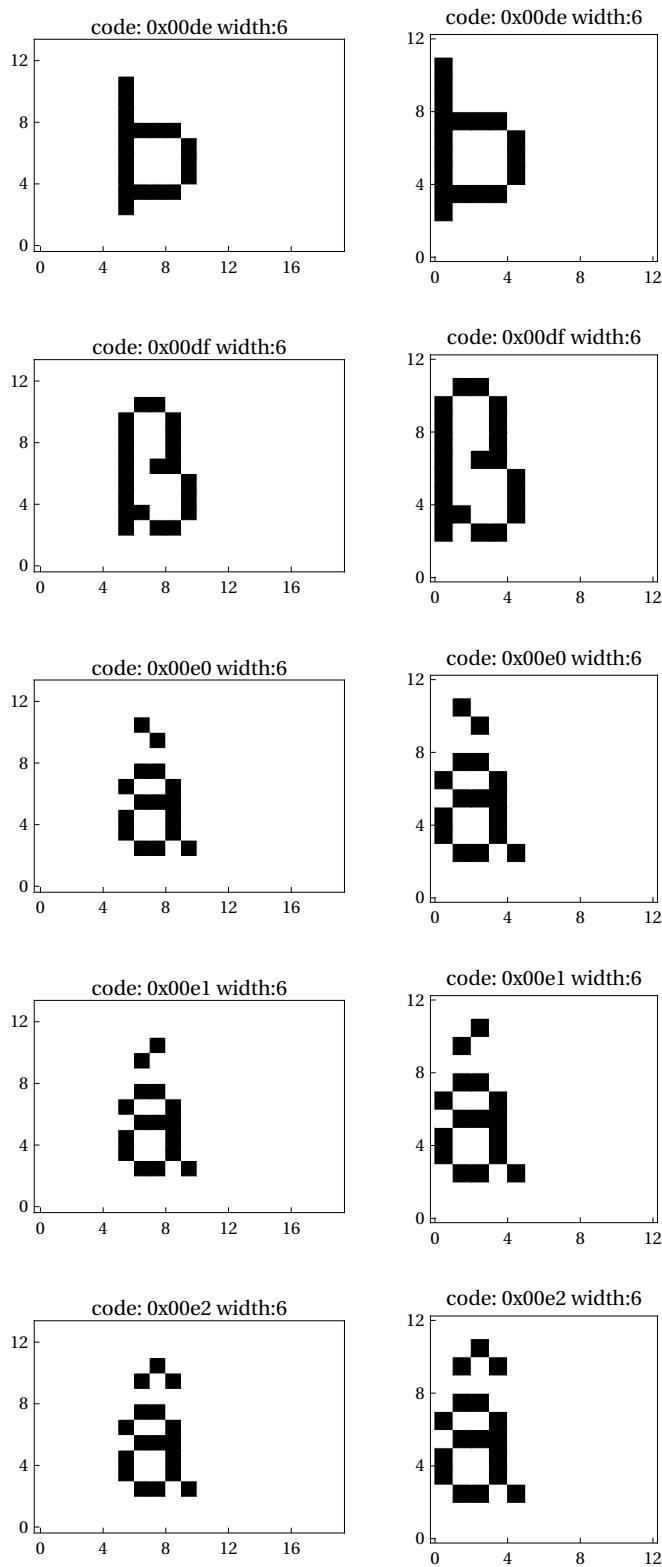


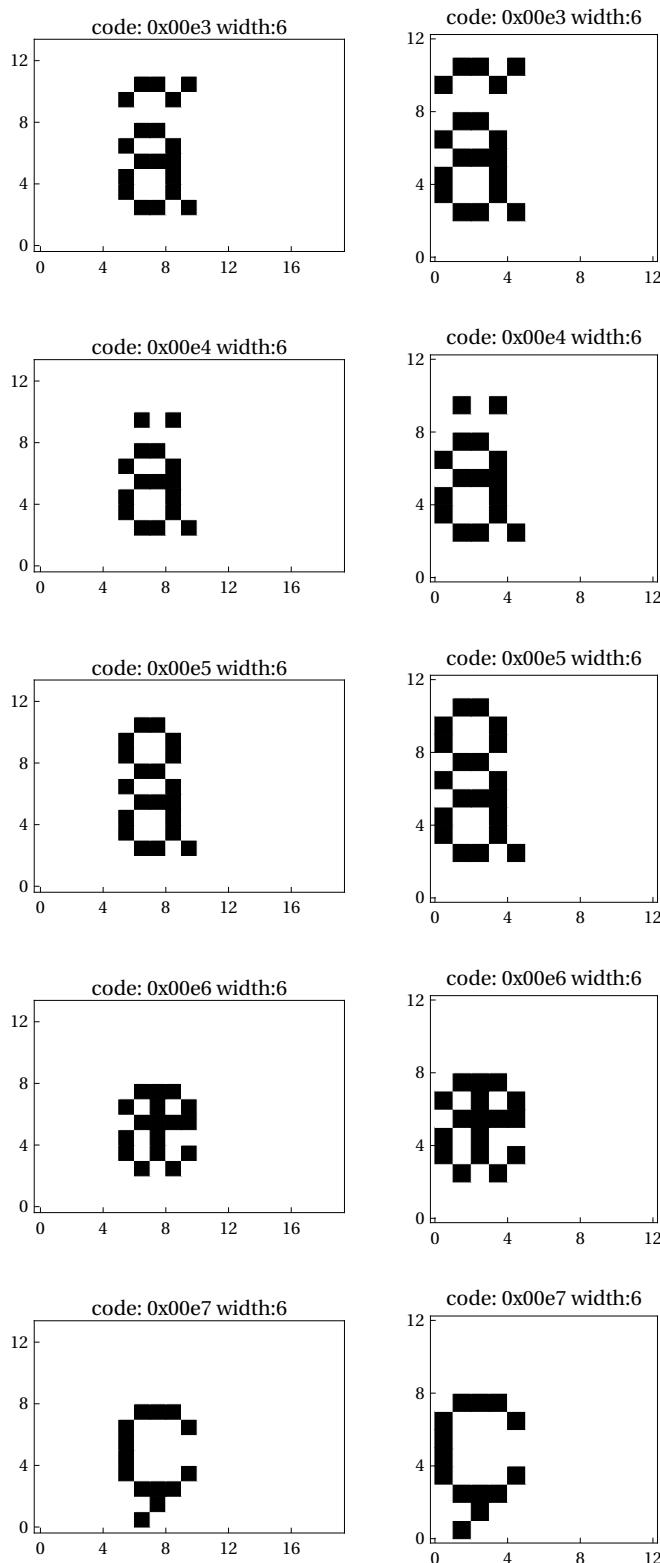


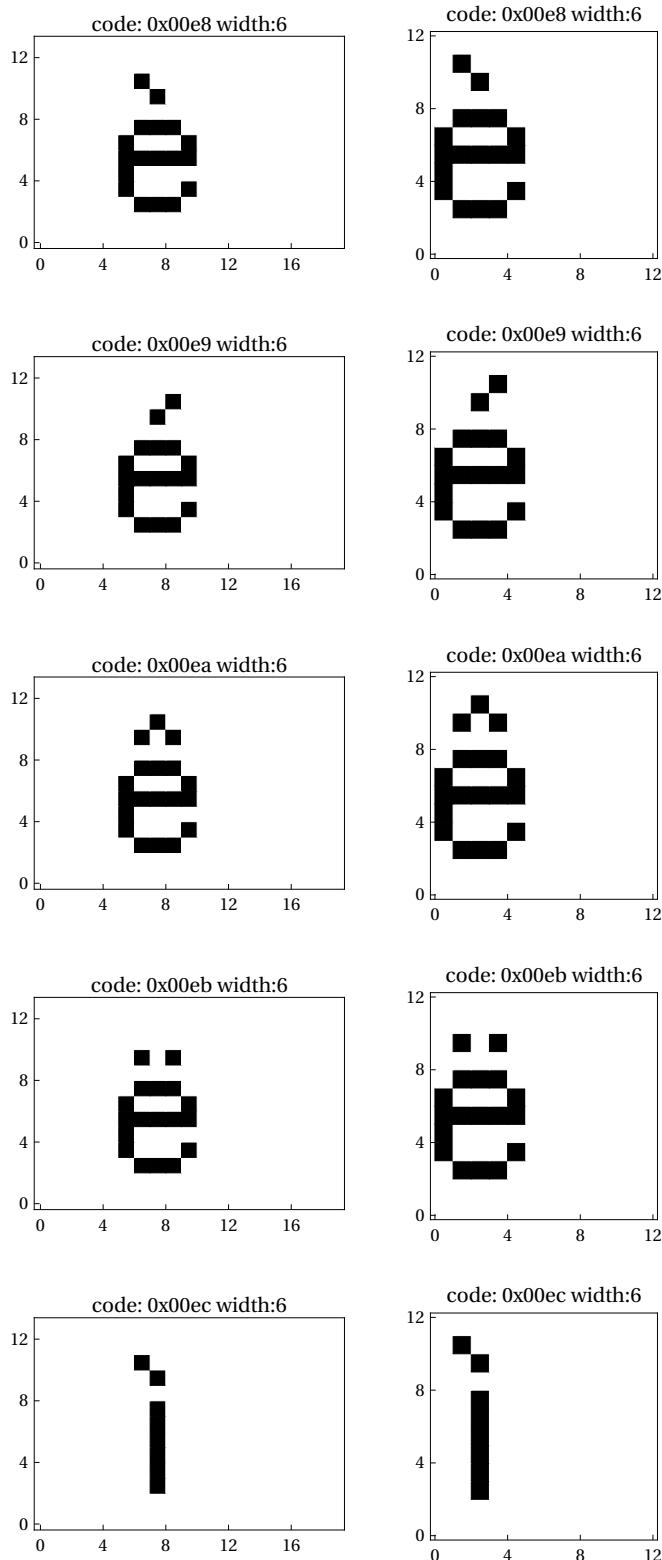


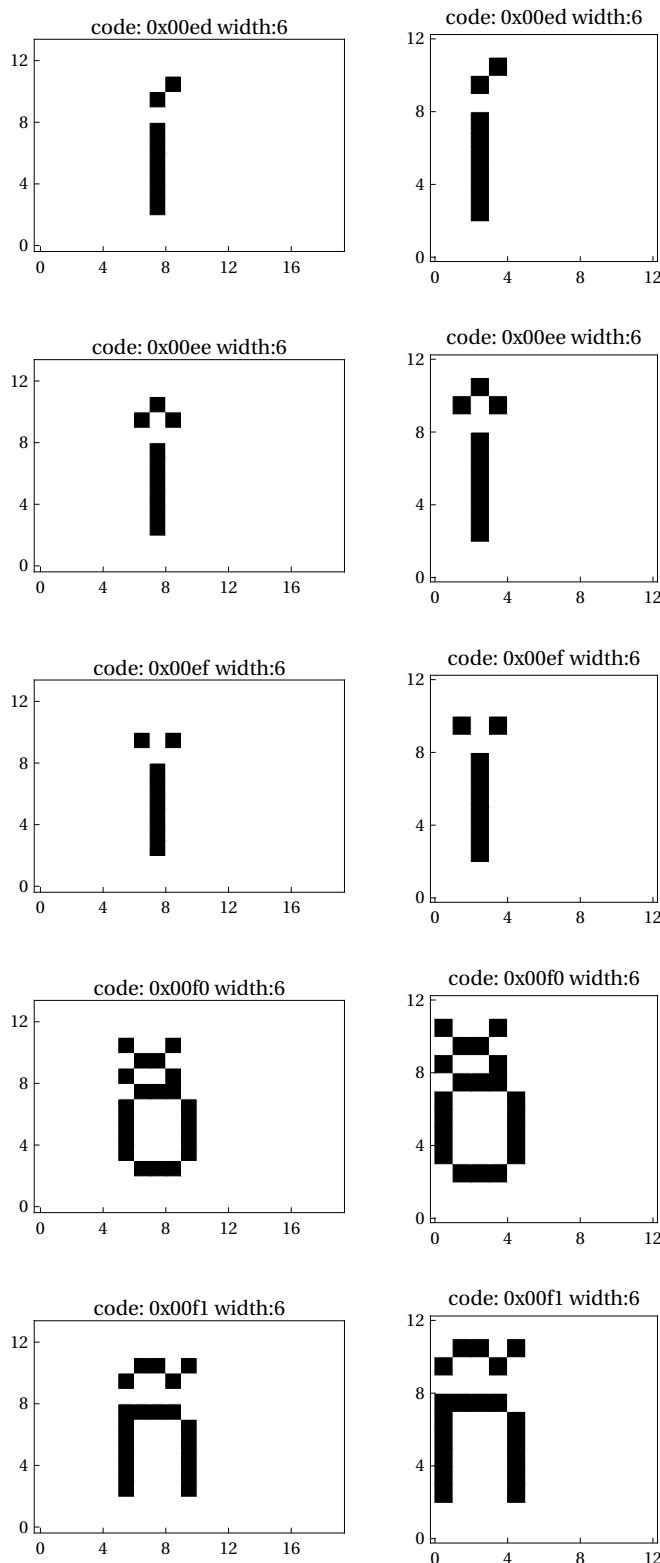


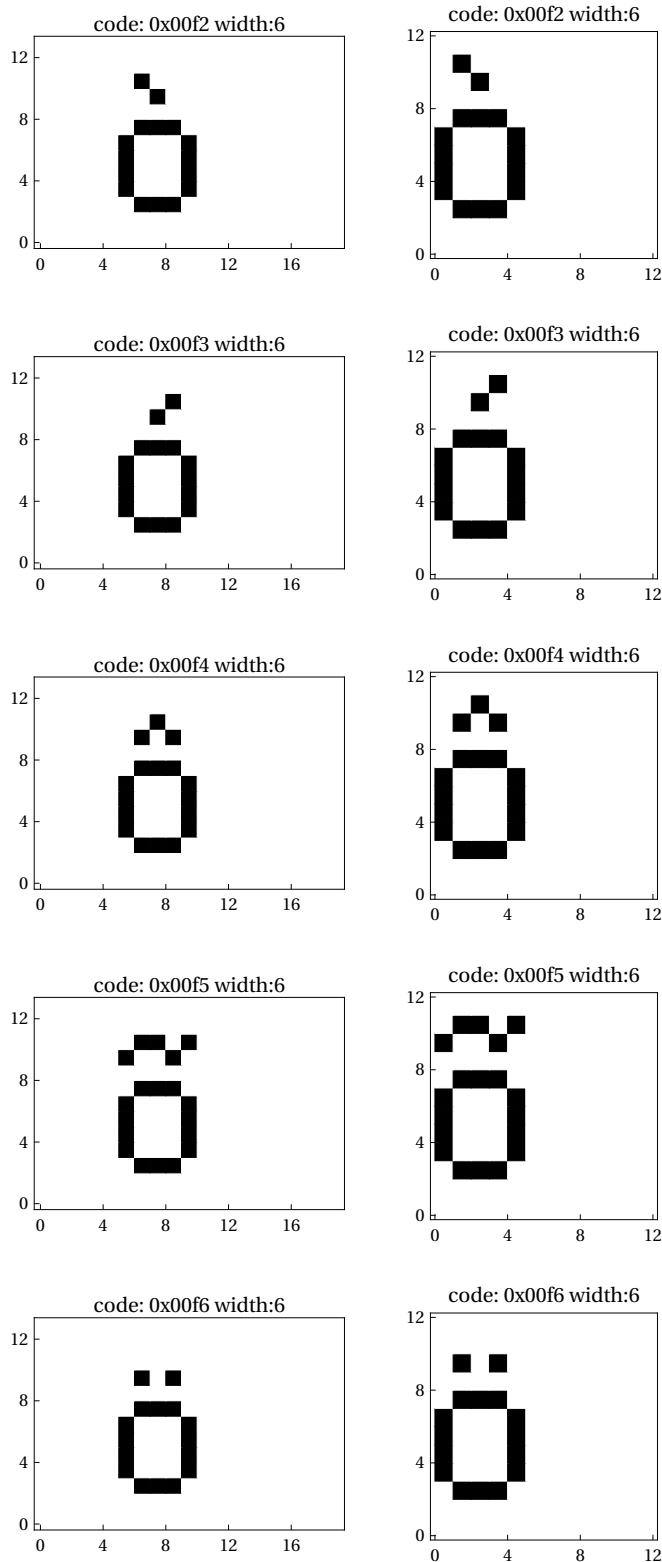


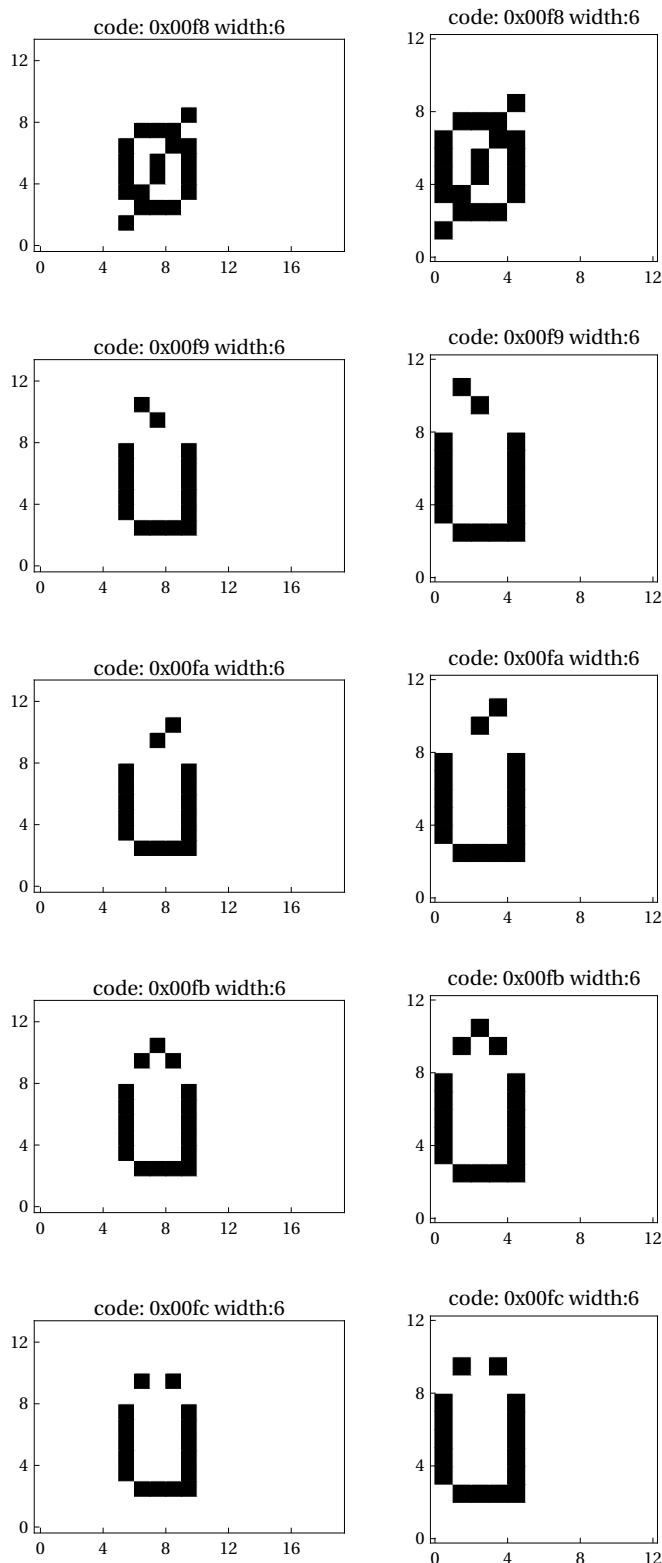


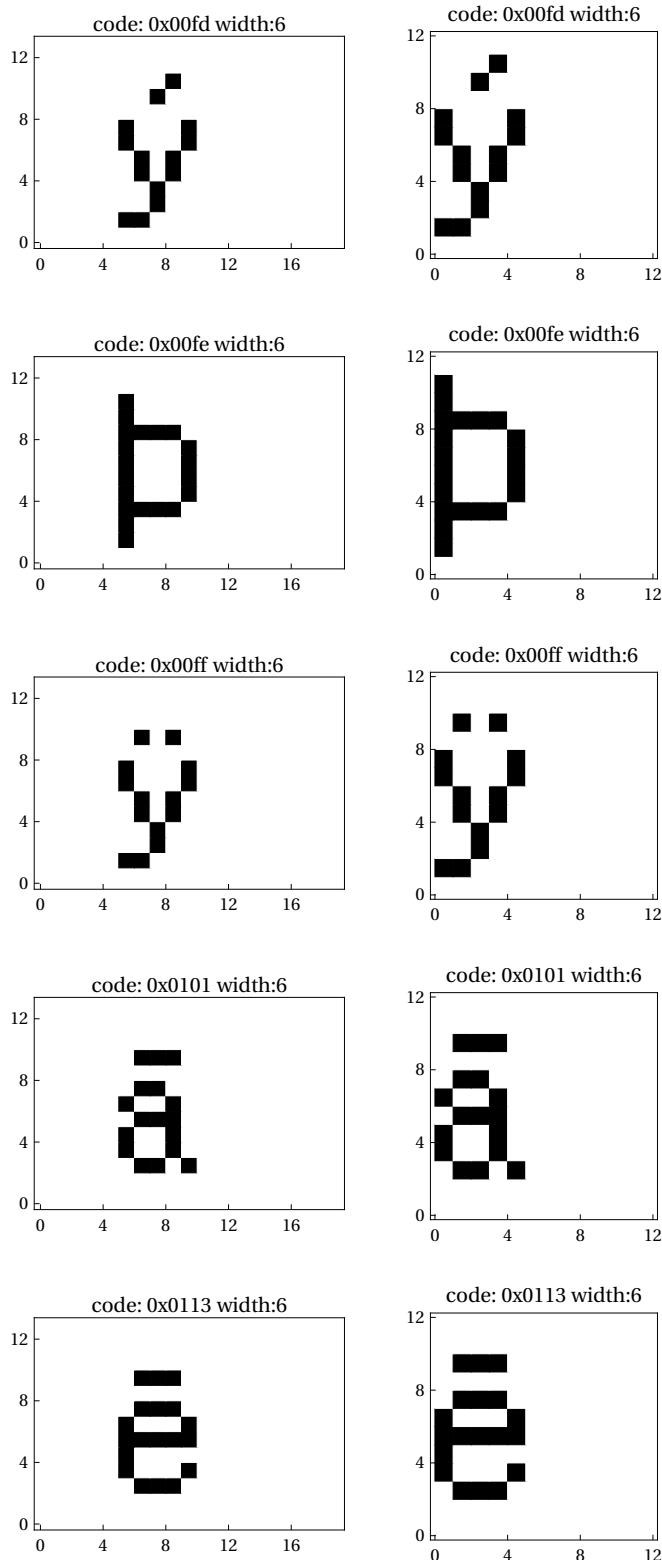


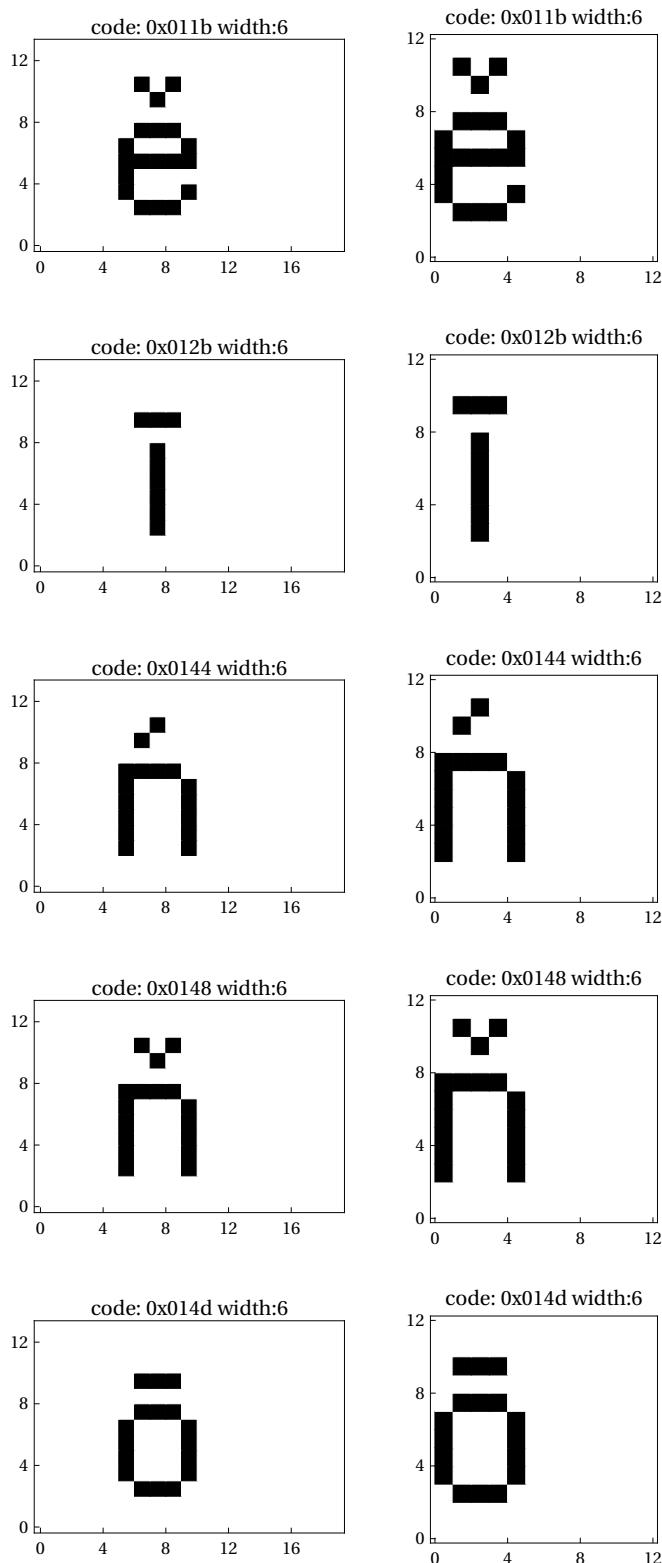


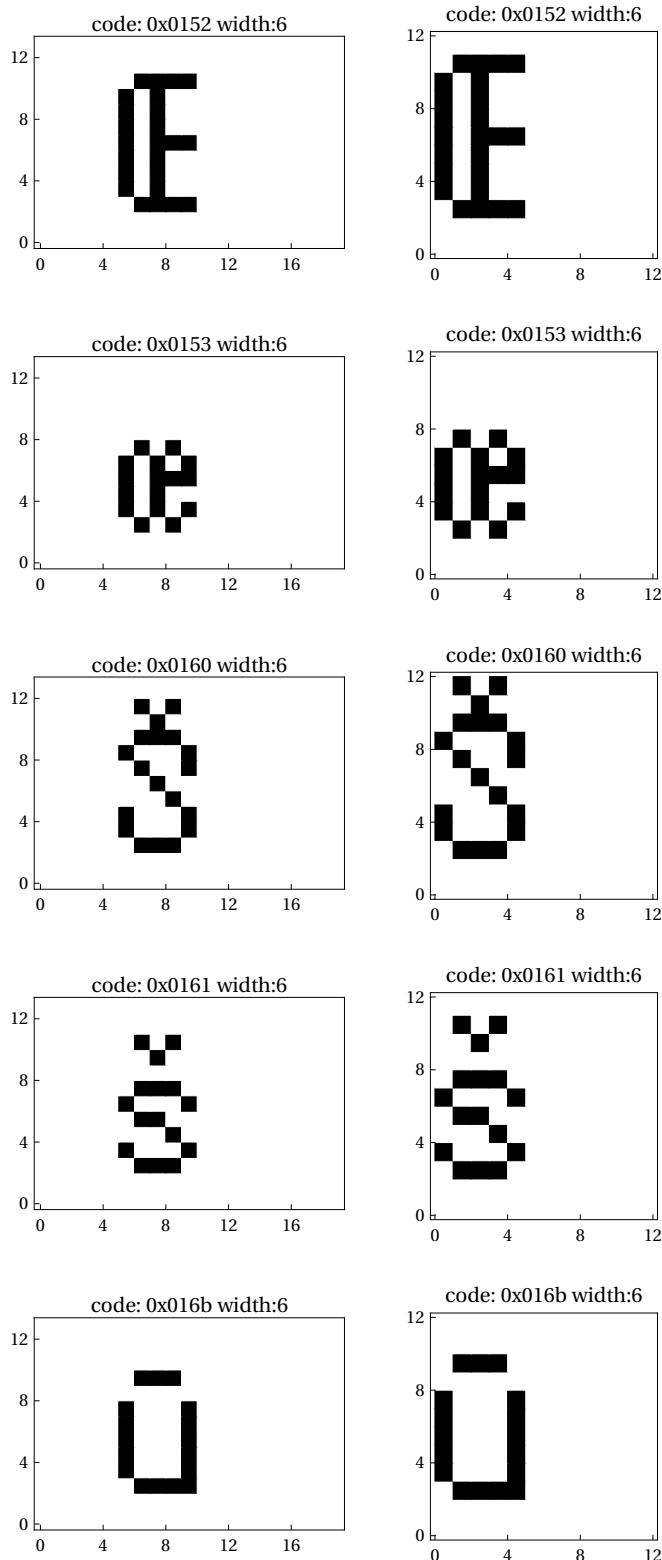


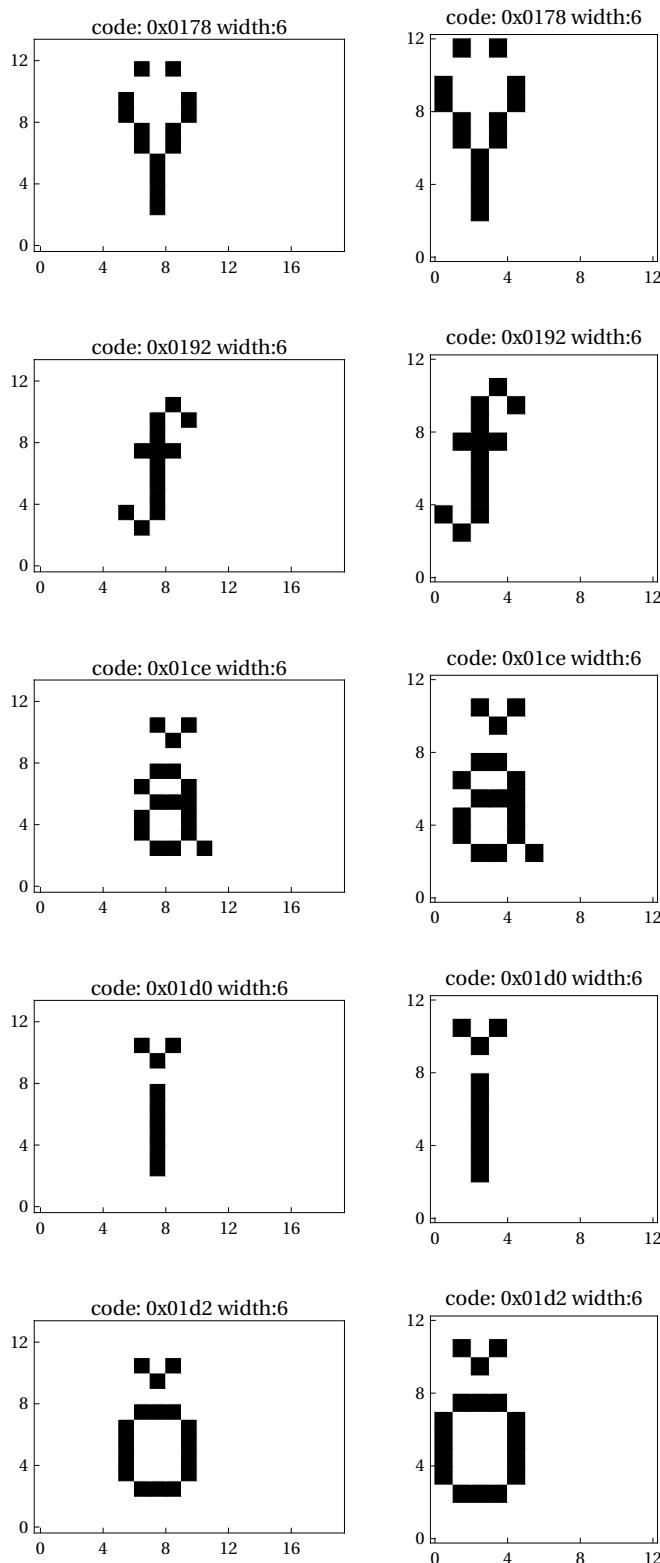


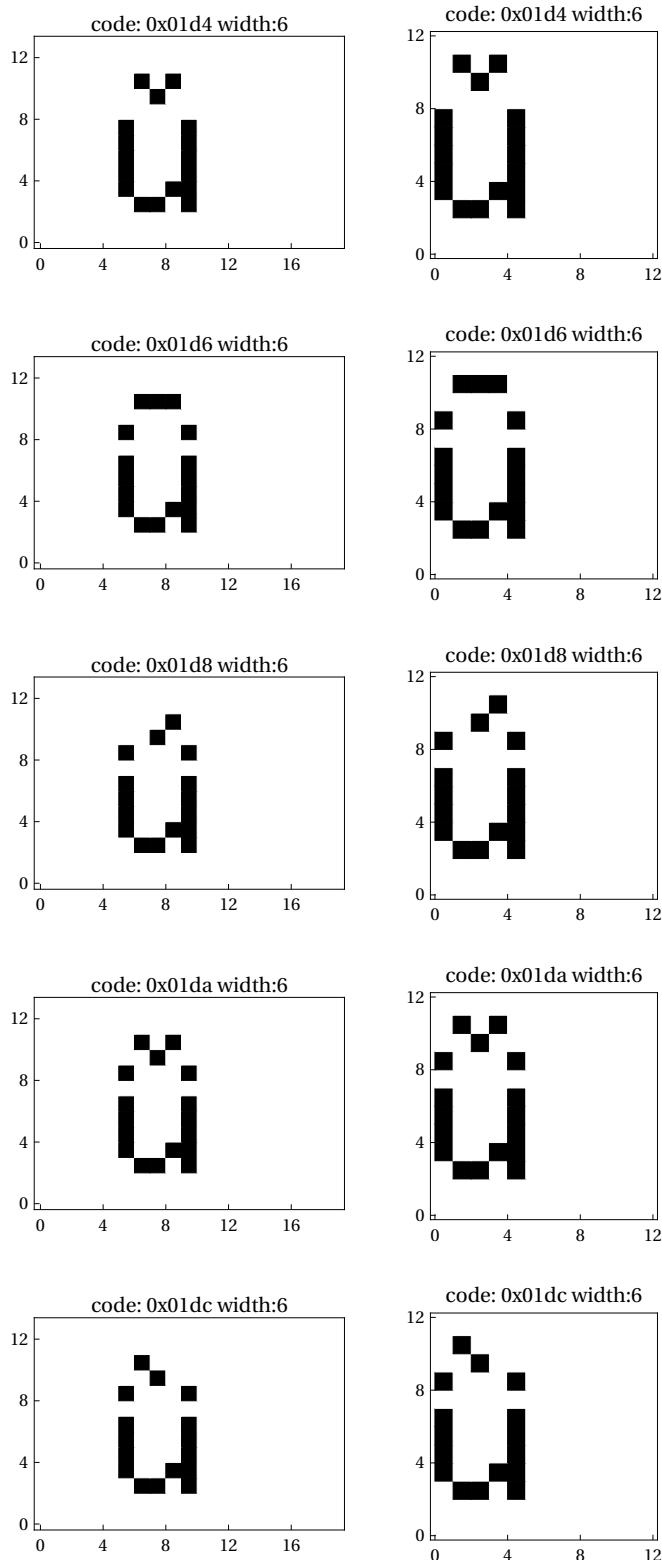


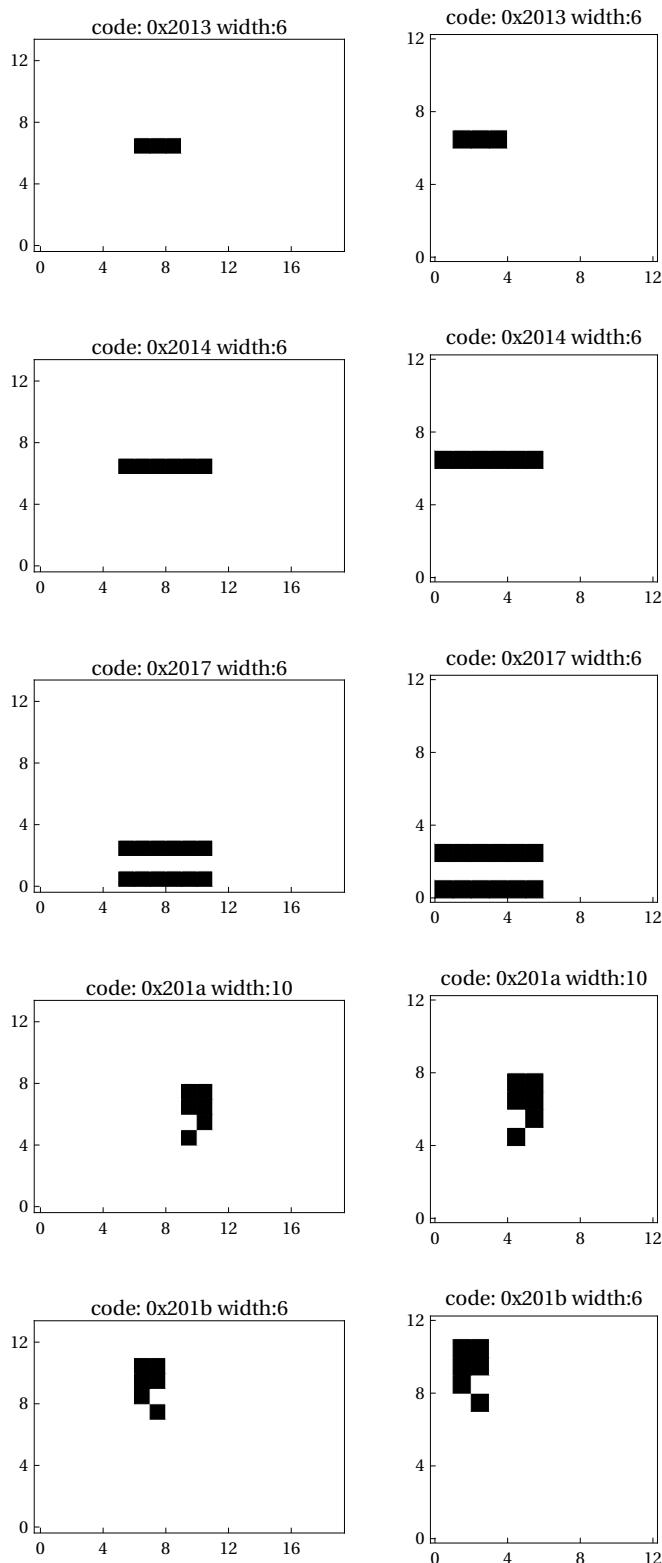


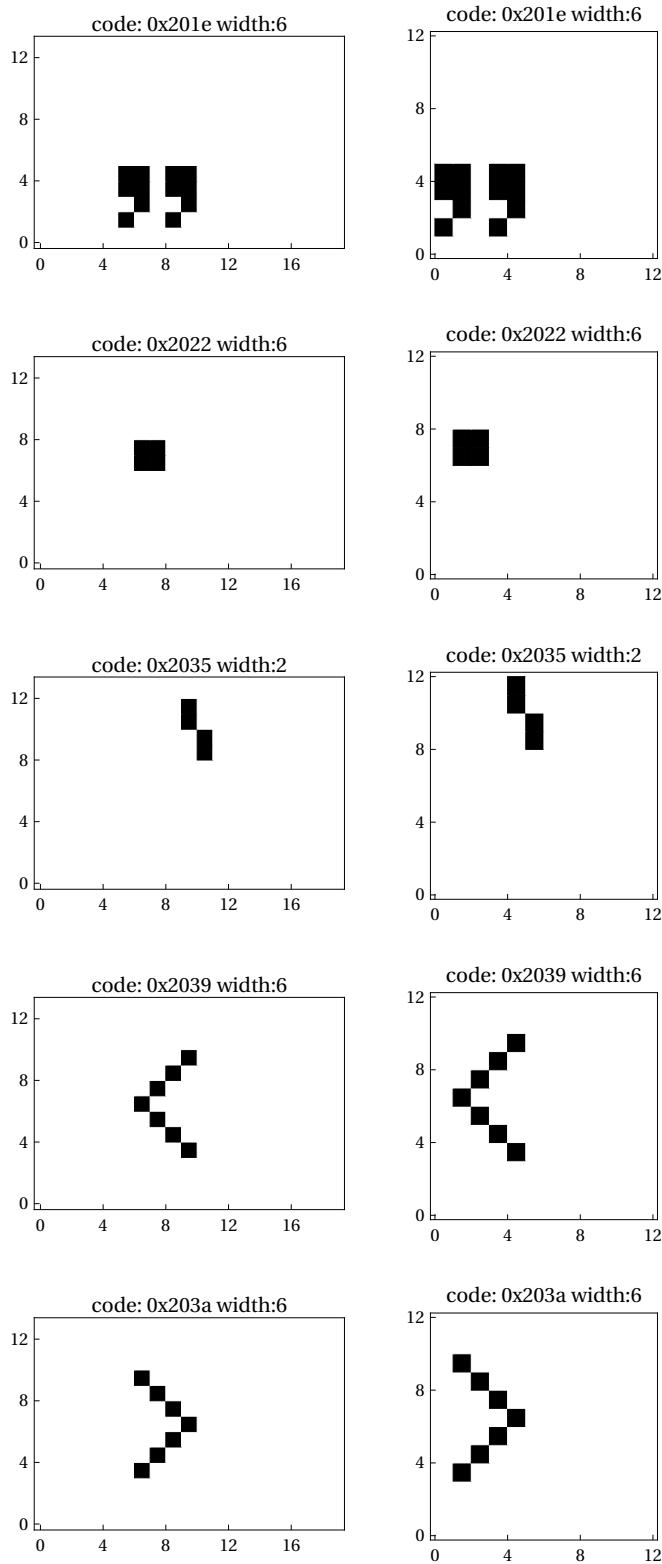


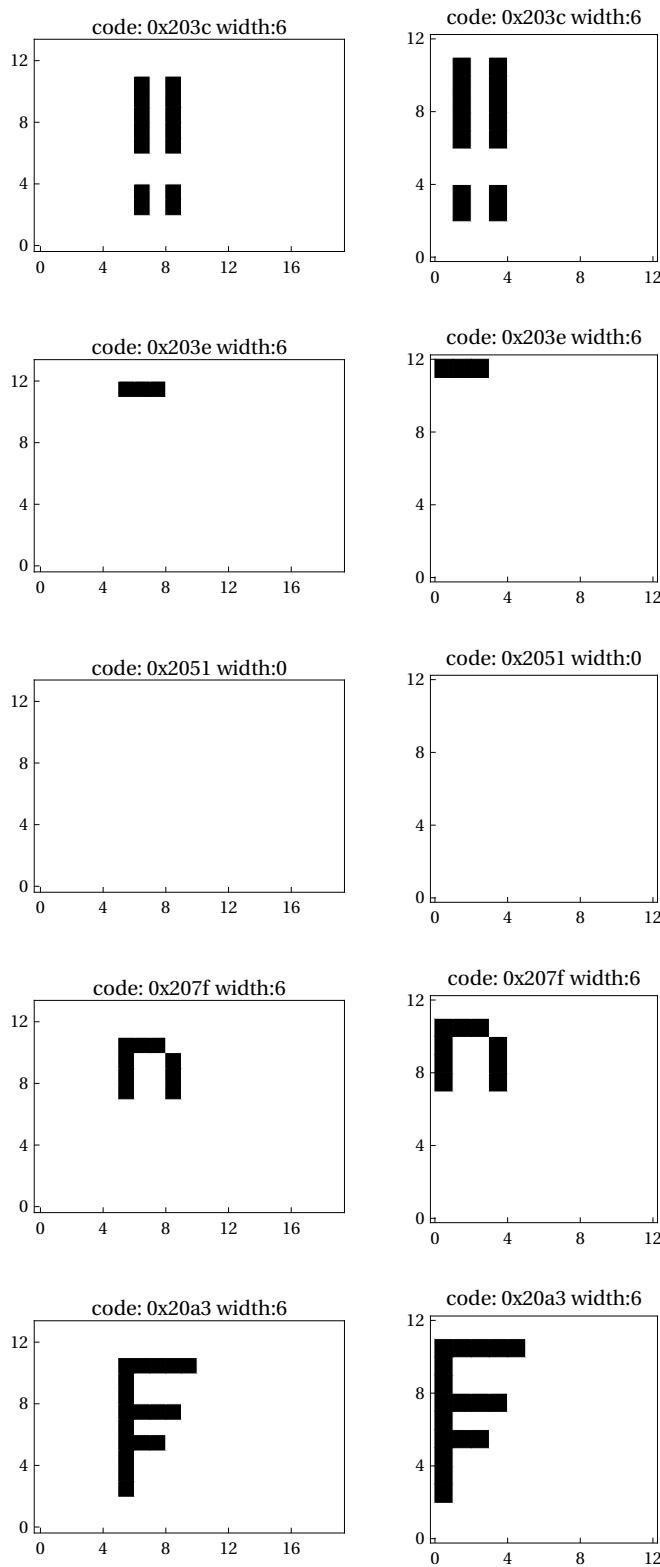


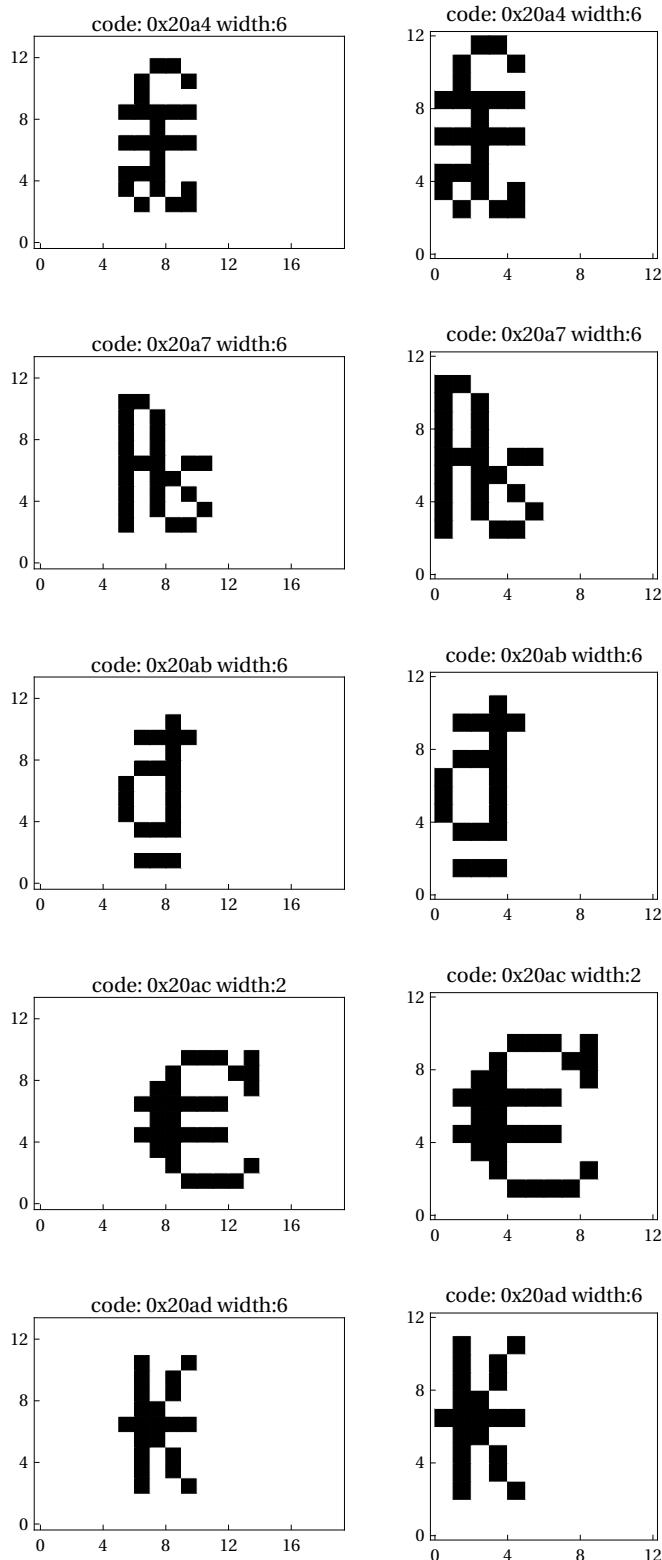


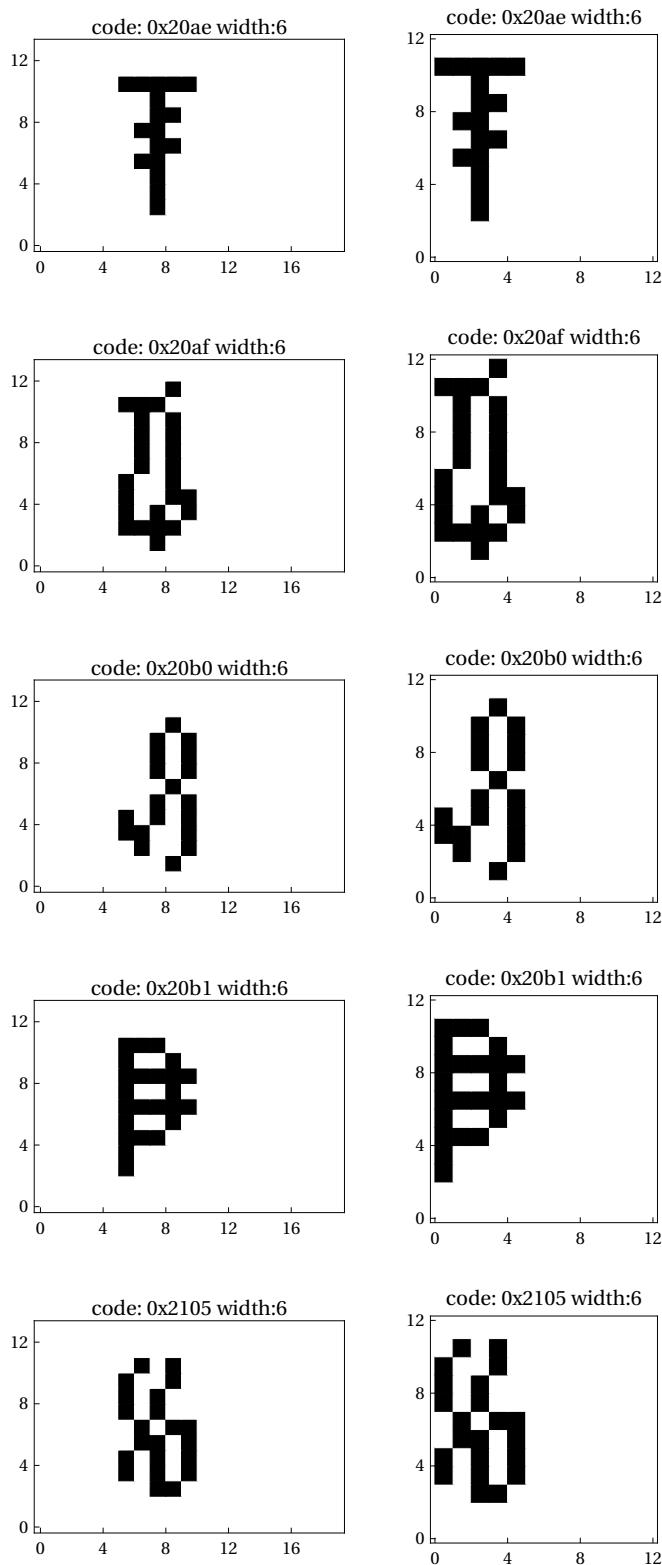


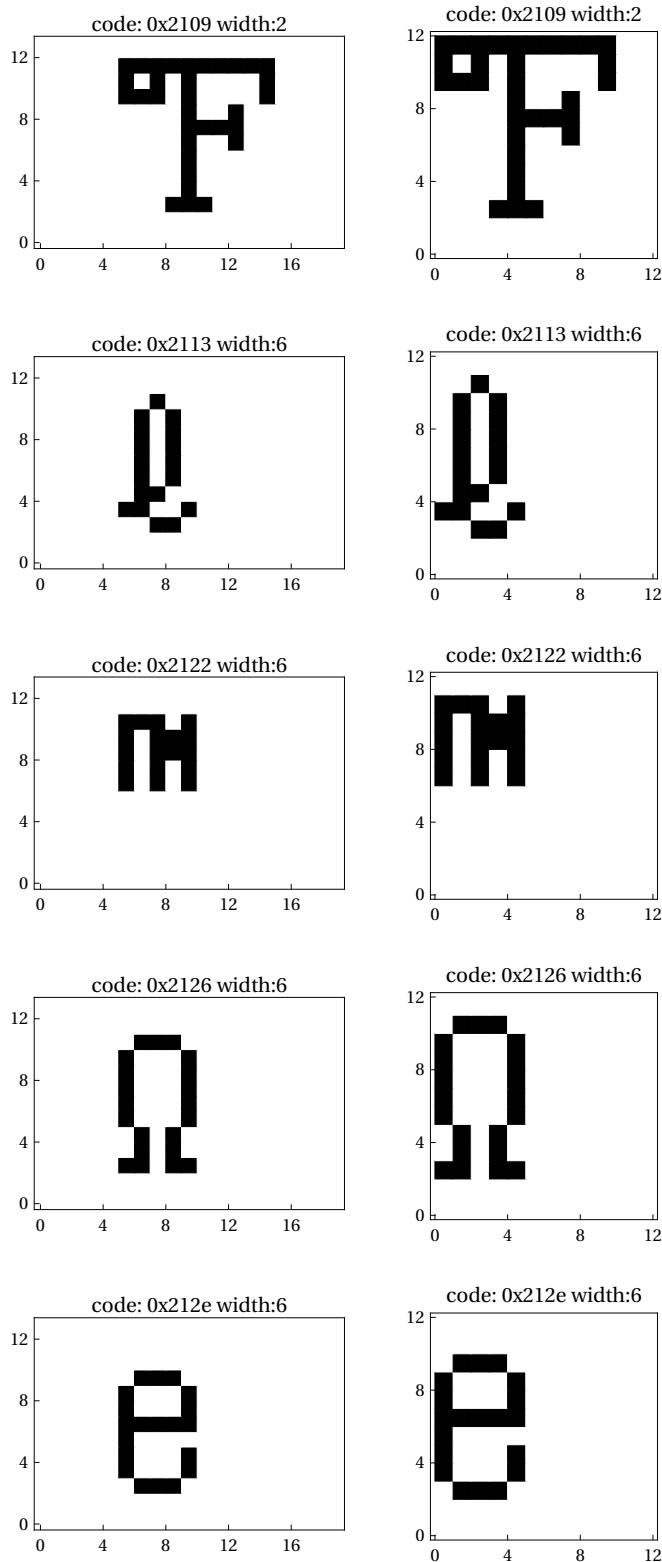


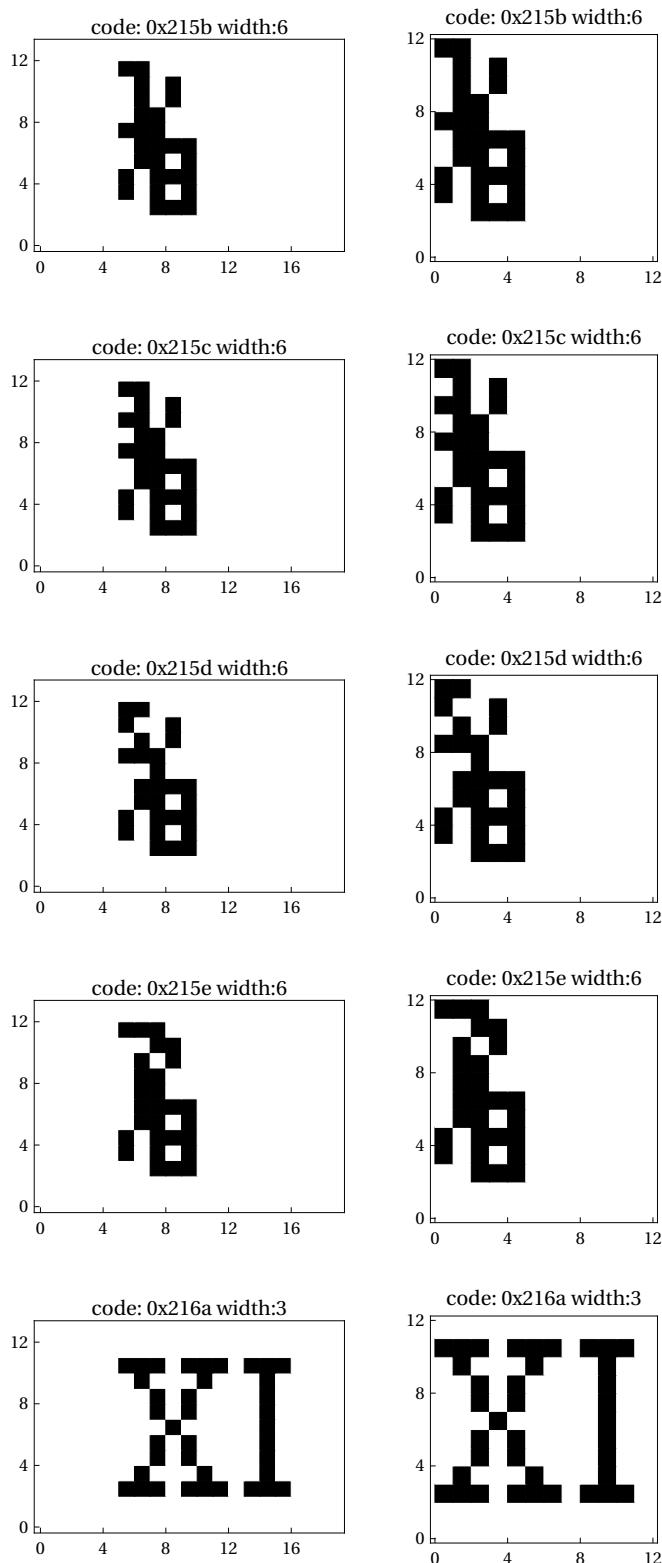


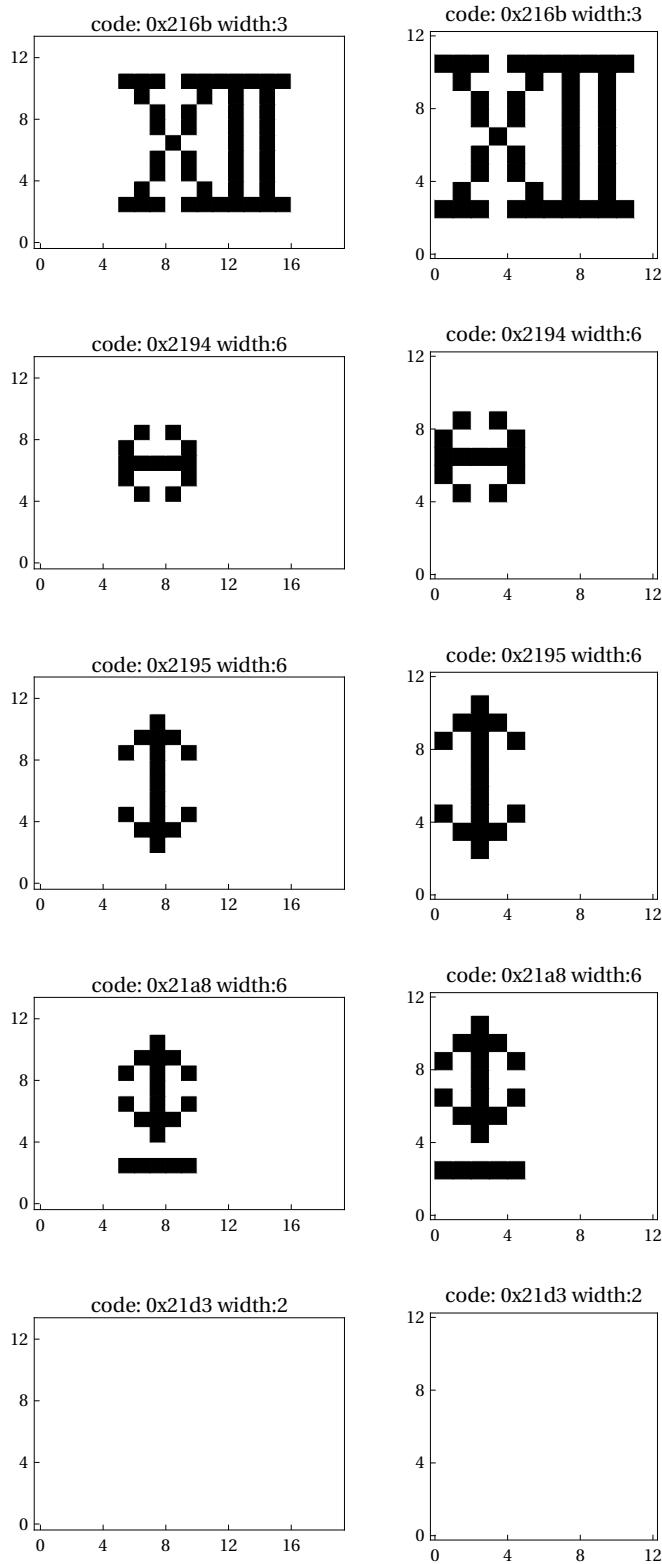


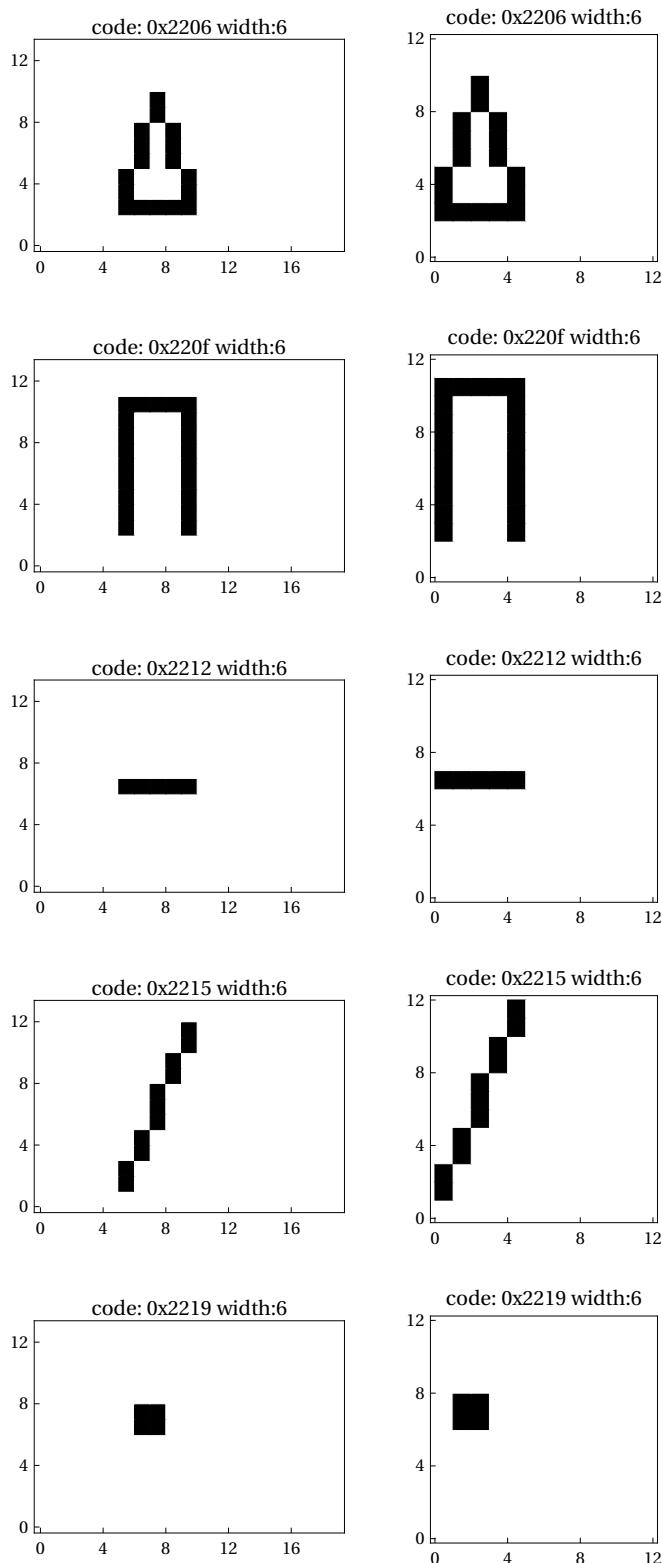


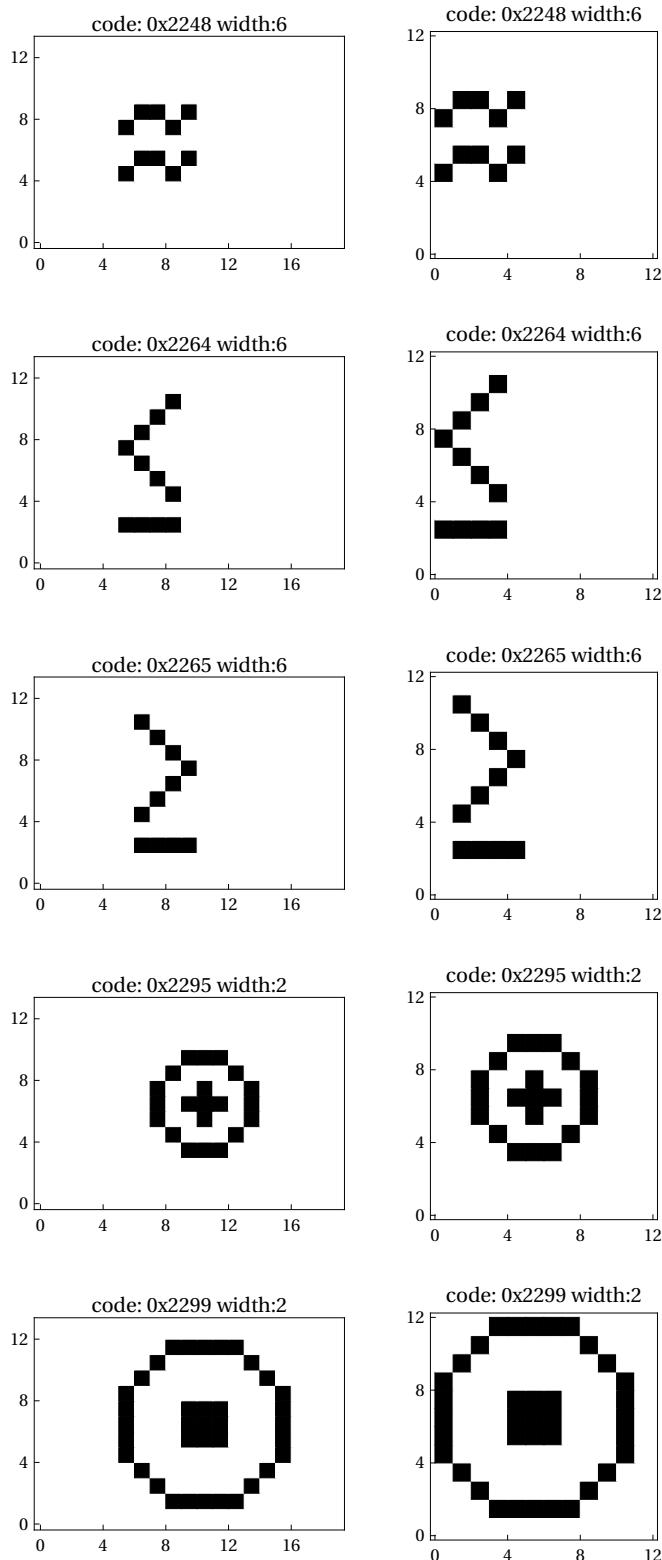


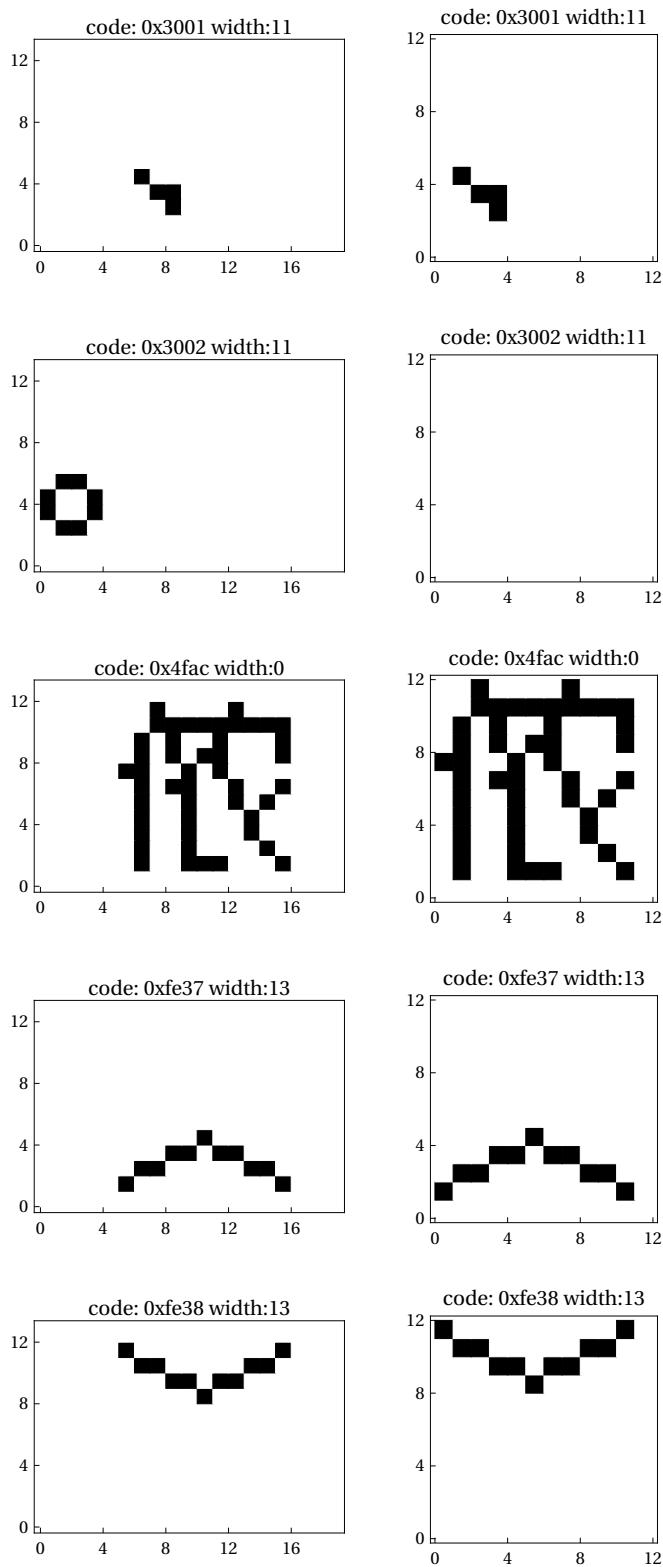


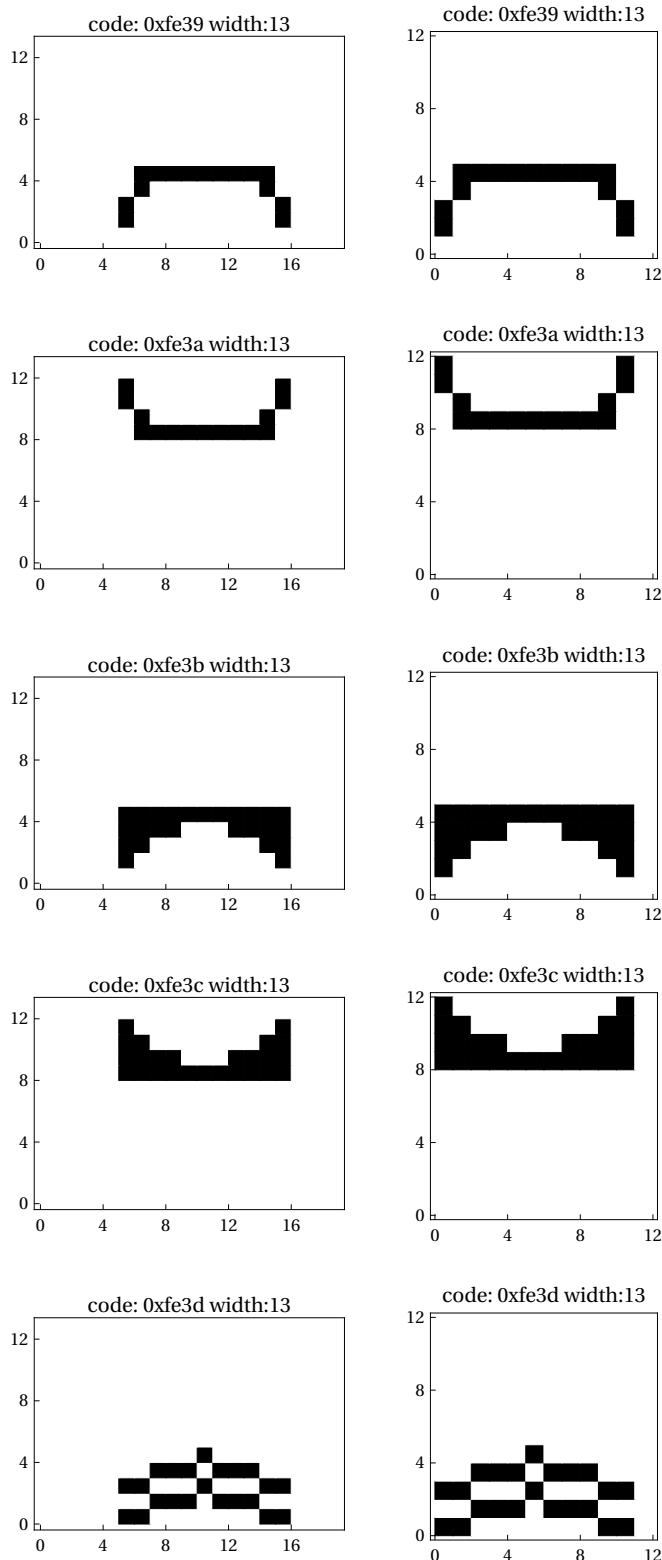


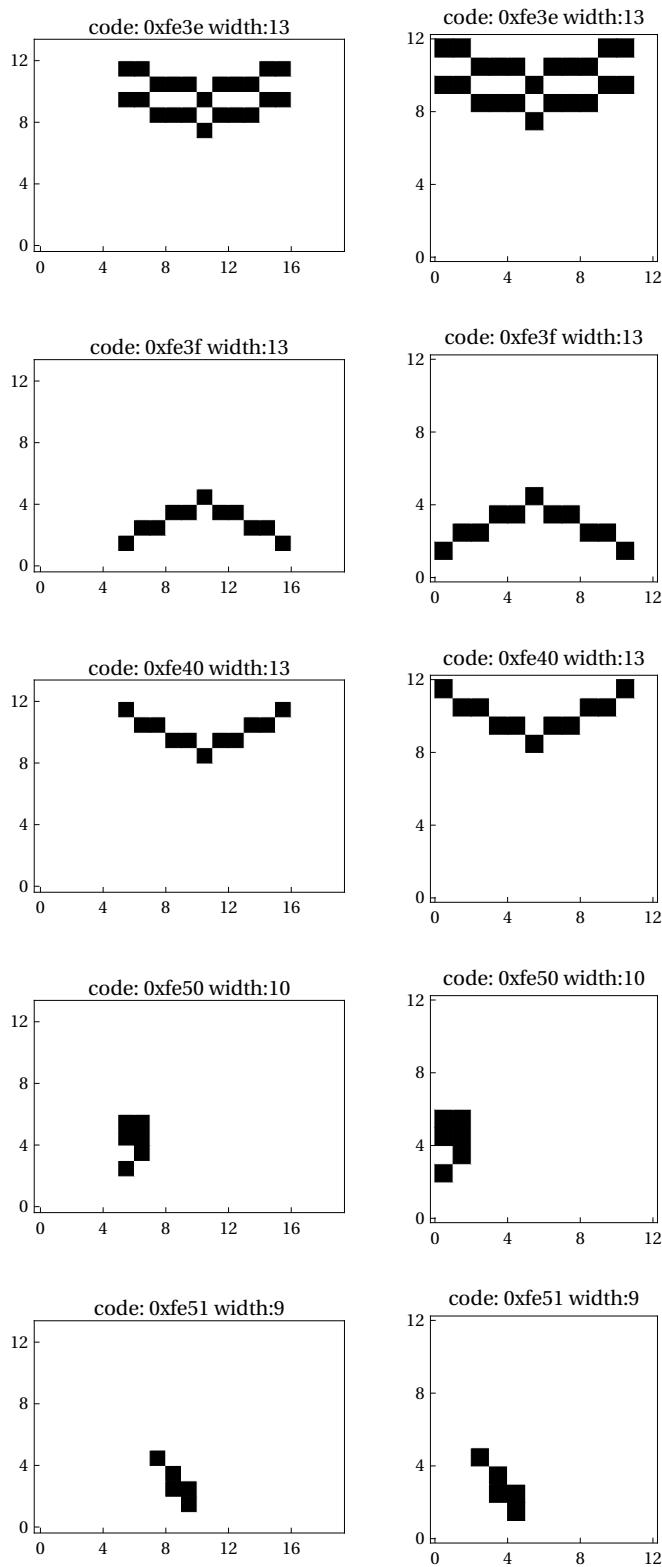


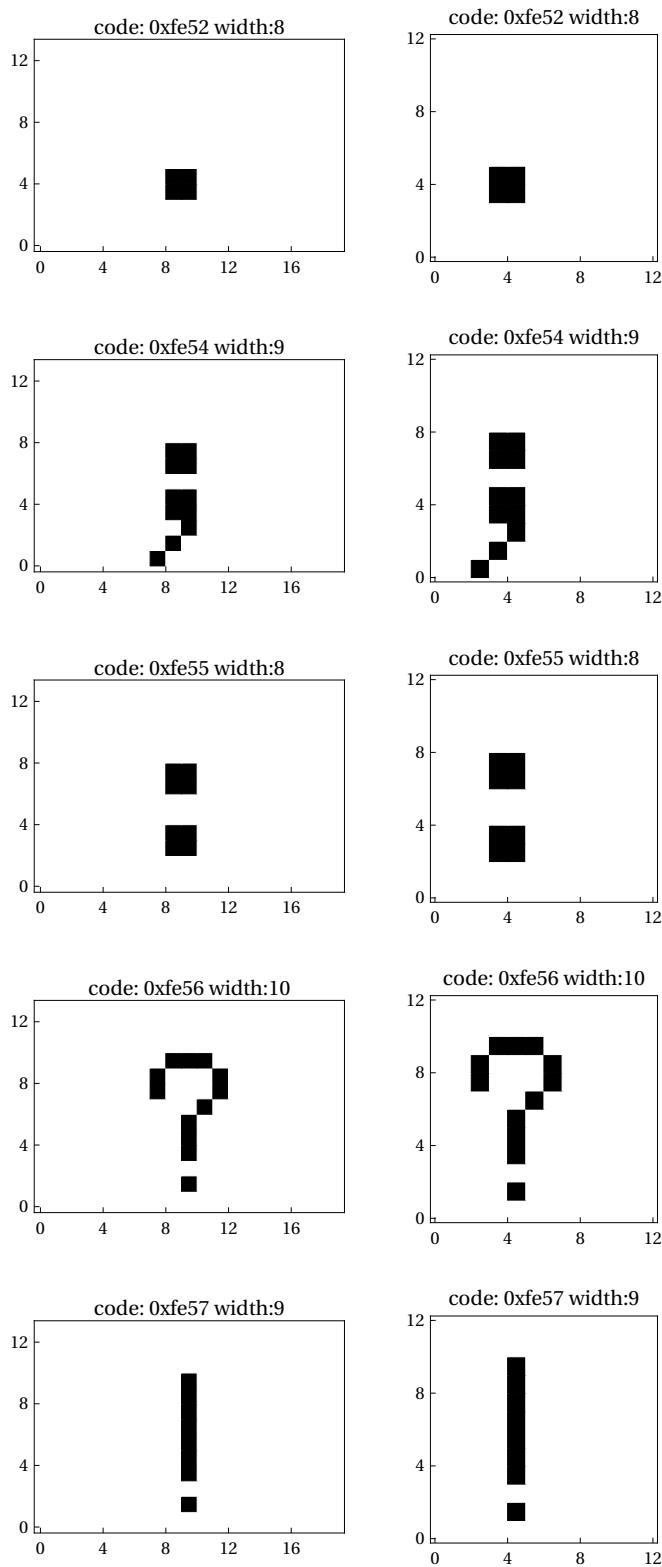


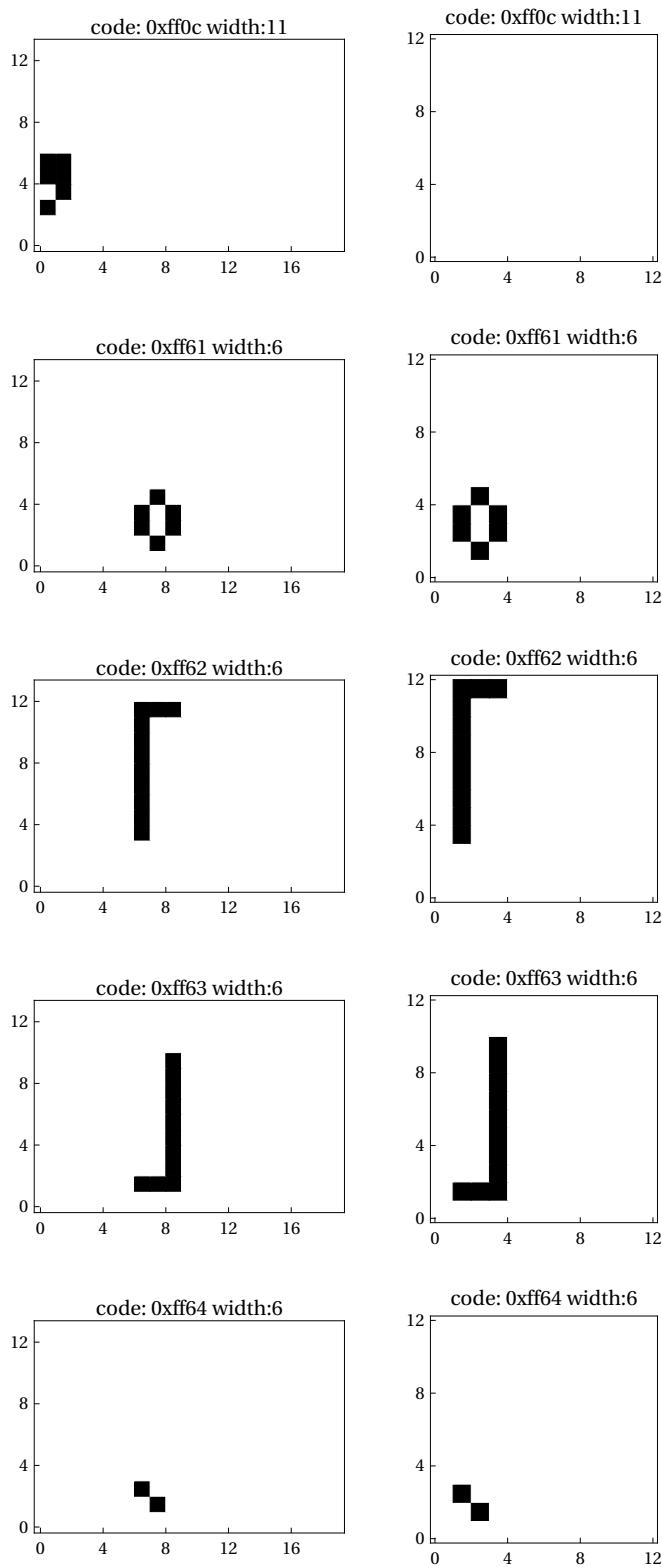


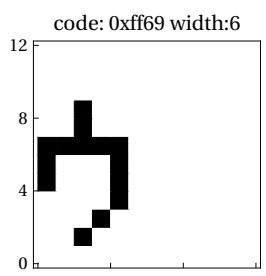
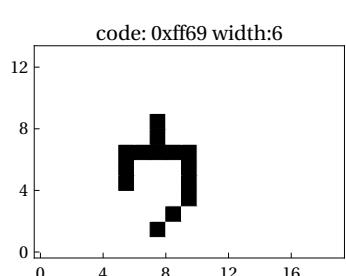
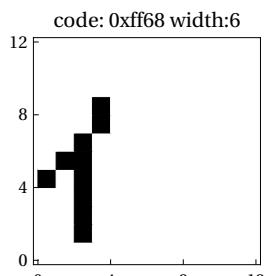
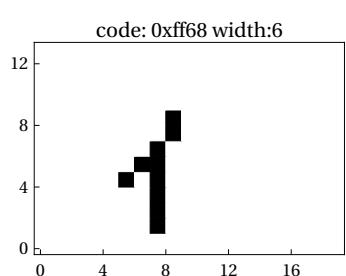
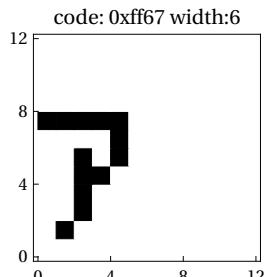
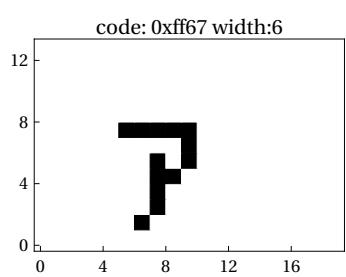
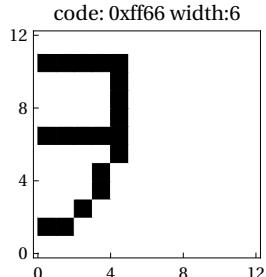
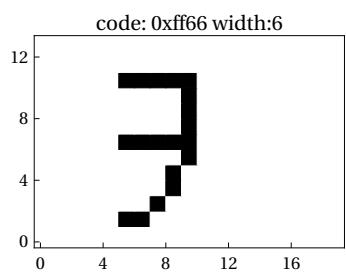
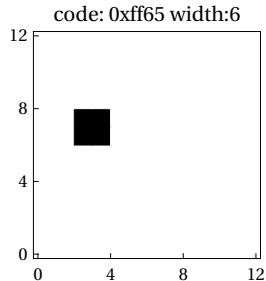
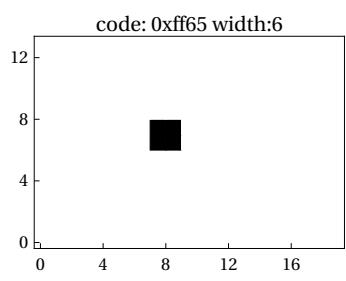


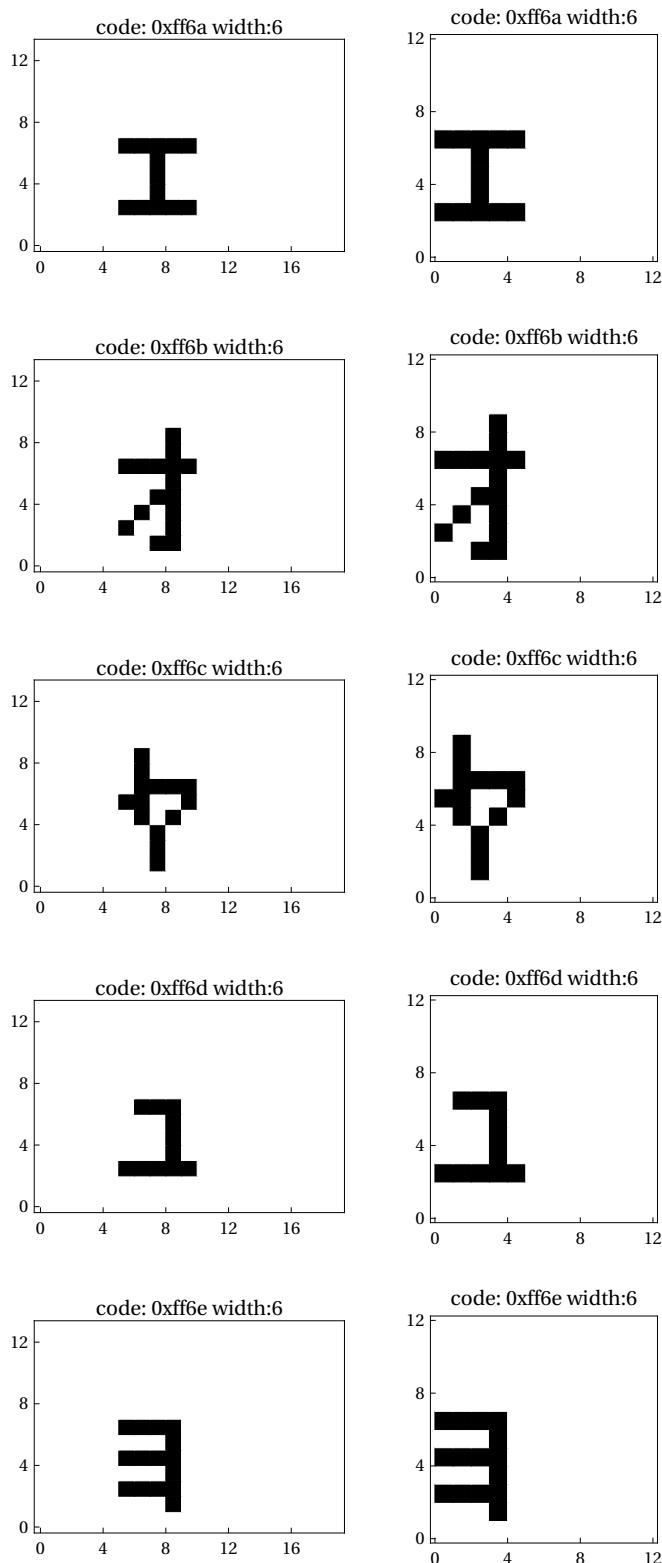


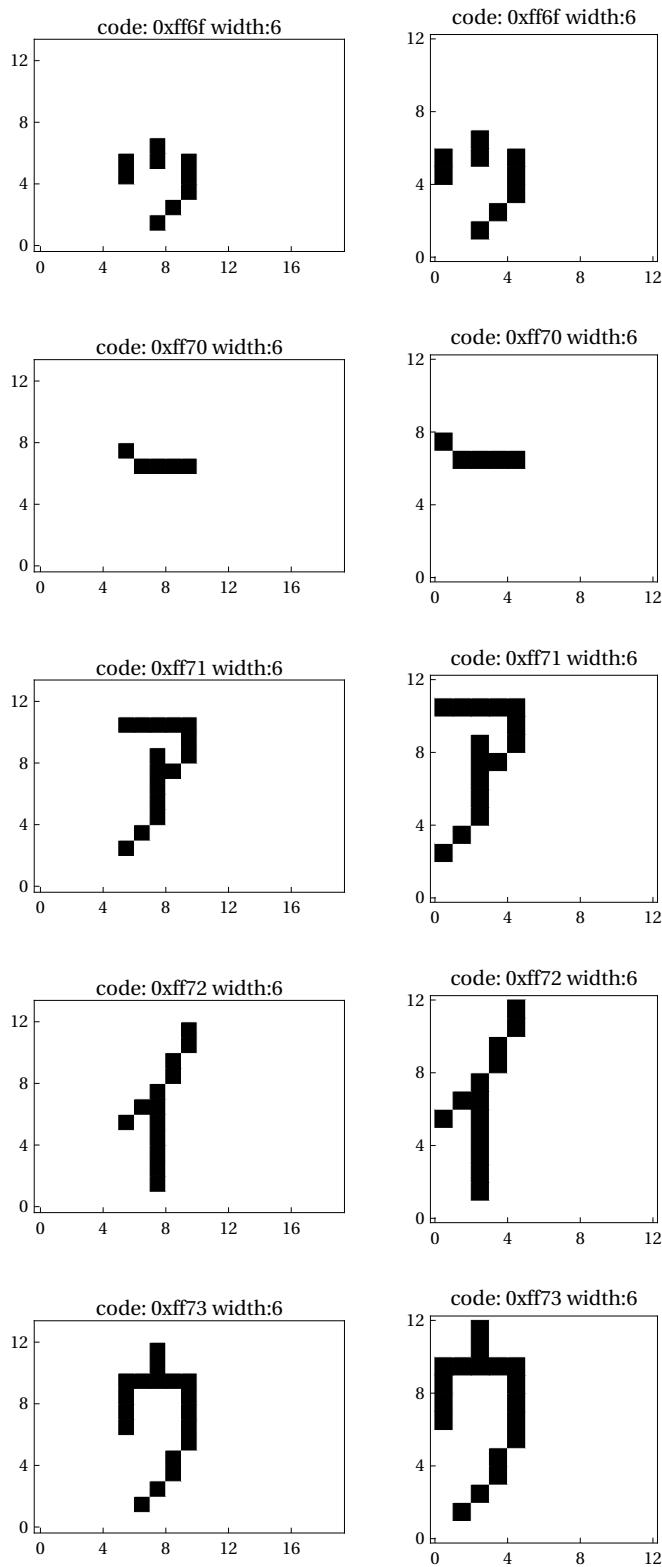


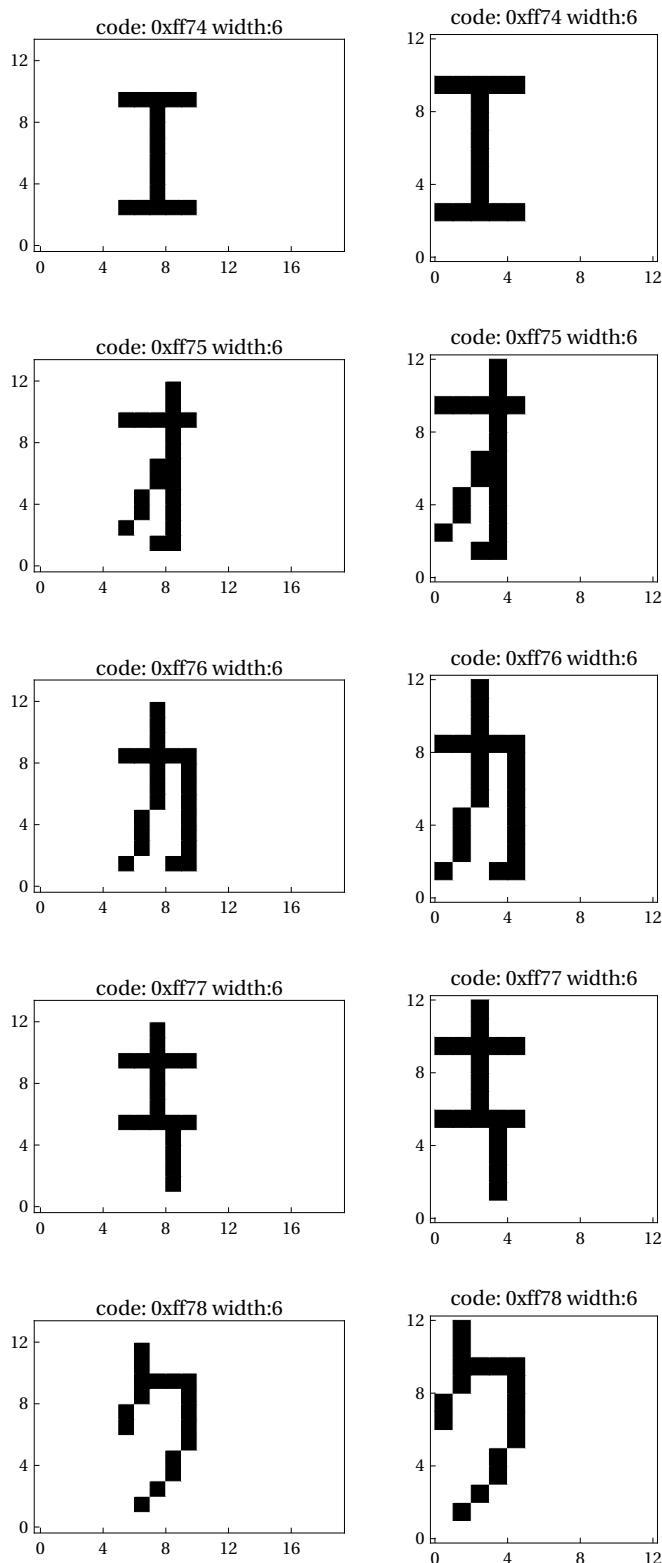


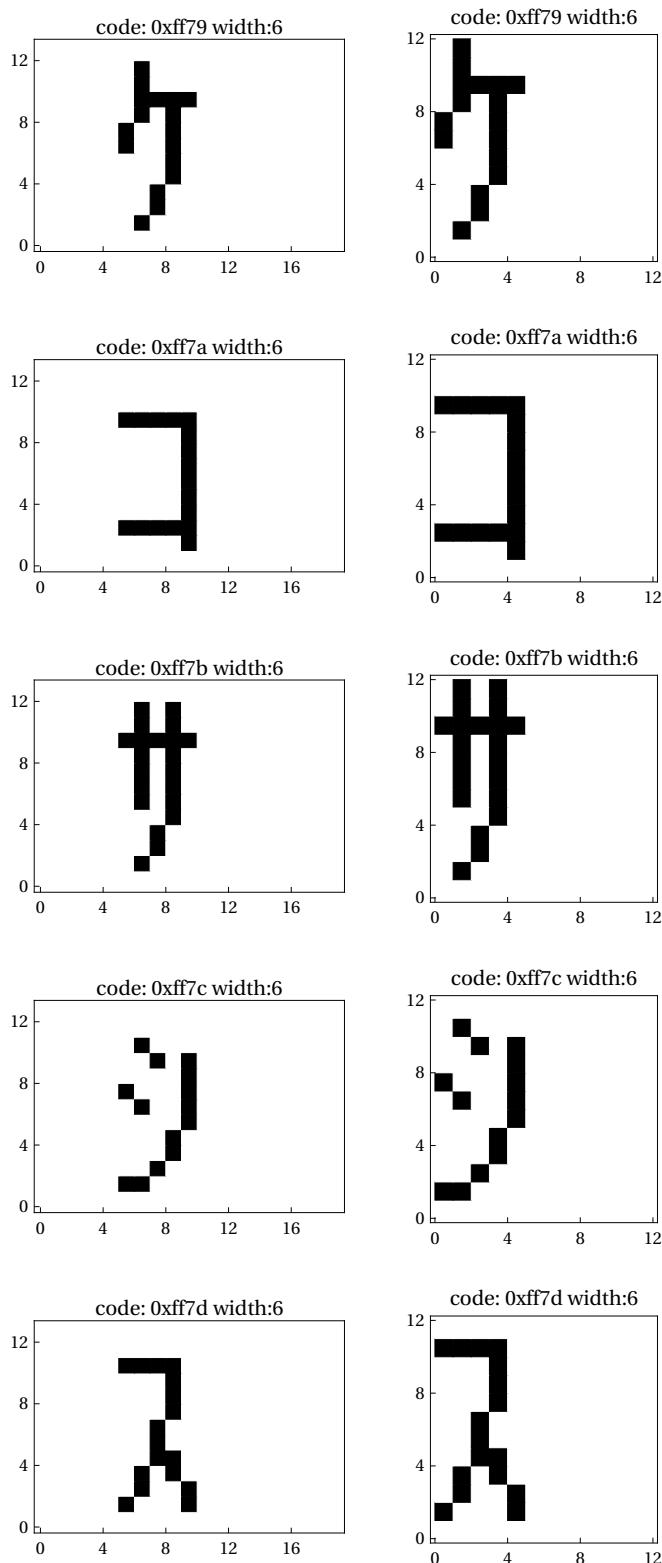


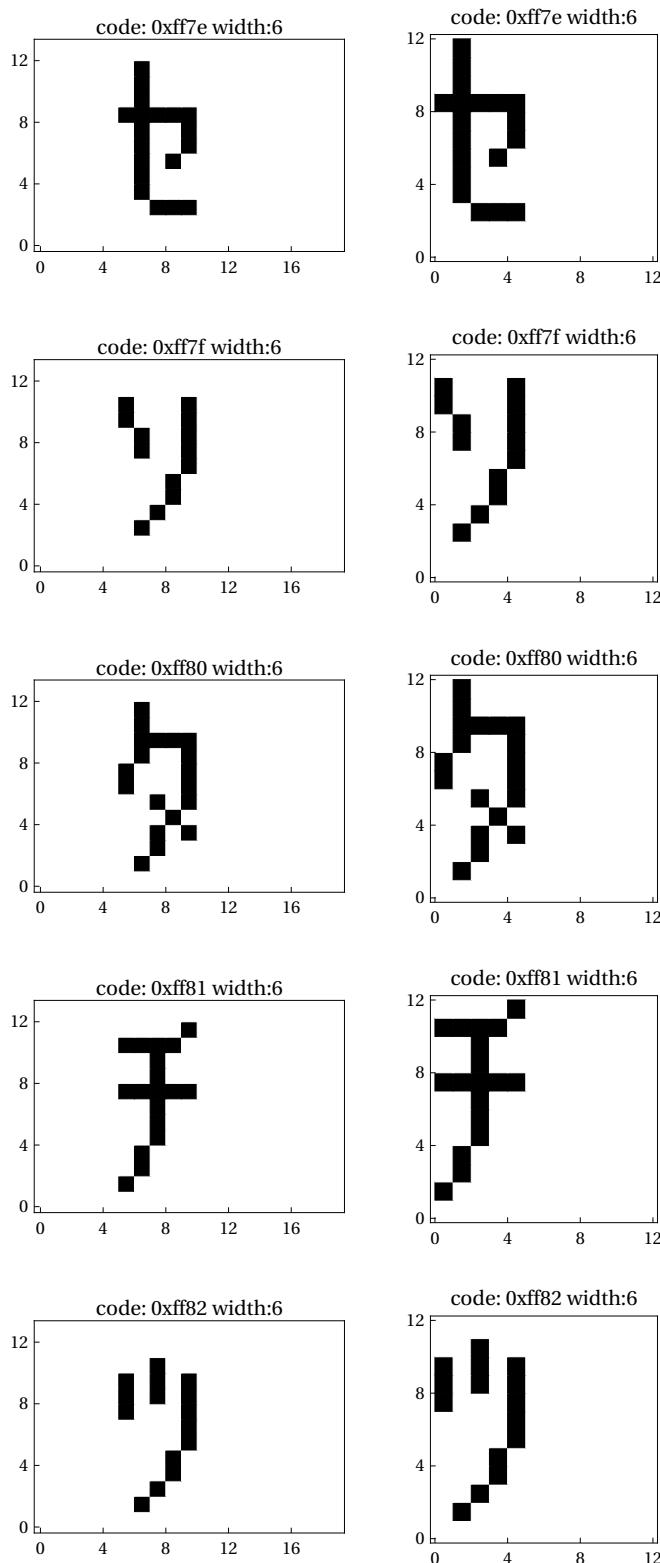


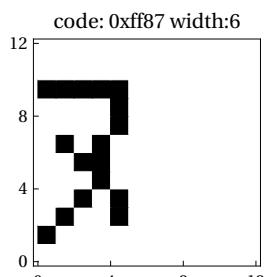
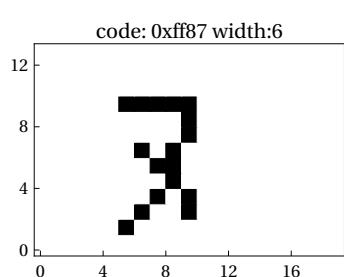
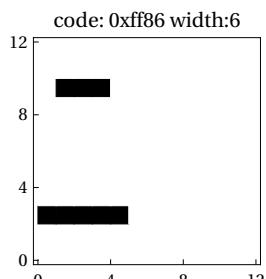
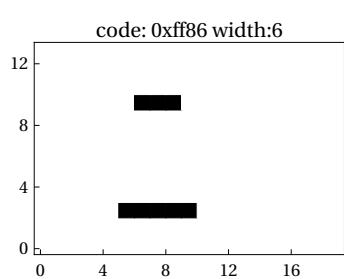
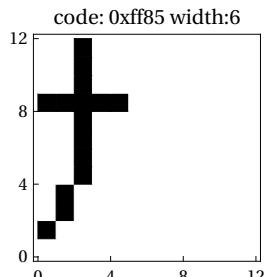
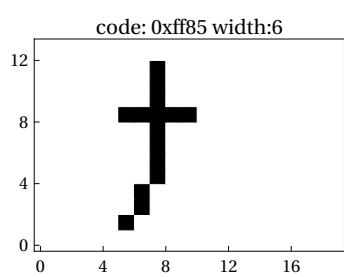
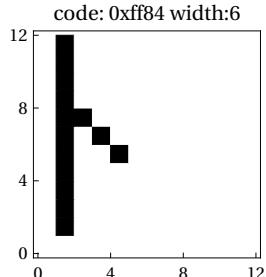
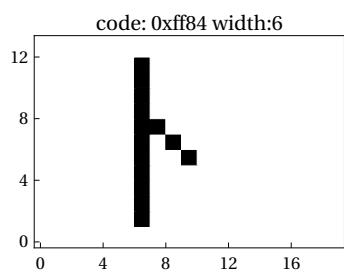
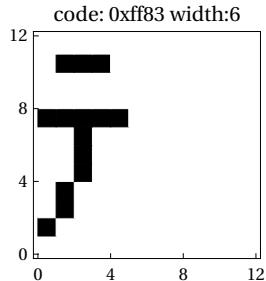
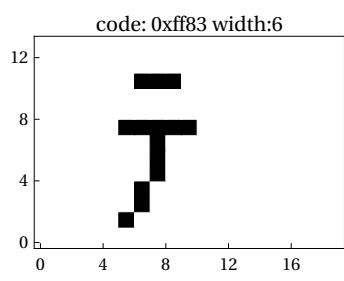


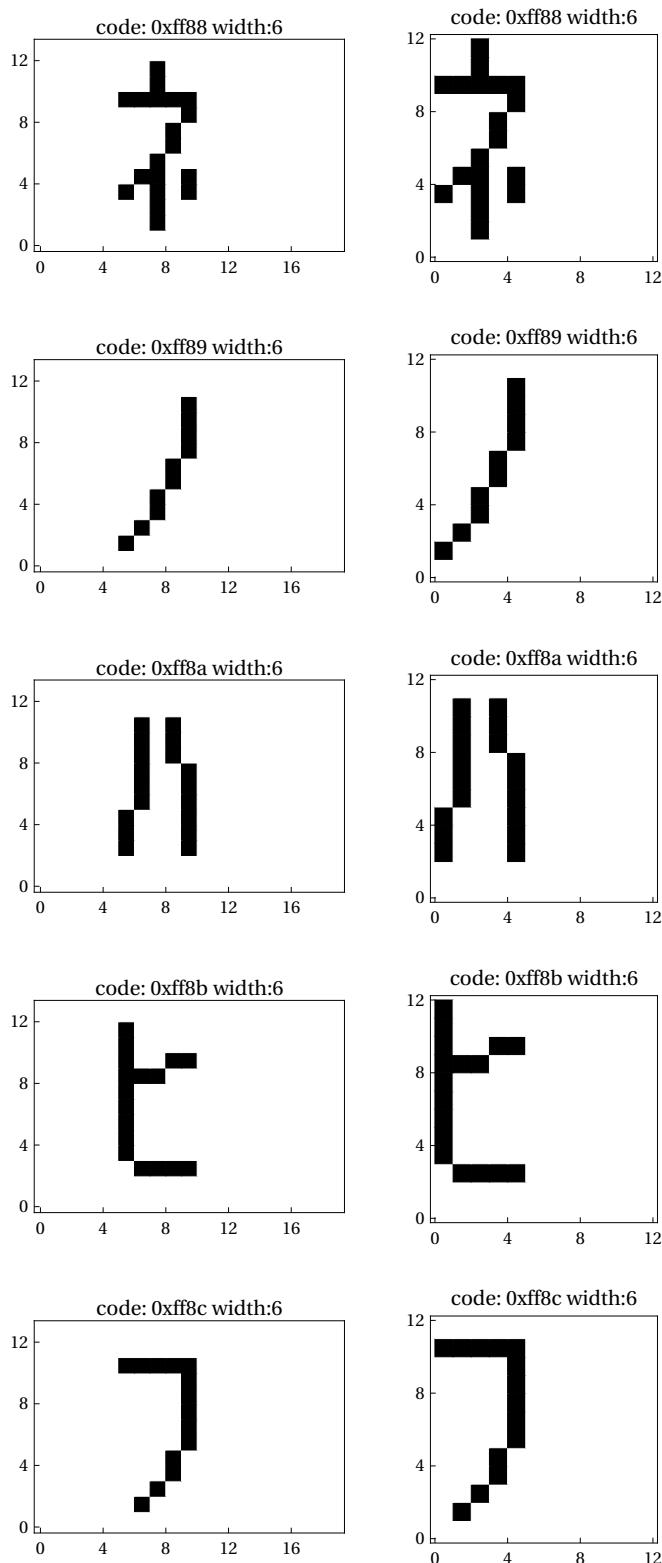


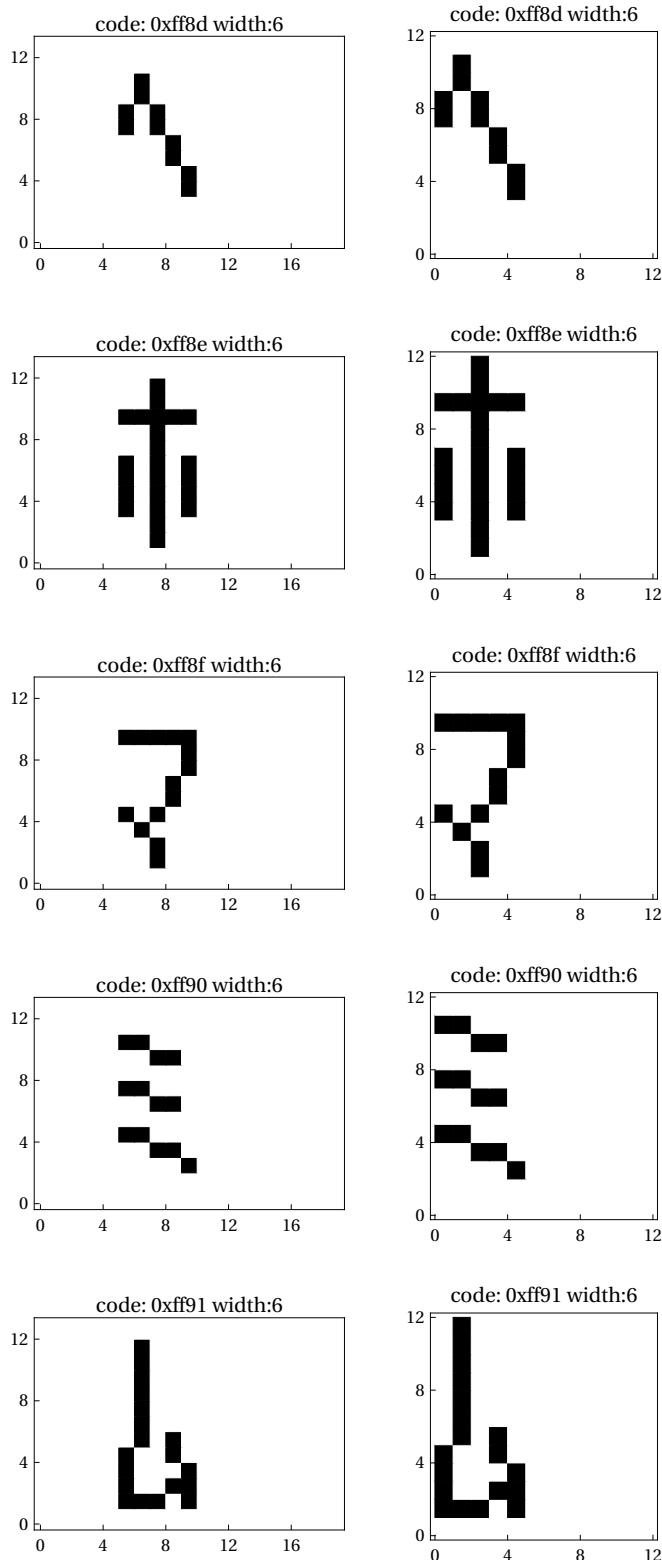


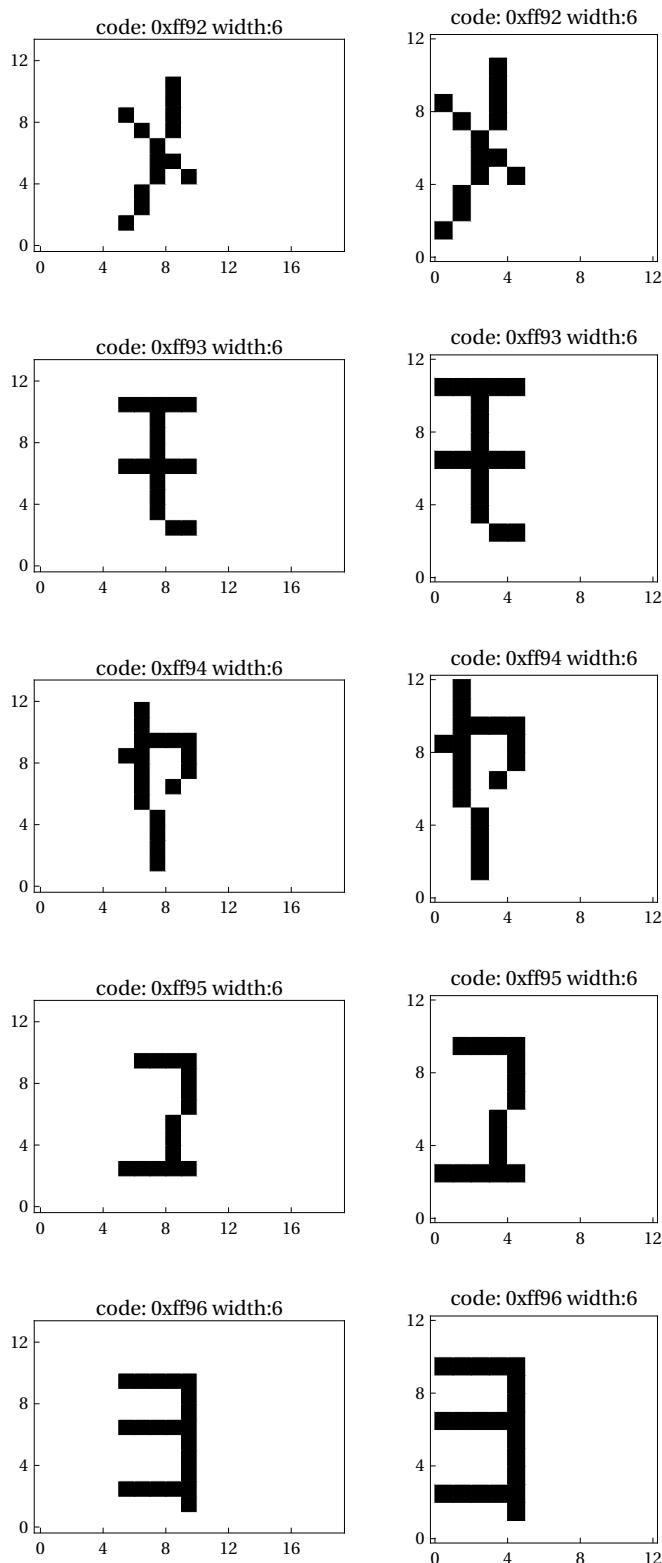


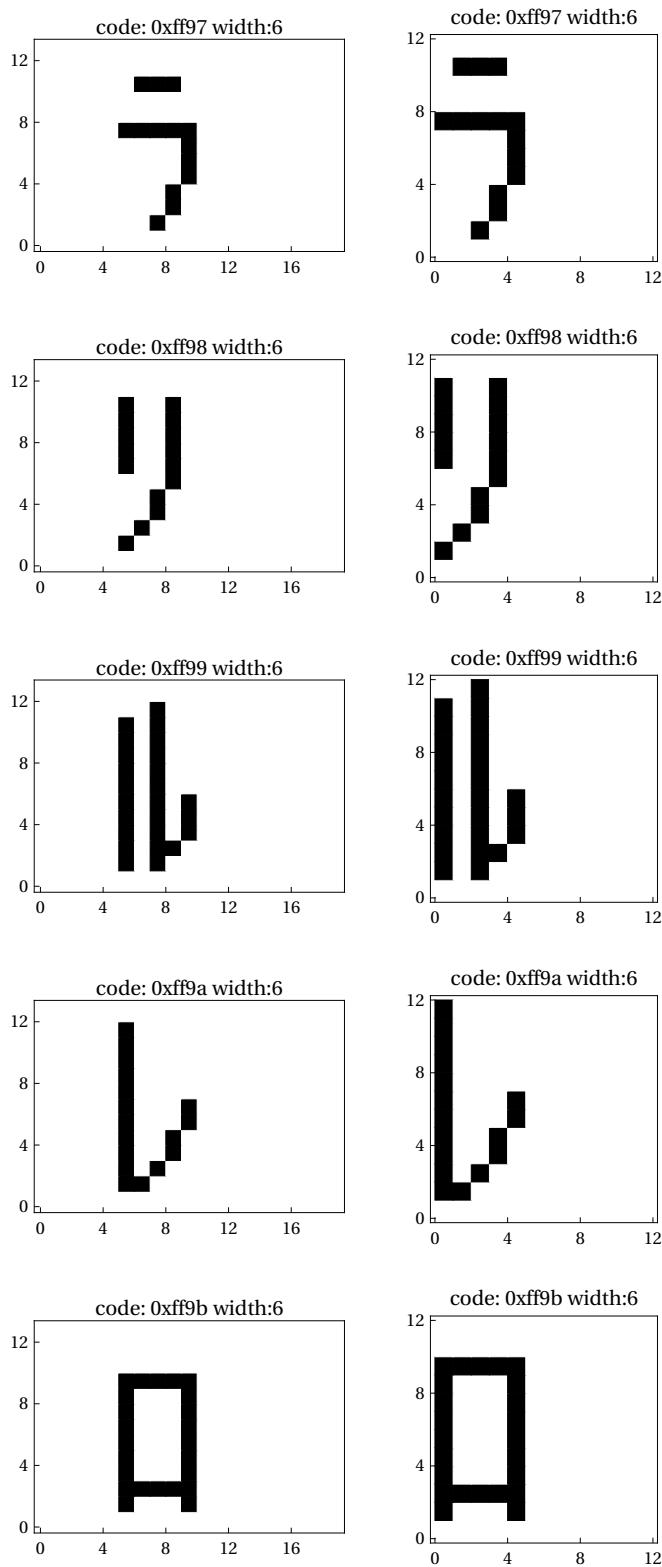


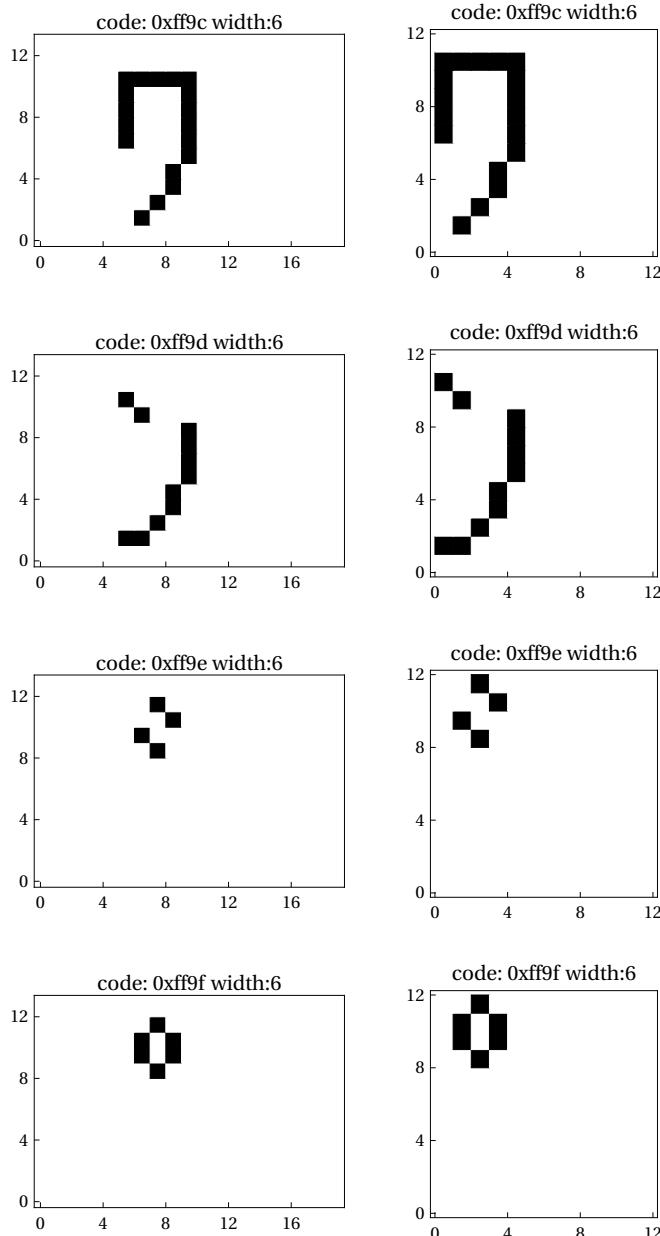






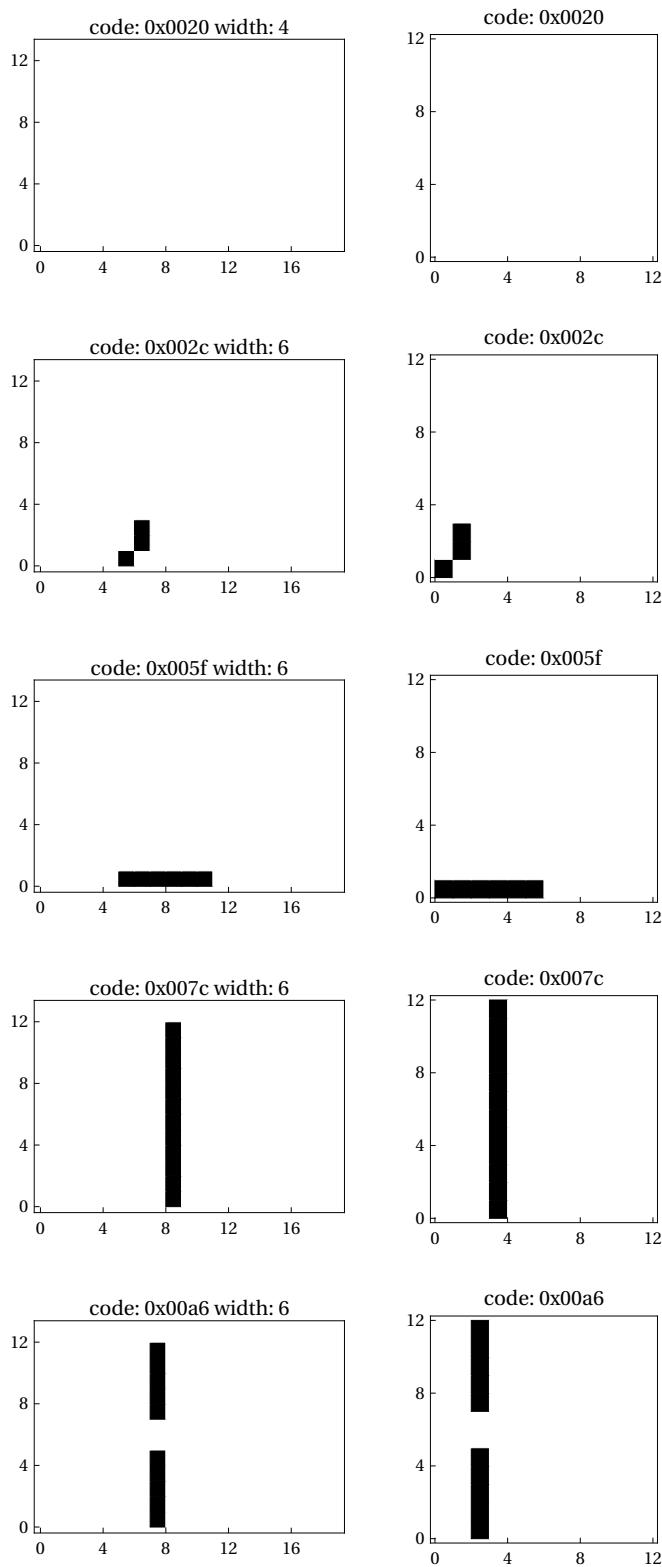


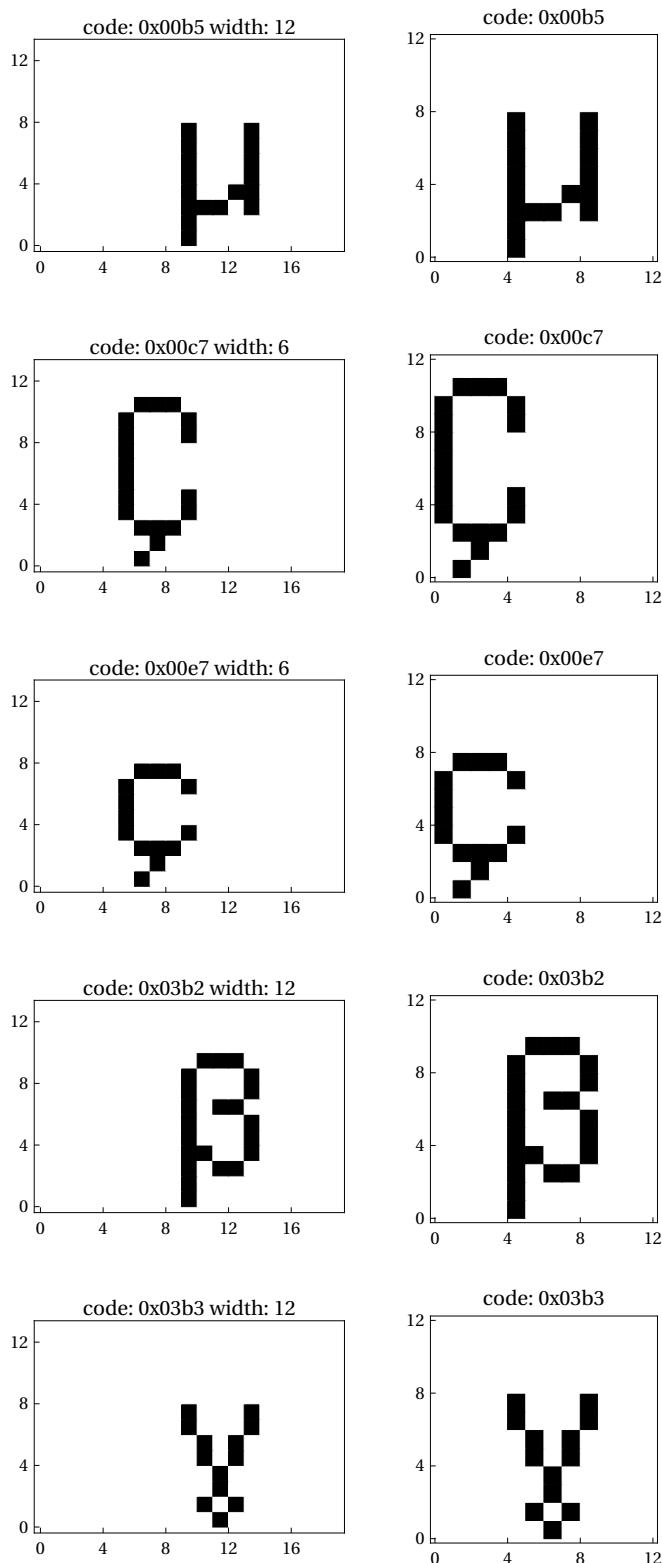


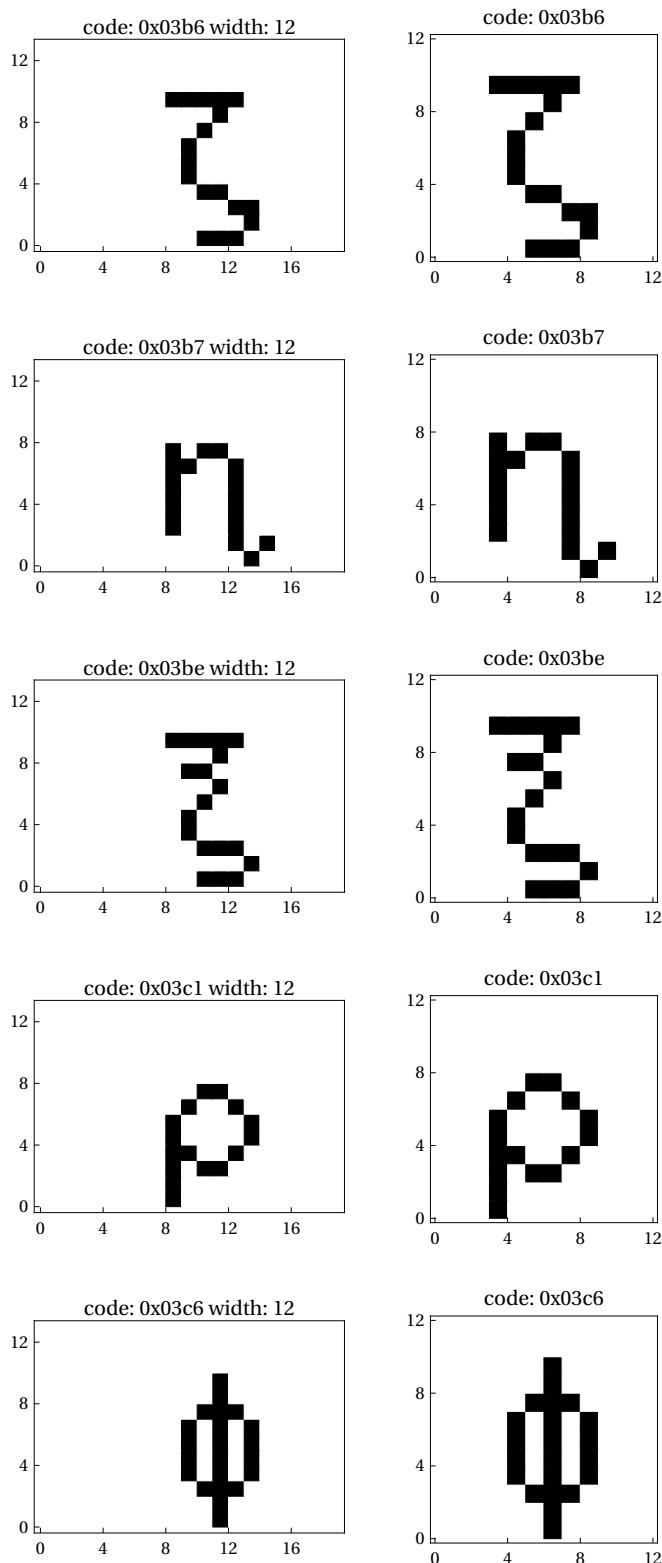


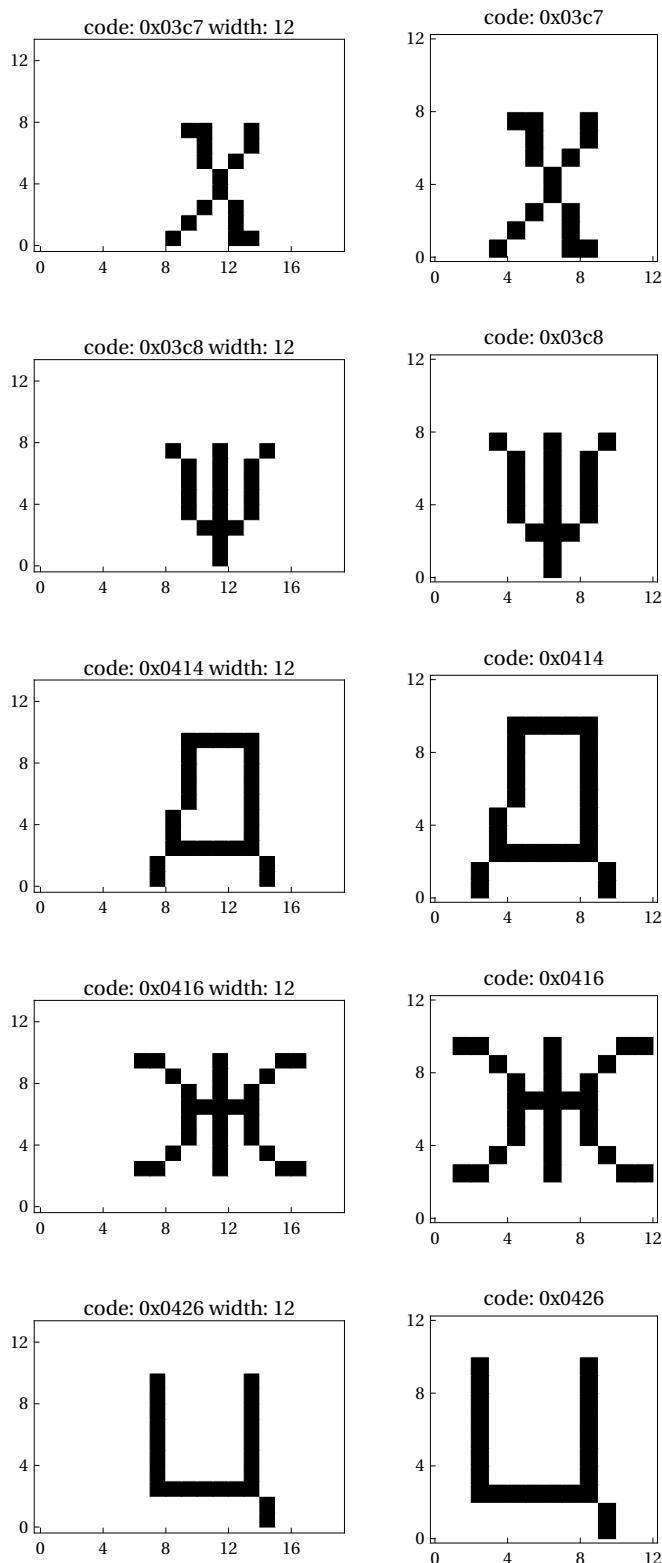
■ Bounding box outlier glyphs

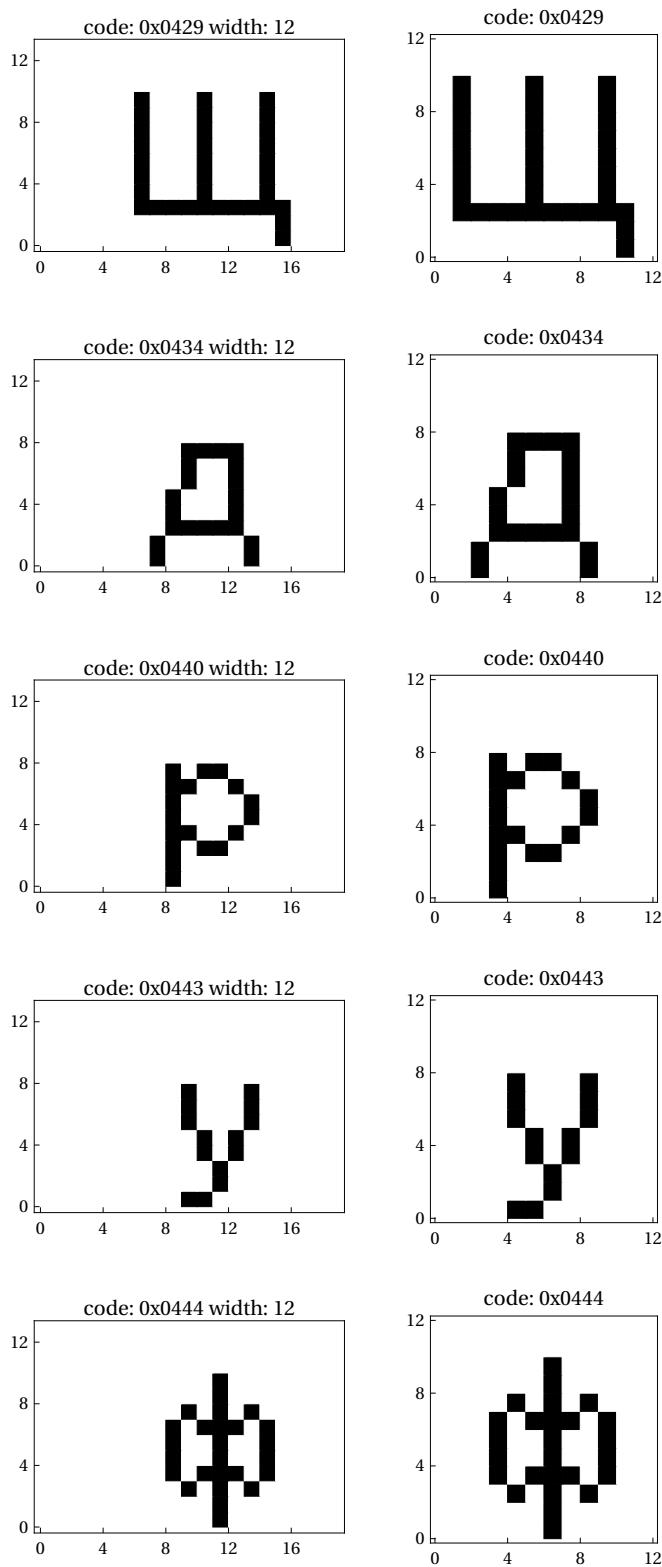
```
In[43]:= outliergraphics = MapThread[Graphics[Raster[Reverse[#1]],  
Frame -> True, AspectRatio -> Divide @@ Dimensions[#1],  
FrameTicks -> ({#, #, {}, {}} & [Range[0, 16, 4]]), PlotLabel -> ("code: 0x" <>  
IntegerString[ToExpression[#2], 16, 4] <> " width: " <> ToString[#3])] &,  
#\[Outliers] & /@ {extbitmaps, charcodes, widths}];  
  
In[44]:= outliertruncgraphics =  
MapThread[Graphics[Raster[Reverse[#1]], Frame -> True, AspectRatio ->  
Divide @@ Dimensions[#1], FrameTicks -> ({#, #, {}, {}} & [Range[0, 16, 4]]),  
PlotLabel -> ("code: 0x" <> IntegerString[ToExpression[#2], 16, 4])] &,  
#\[Outliers] & /@ {trbitmaps, charcodes, widths}];  
  
In[45]:= MapThread[Print[GraphicsGrid[{{##}}]] &, {outliergraphics, outliertruncgraphics}];
```

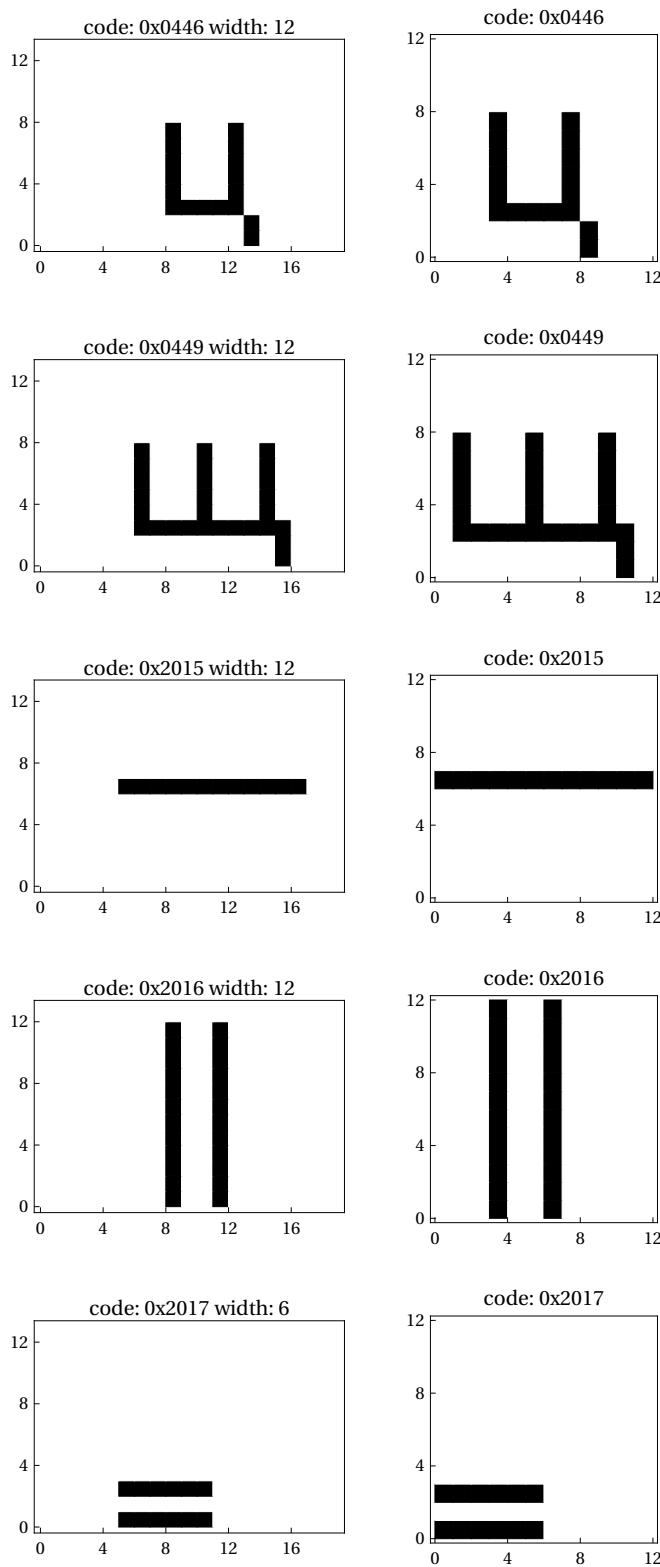


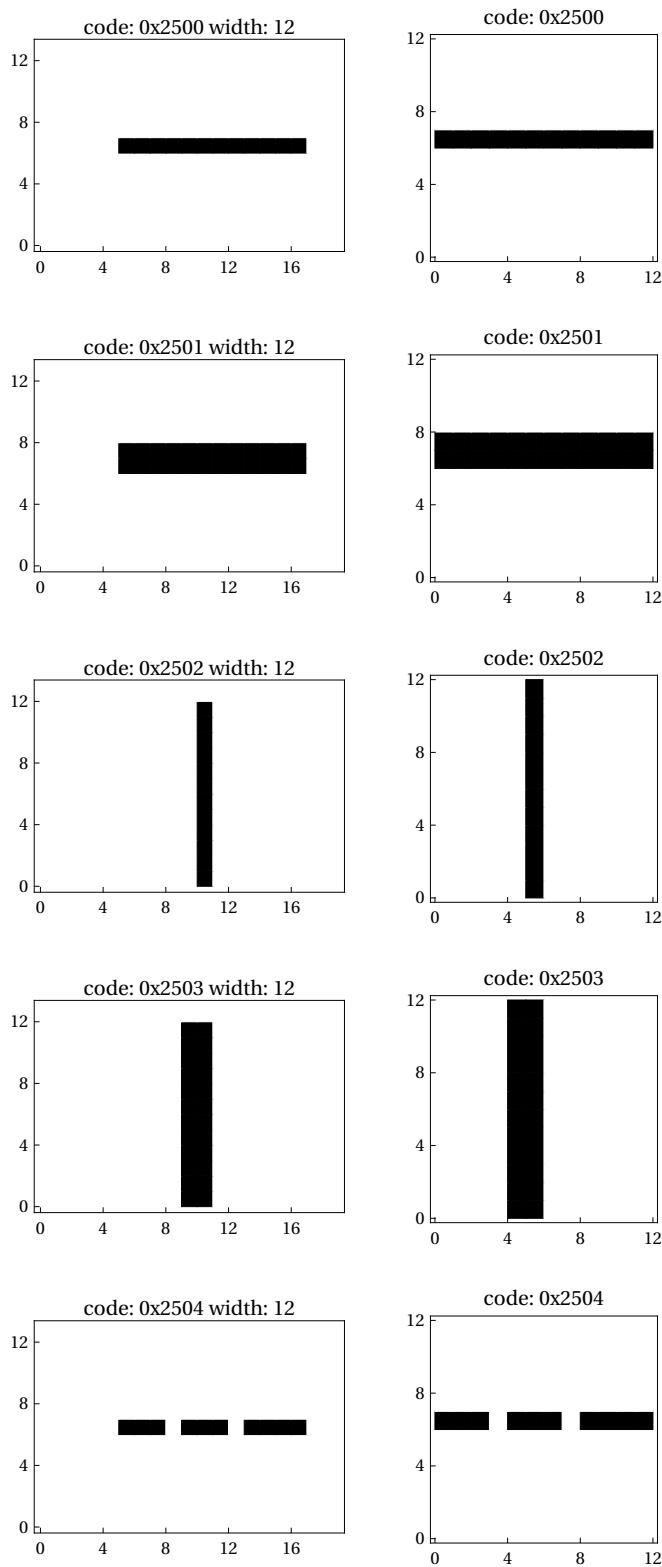


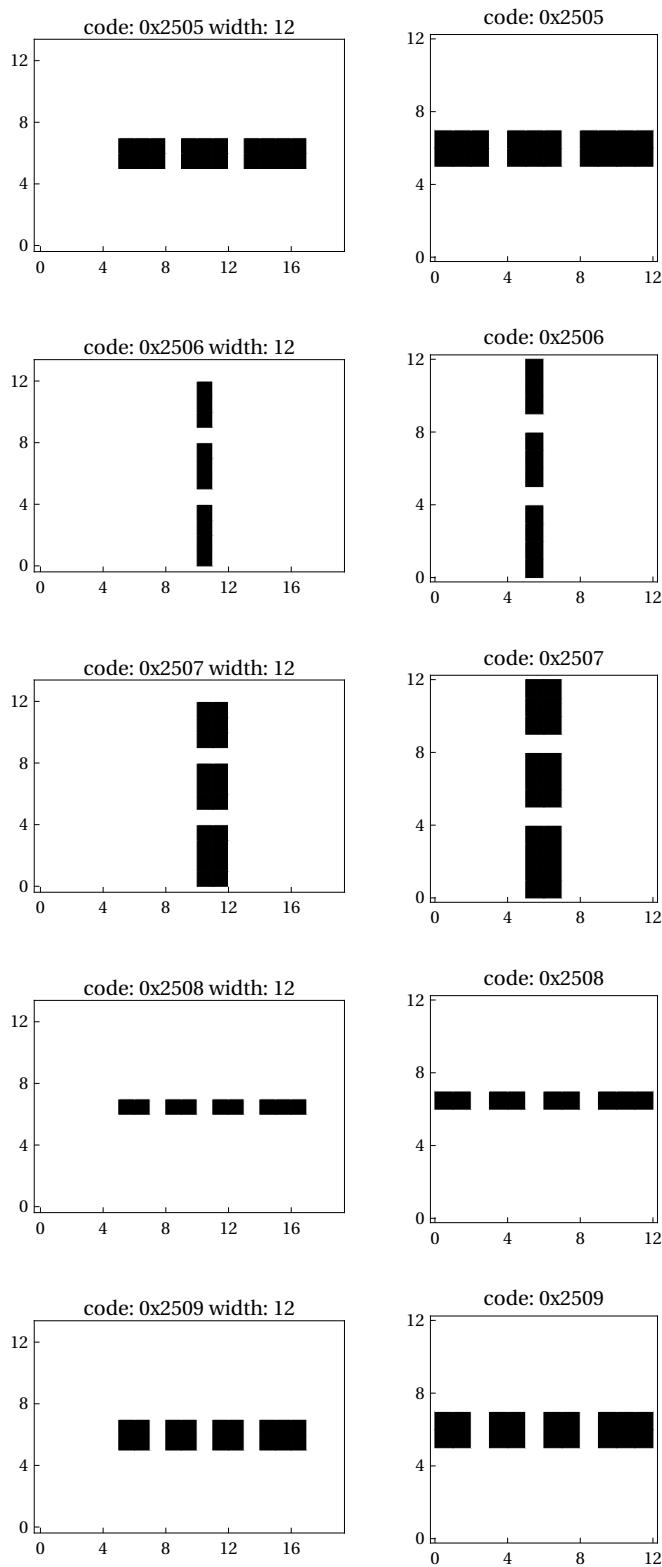


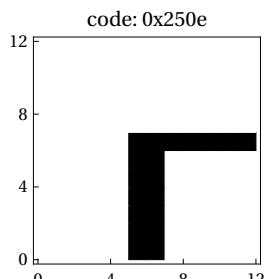
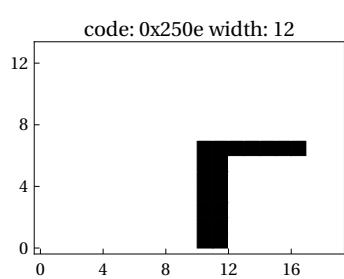
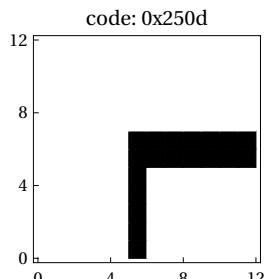
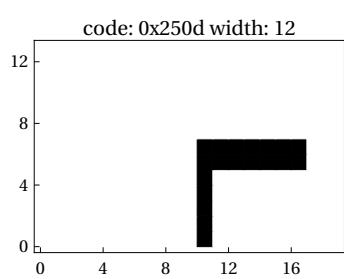
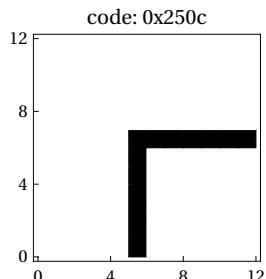
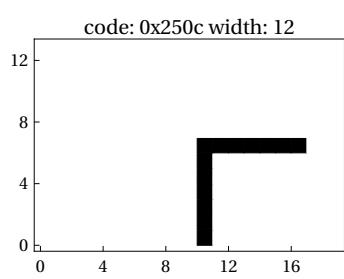
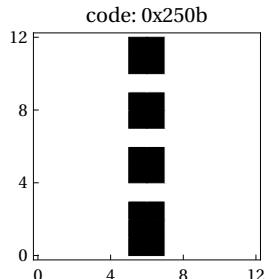
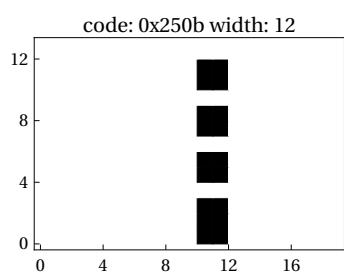
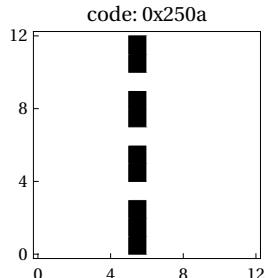
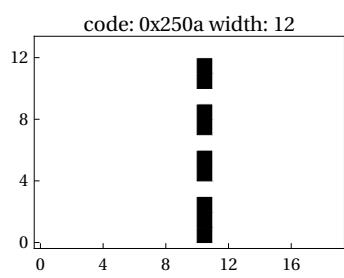


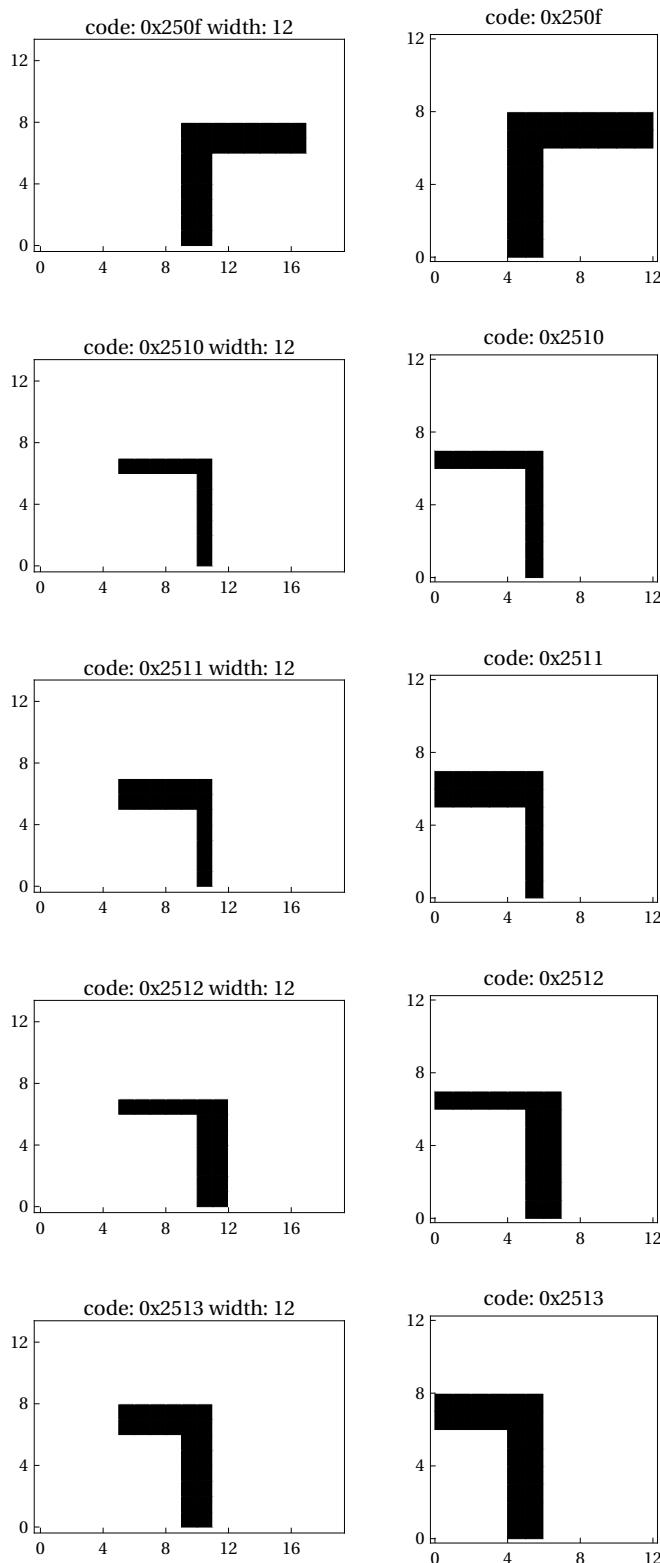


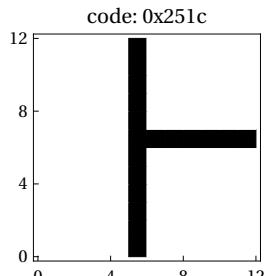
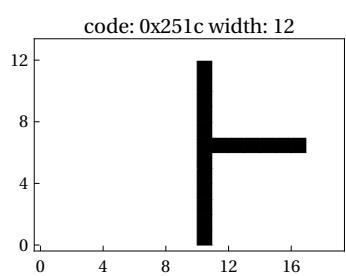
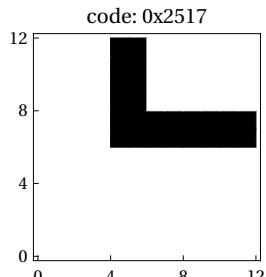
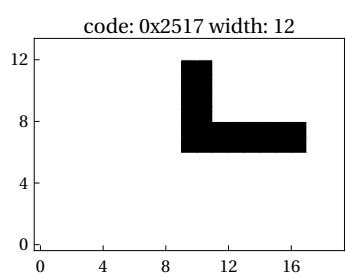
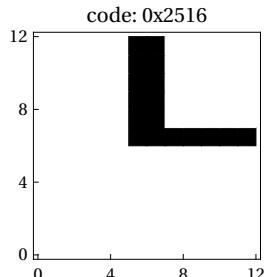
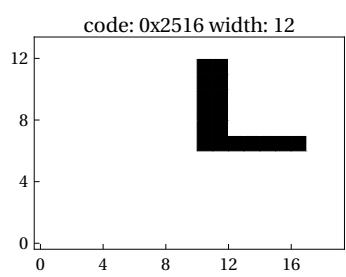
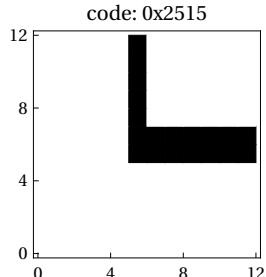
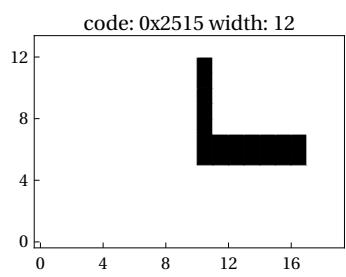
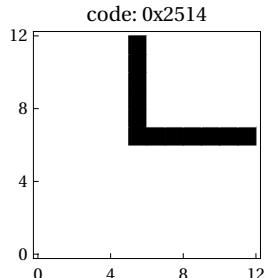
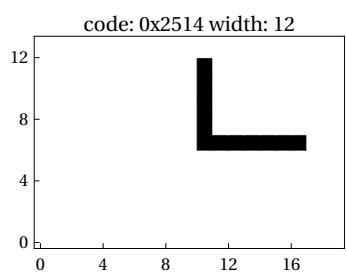


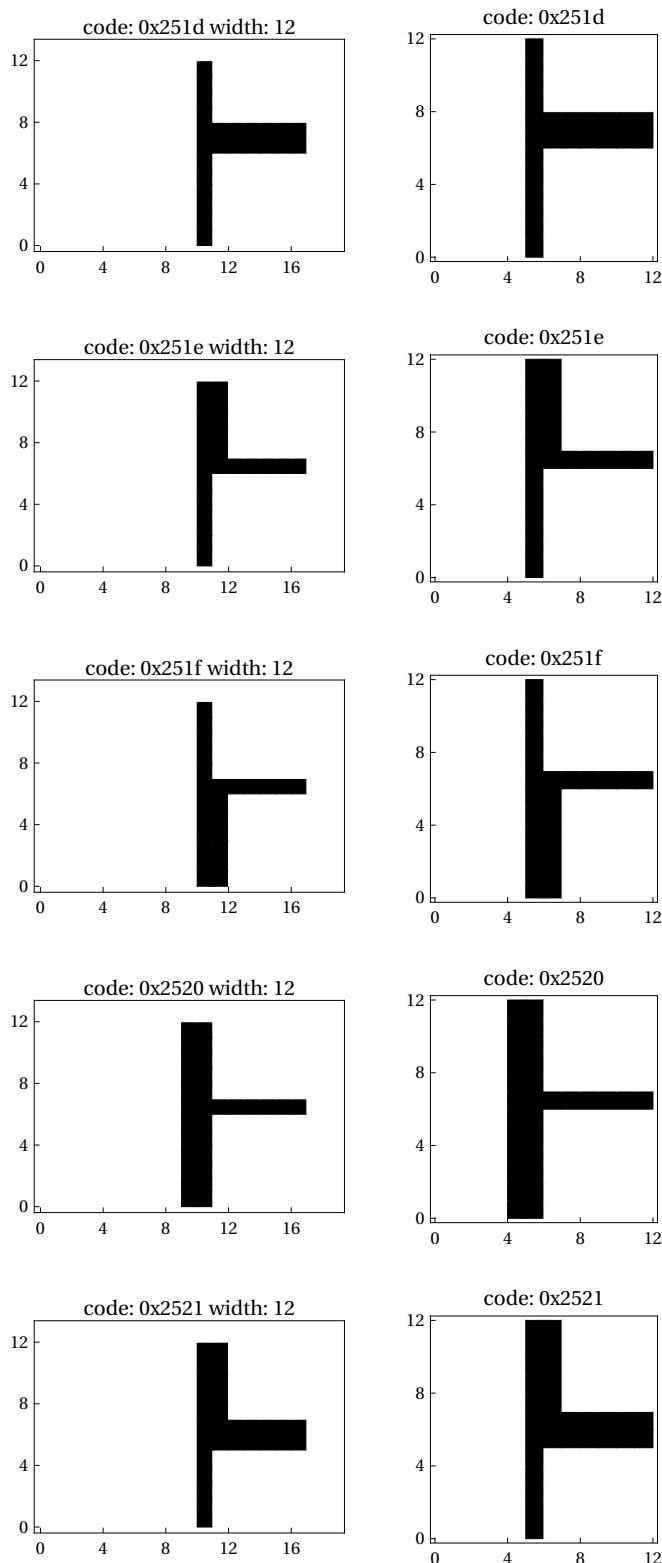


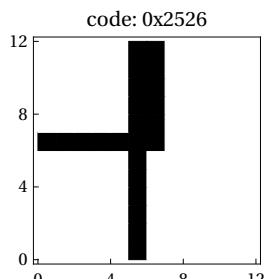
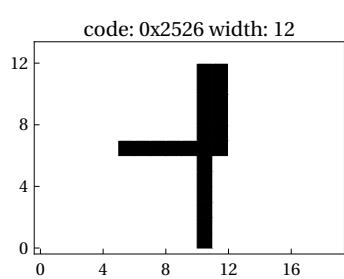
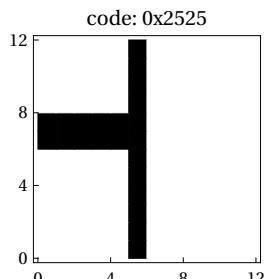
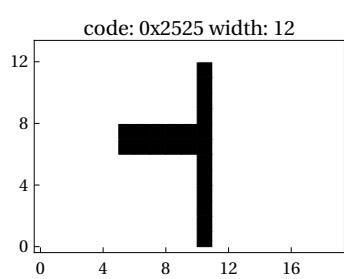
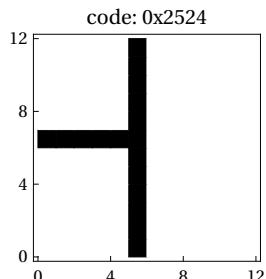
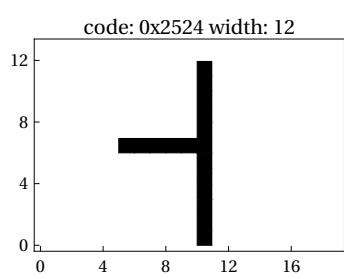
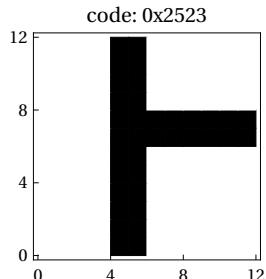
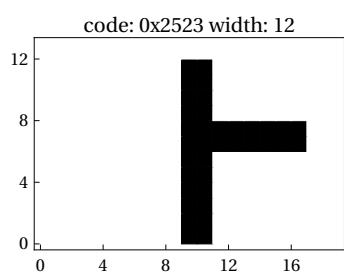
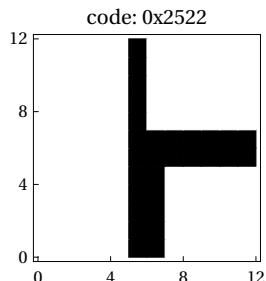
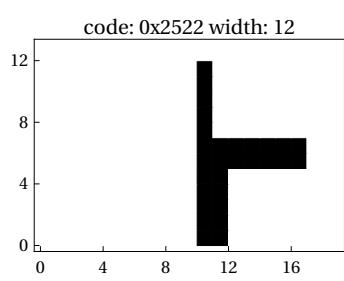


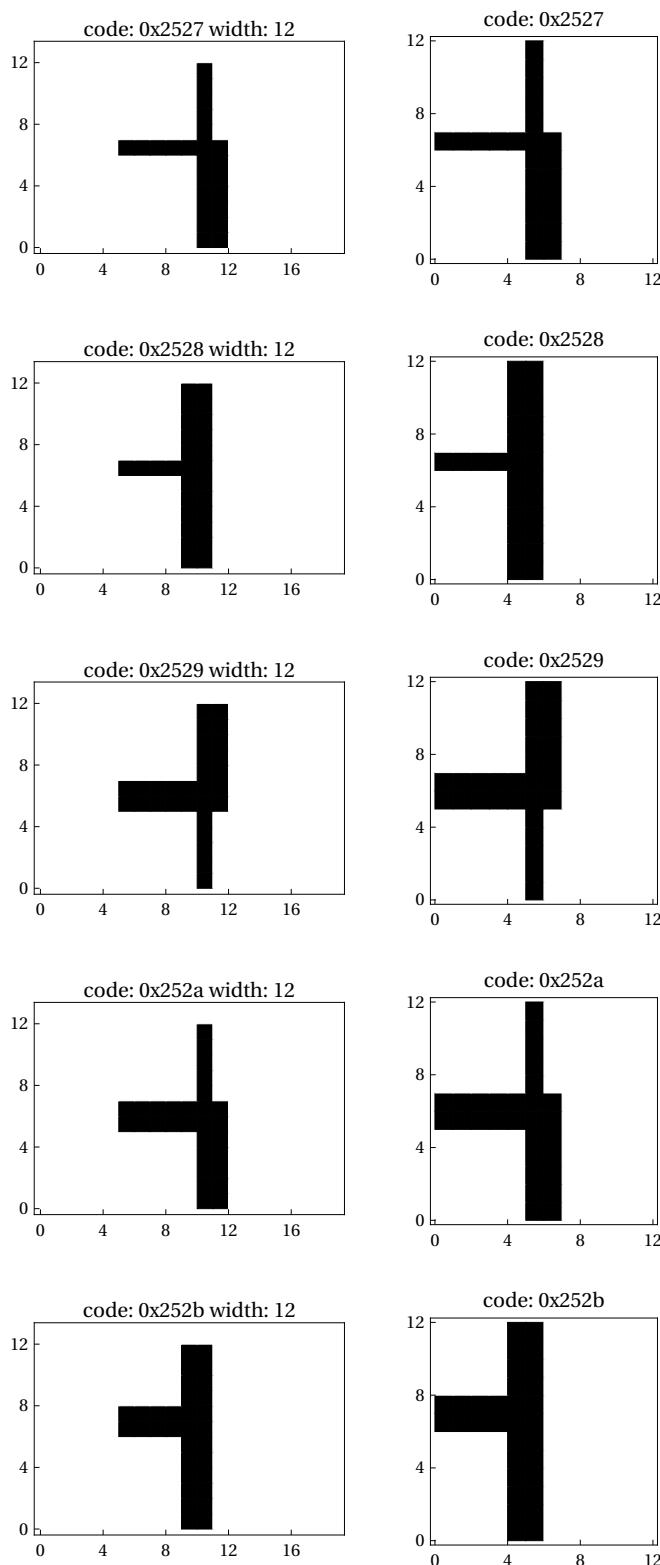


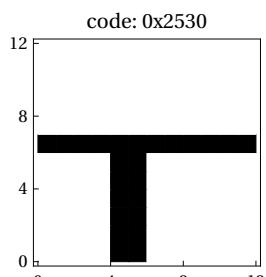
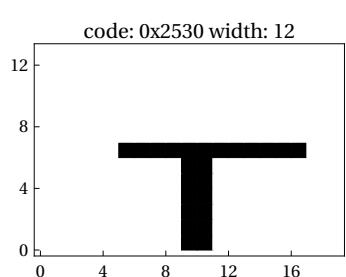
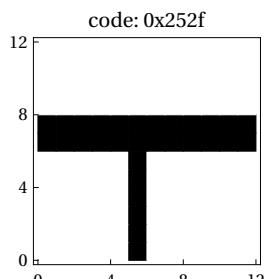
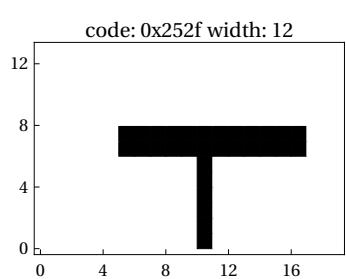
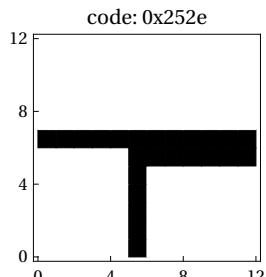
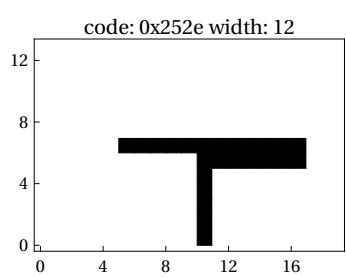
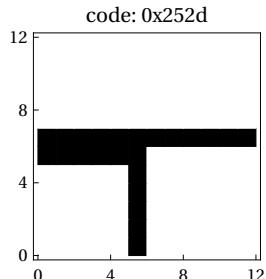
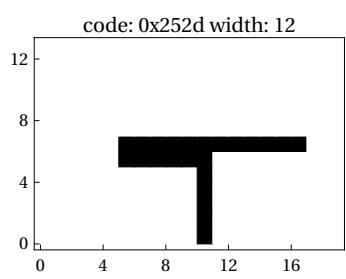
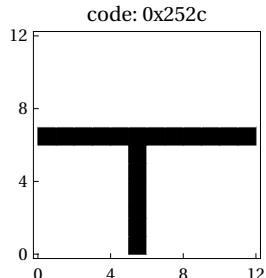
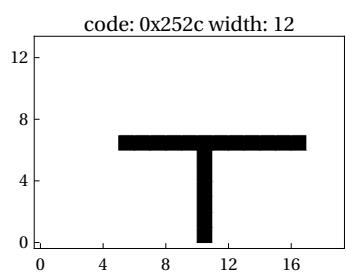


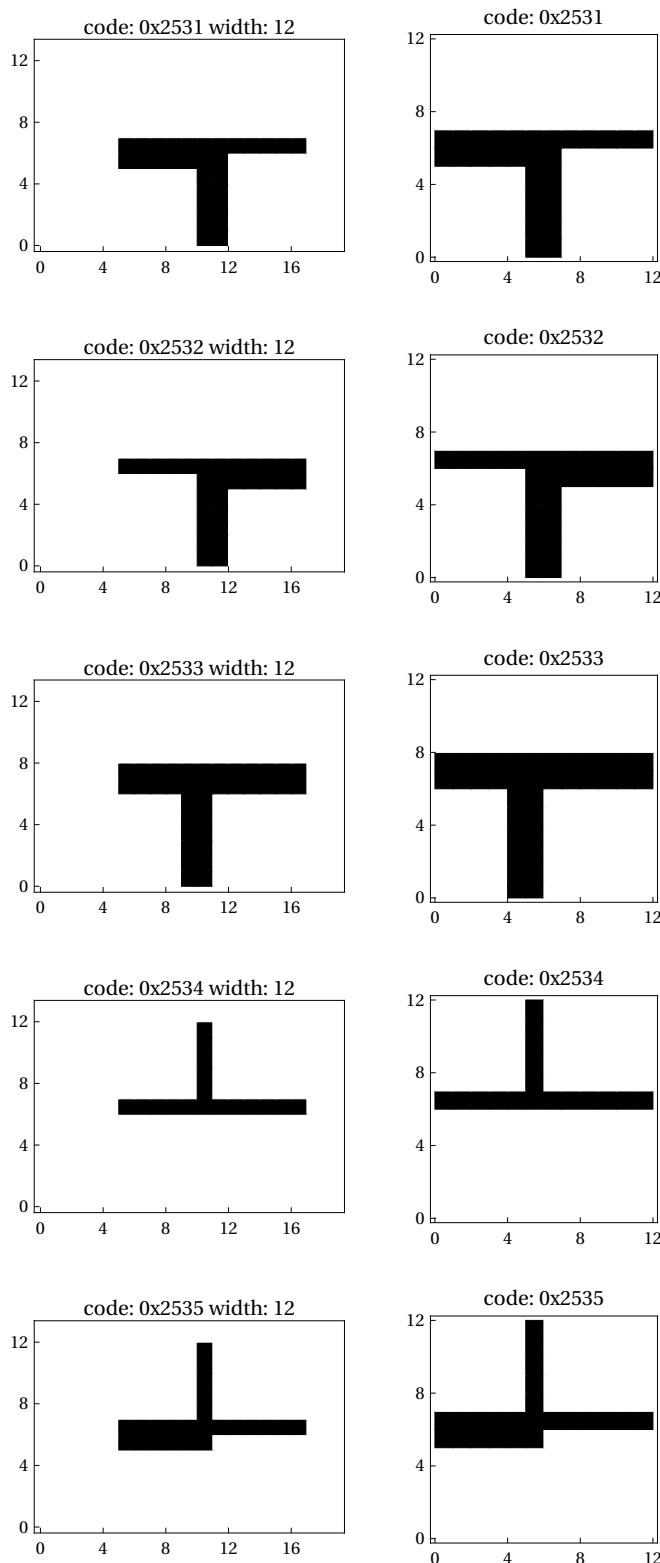


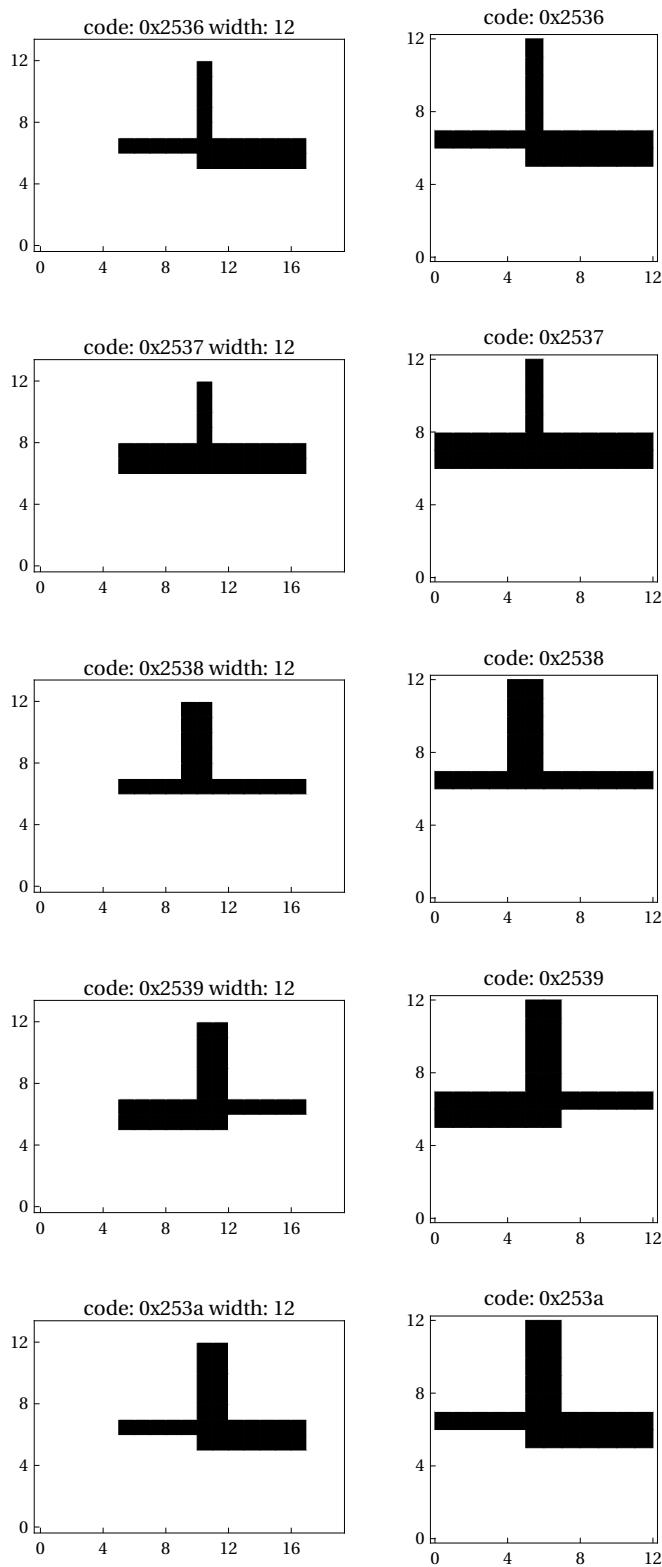


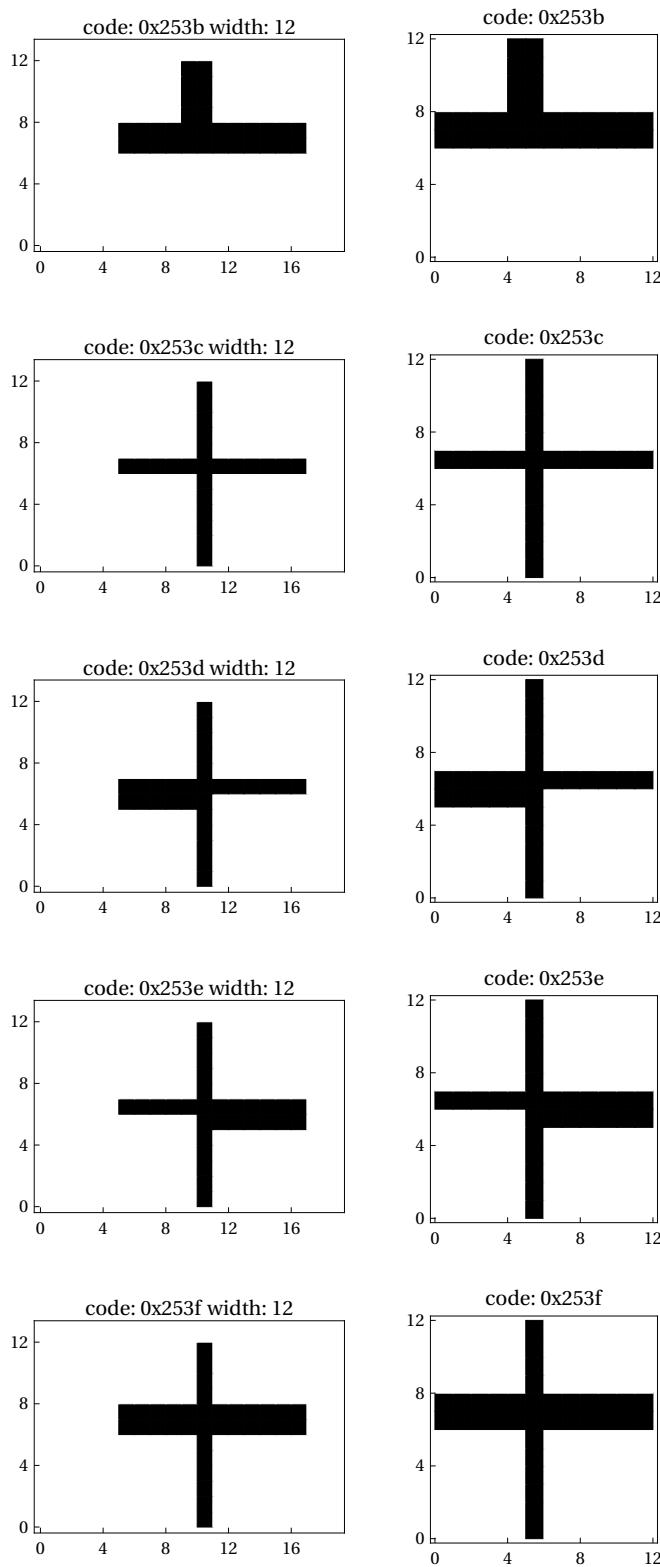


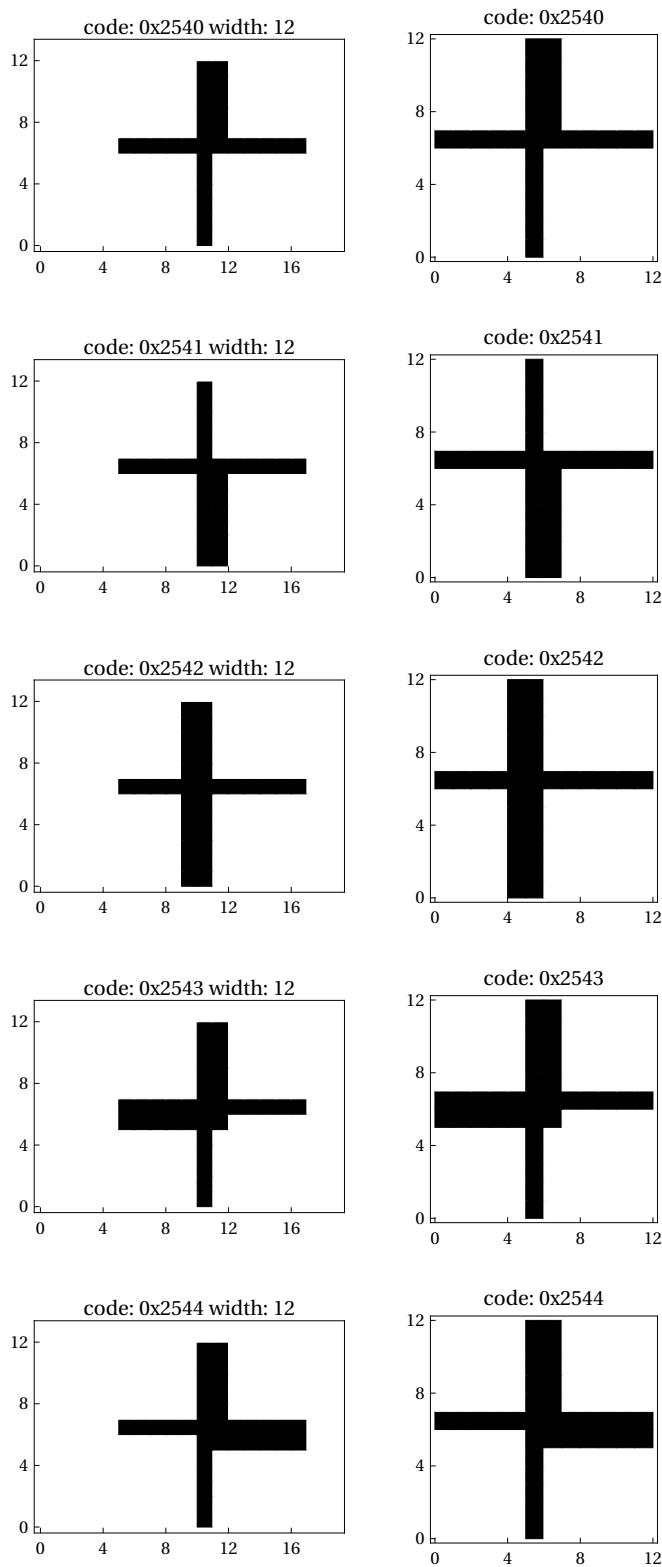


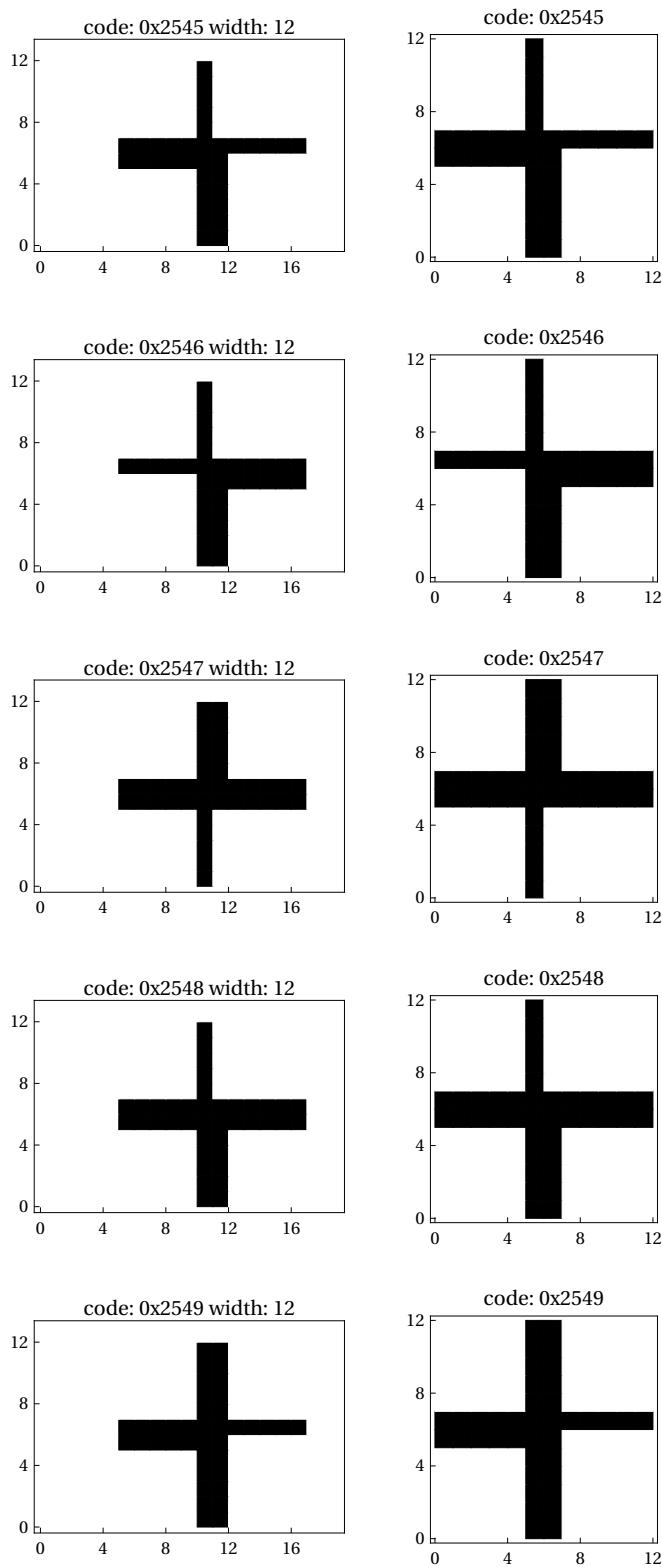


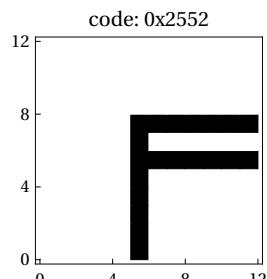
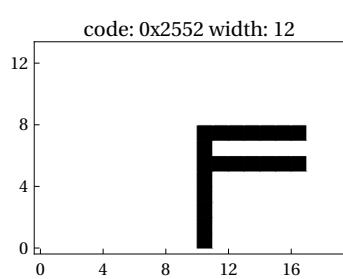
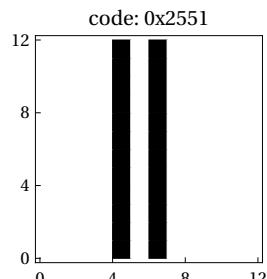
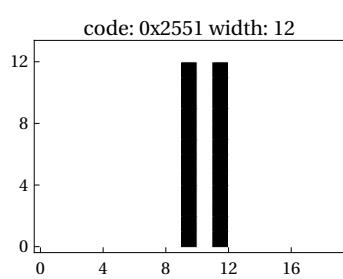
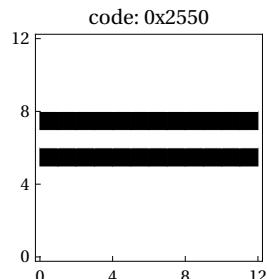
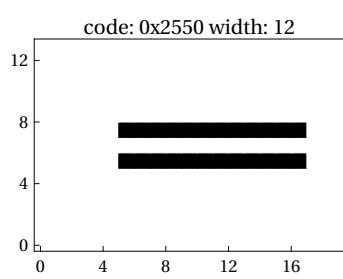
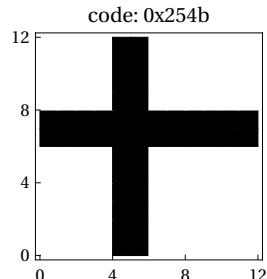
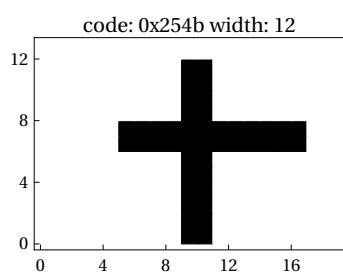
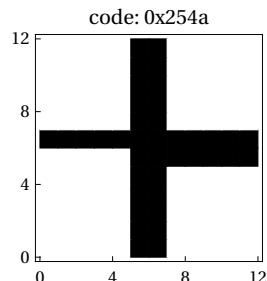
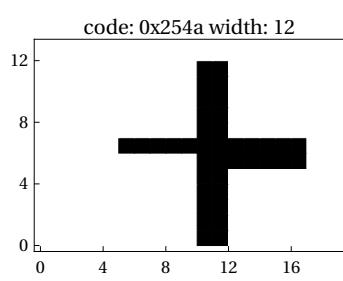


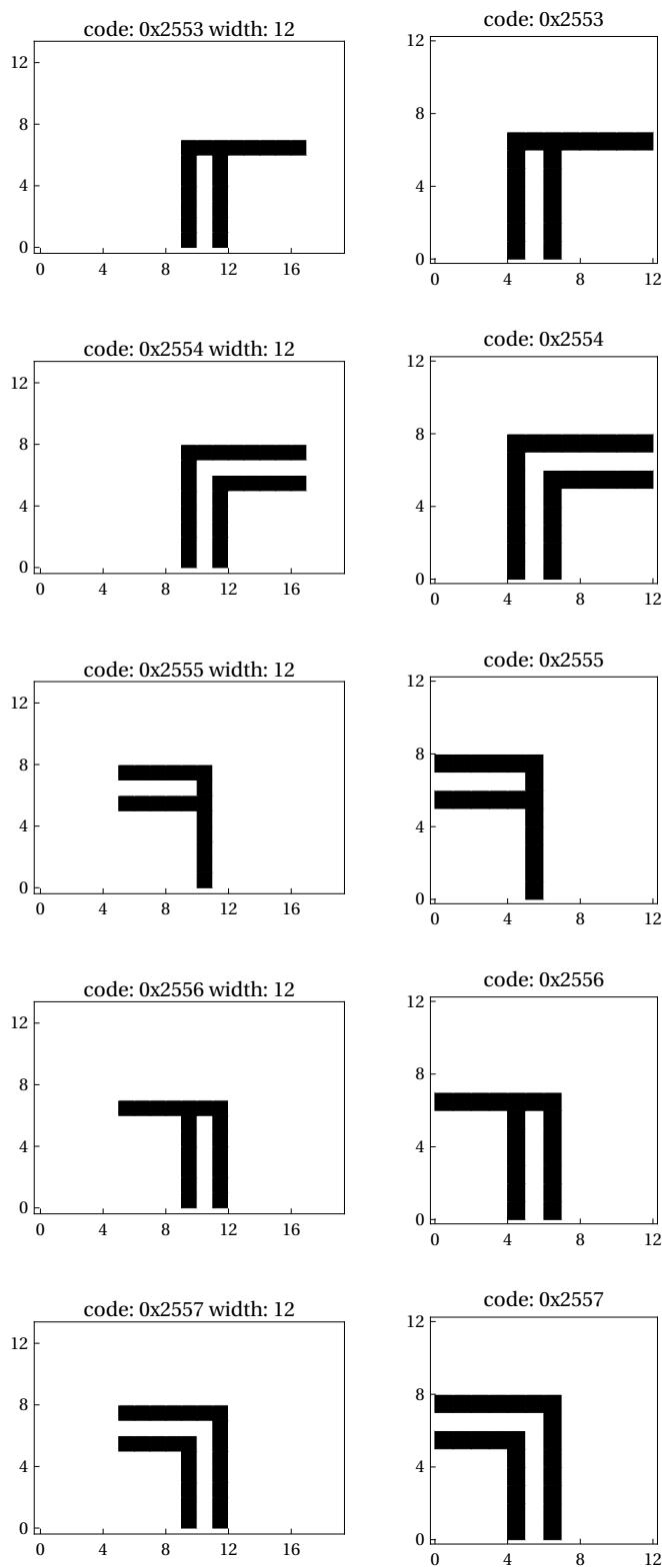


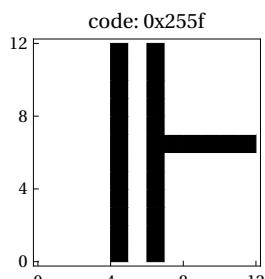
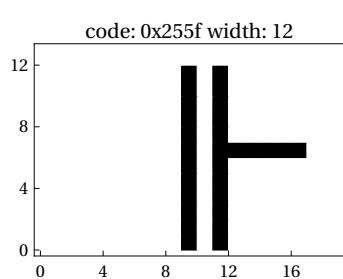
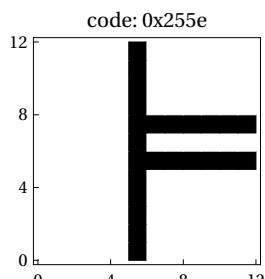
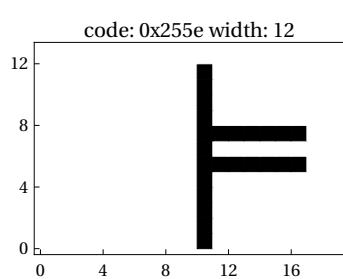
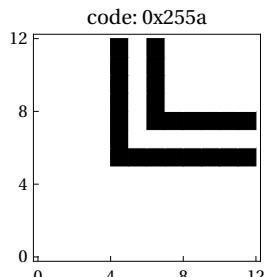
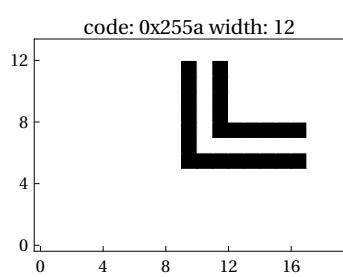
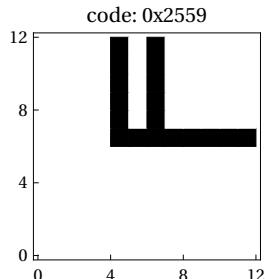
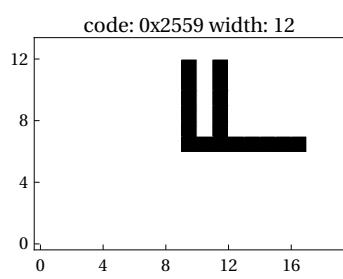
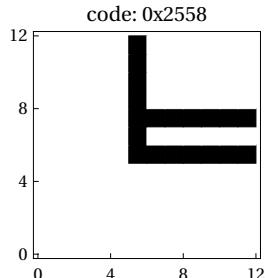
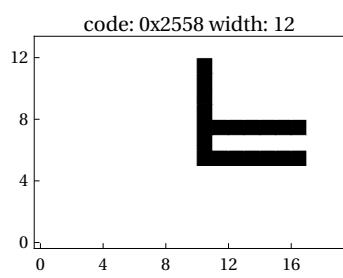


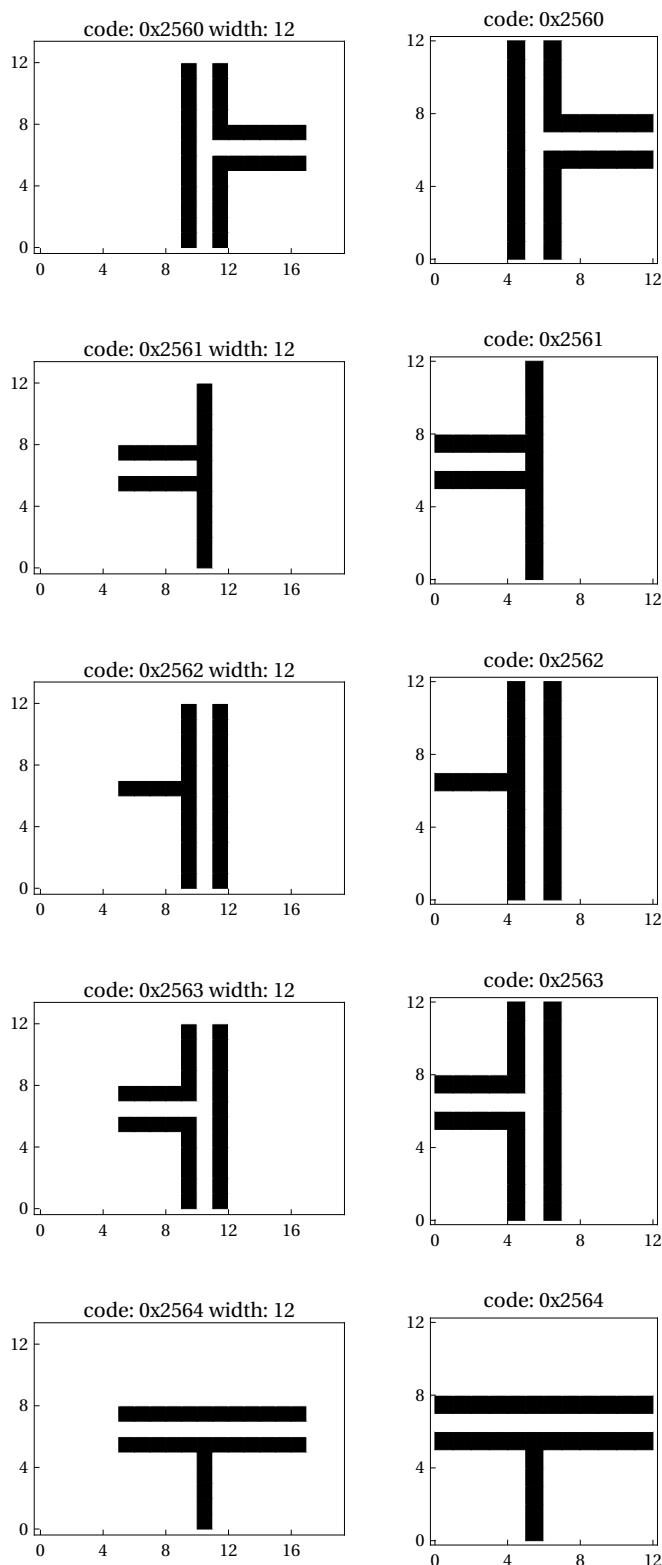


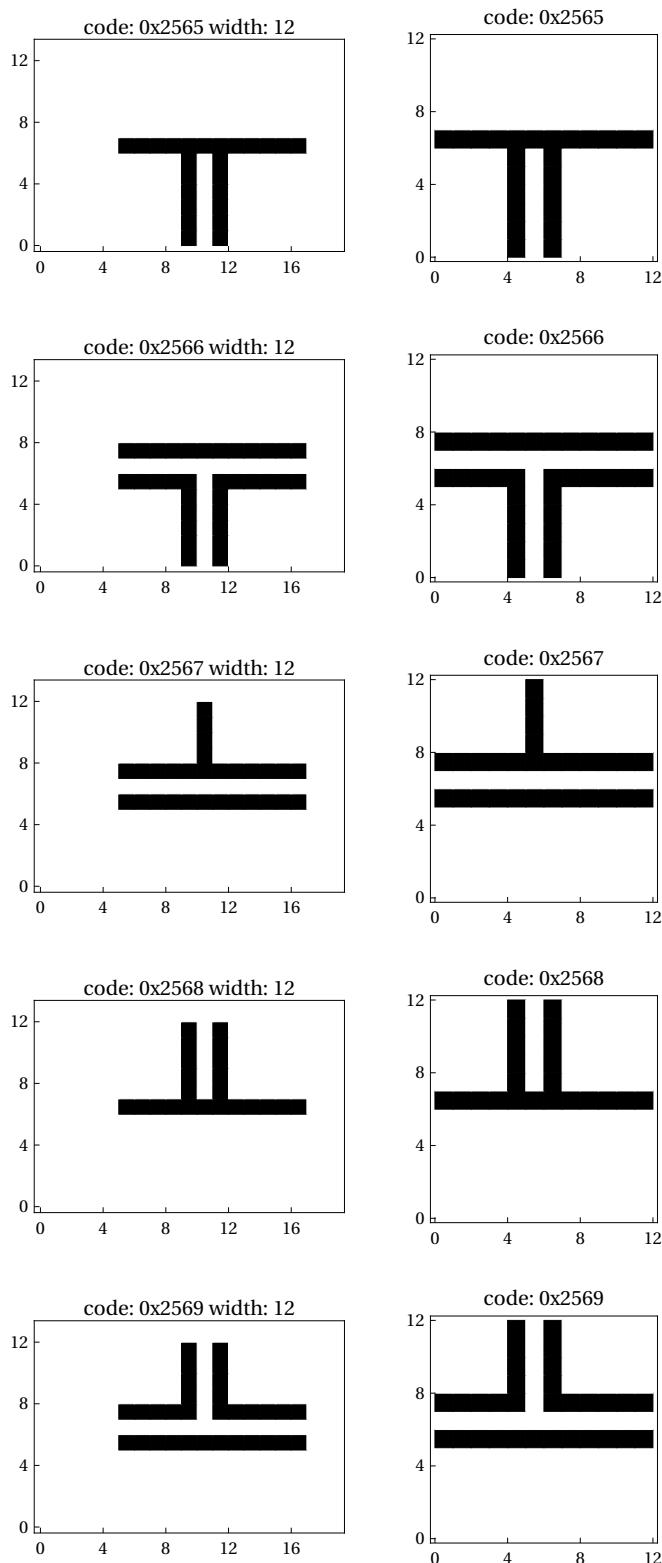


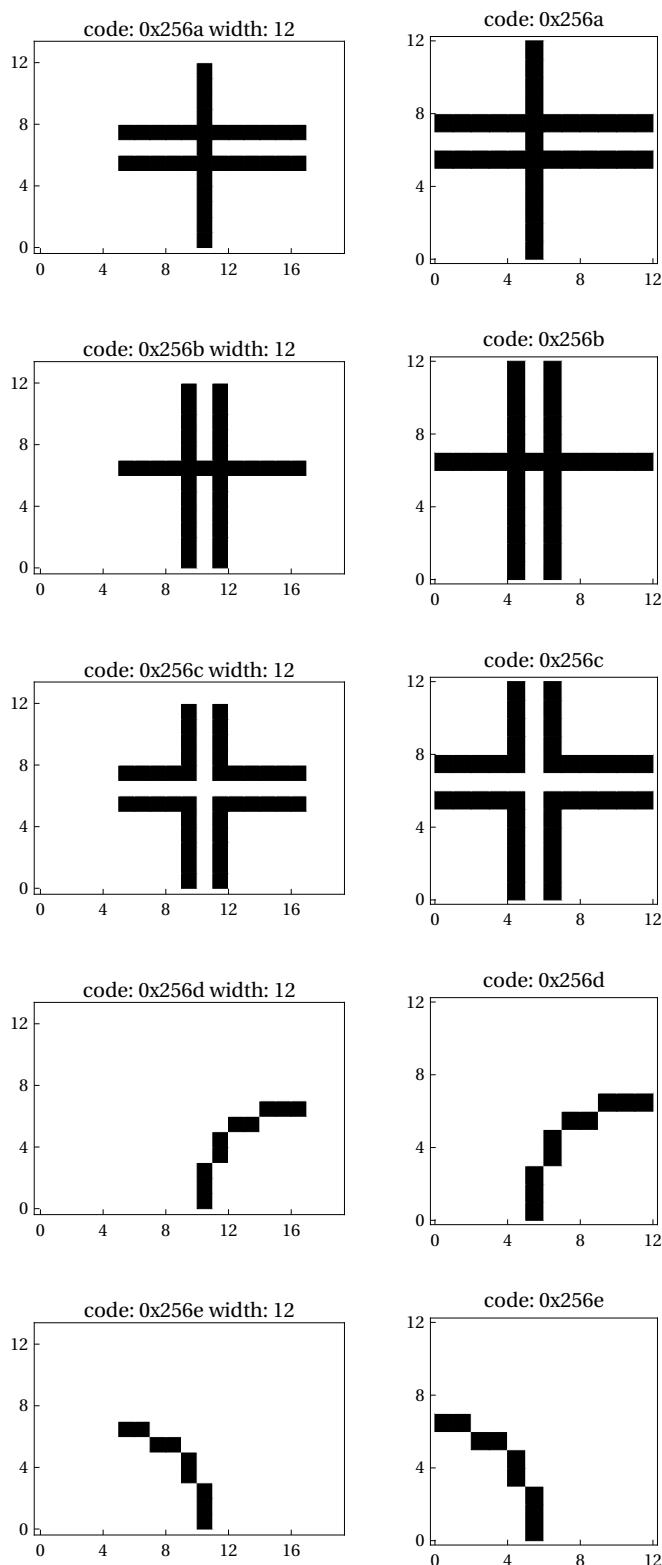


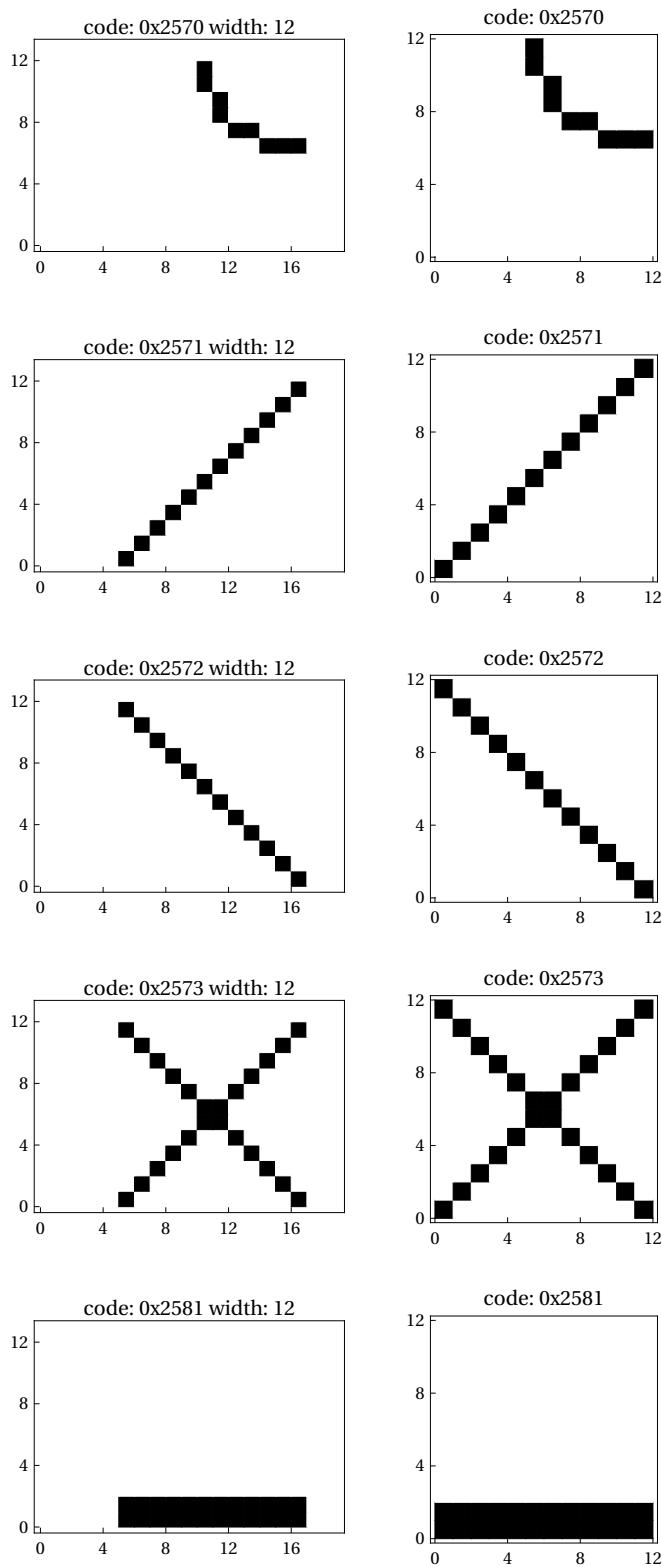


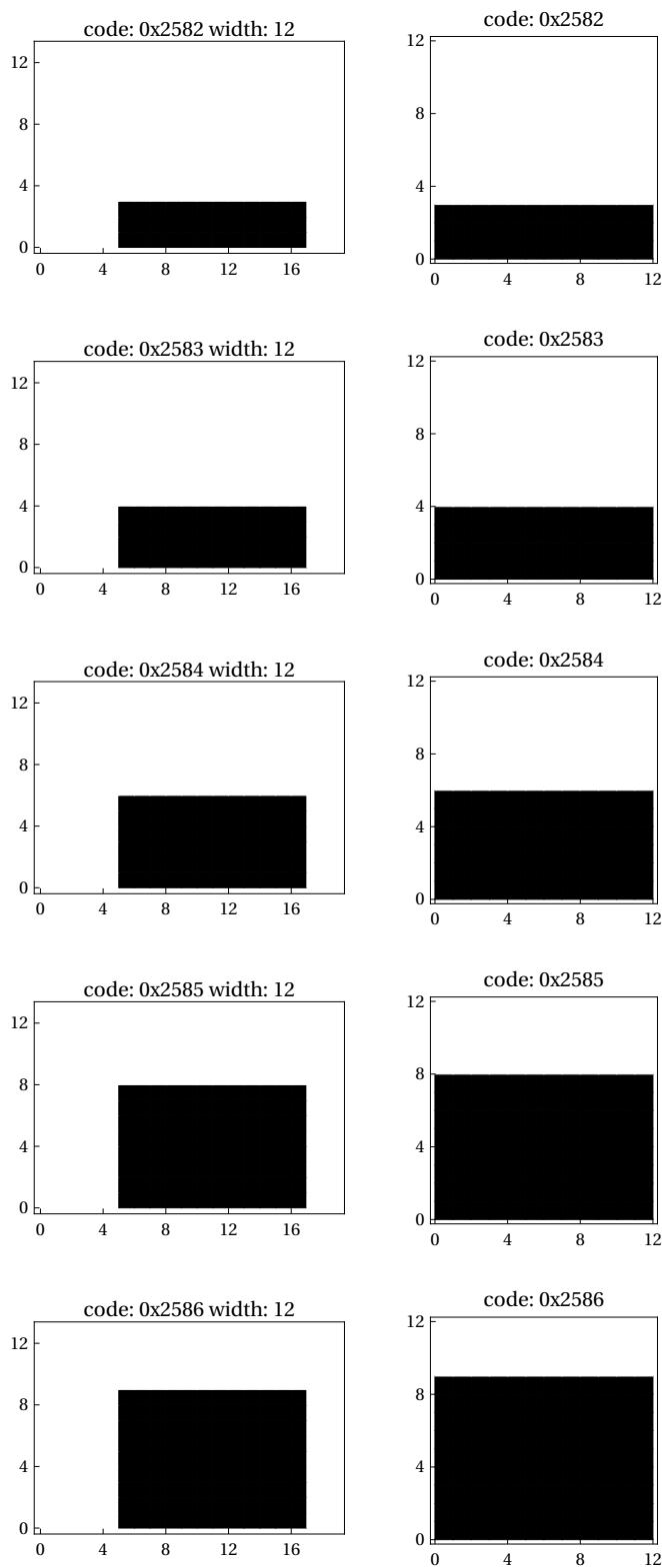


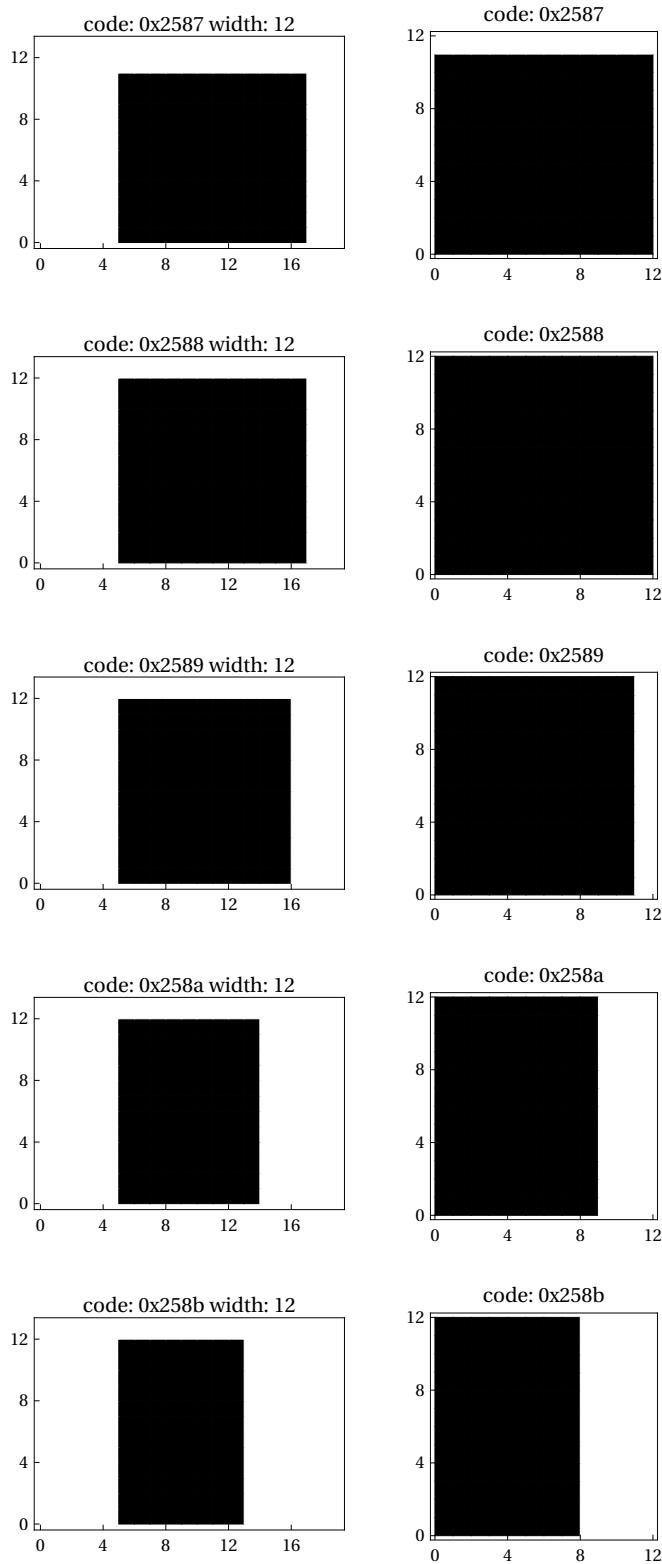


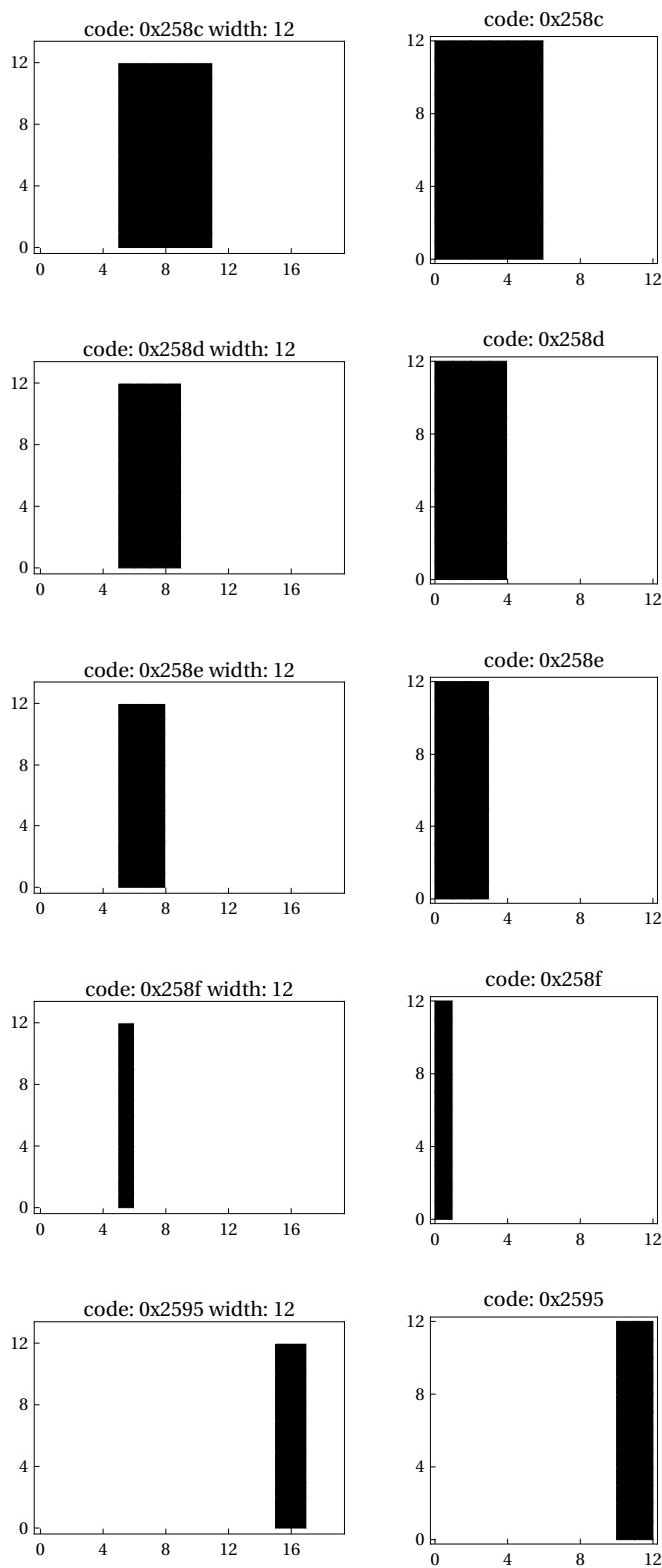


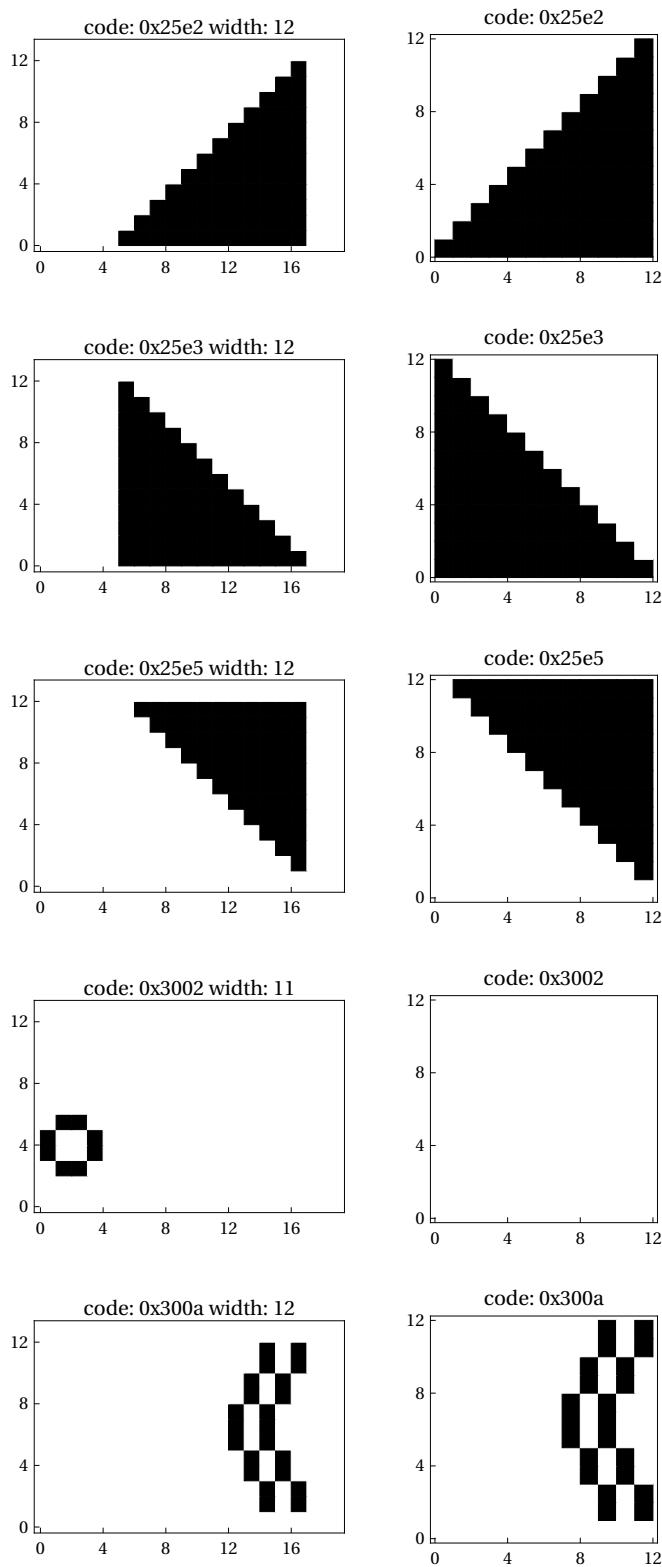


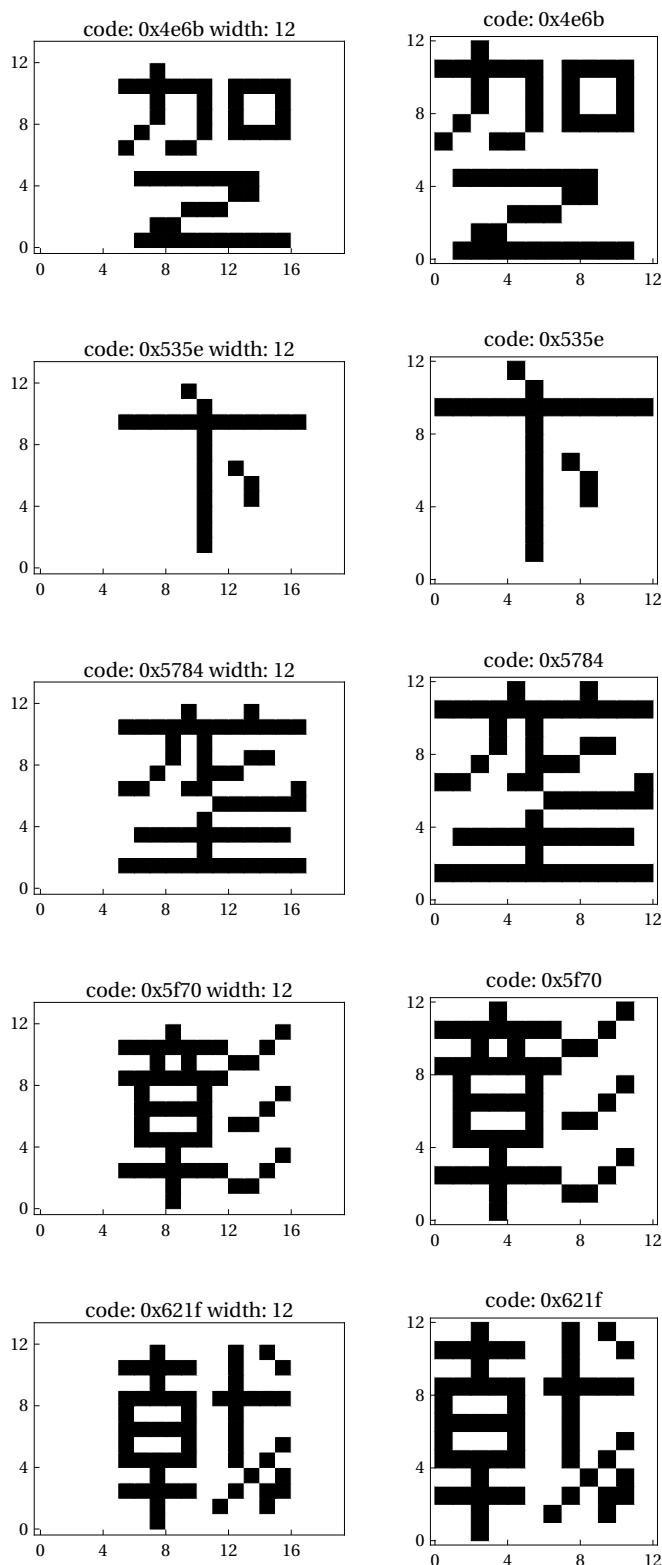


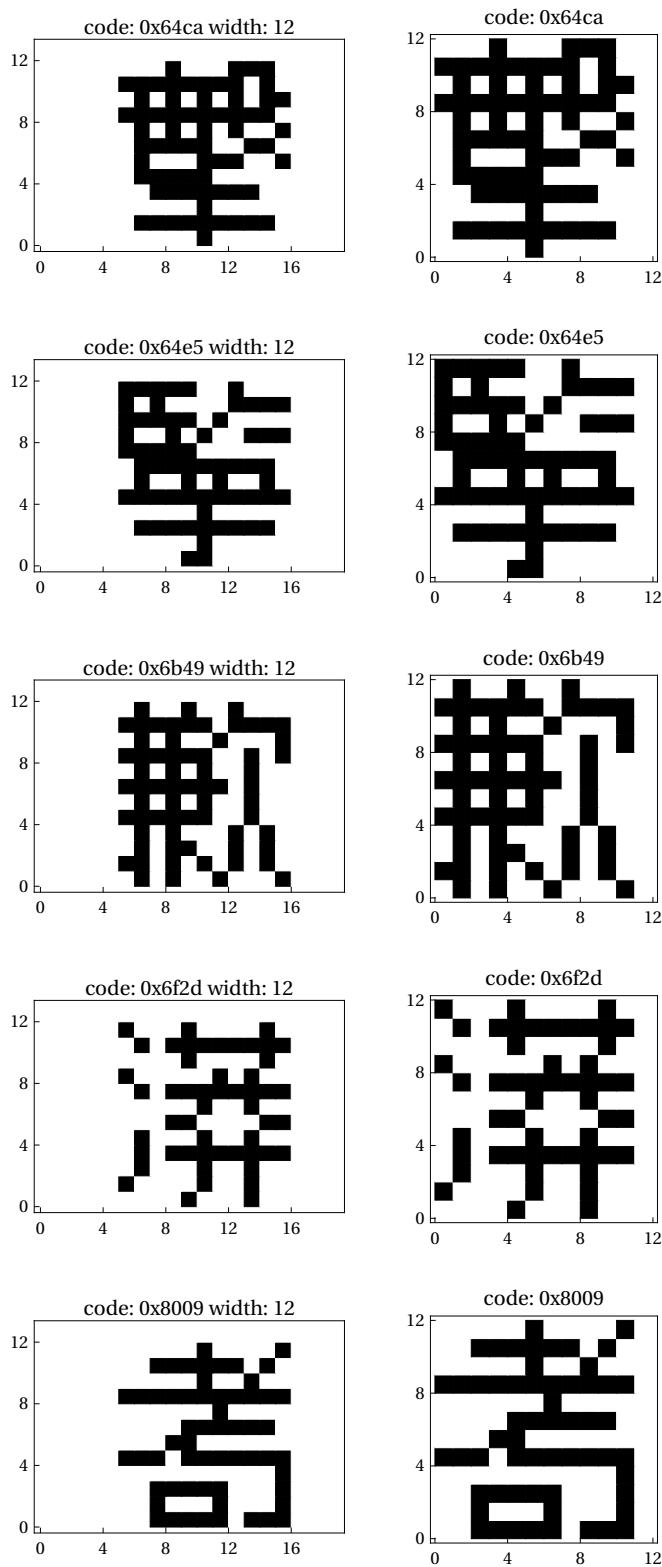


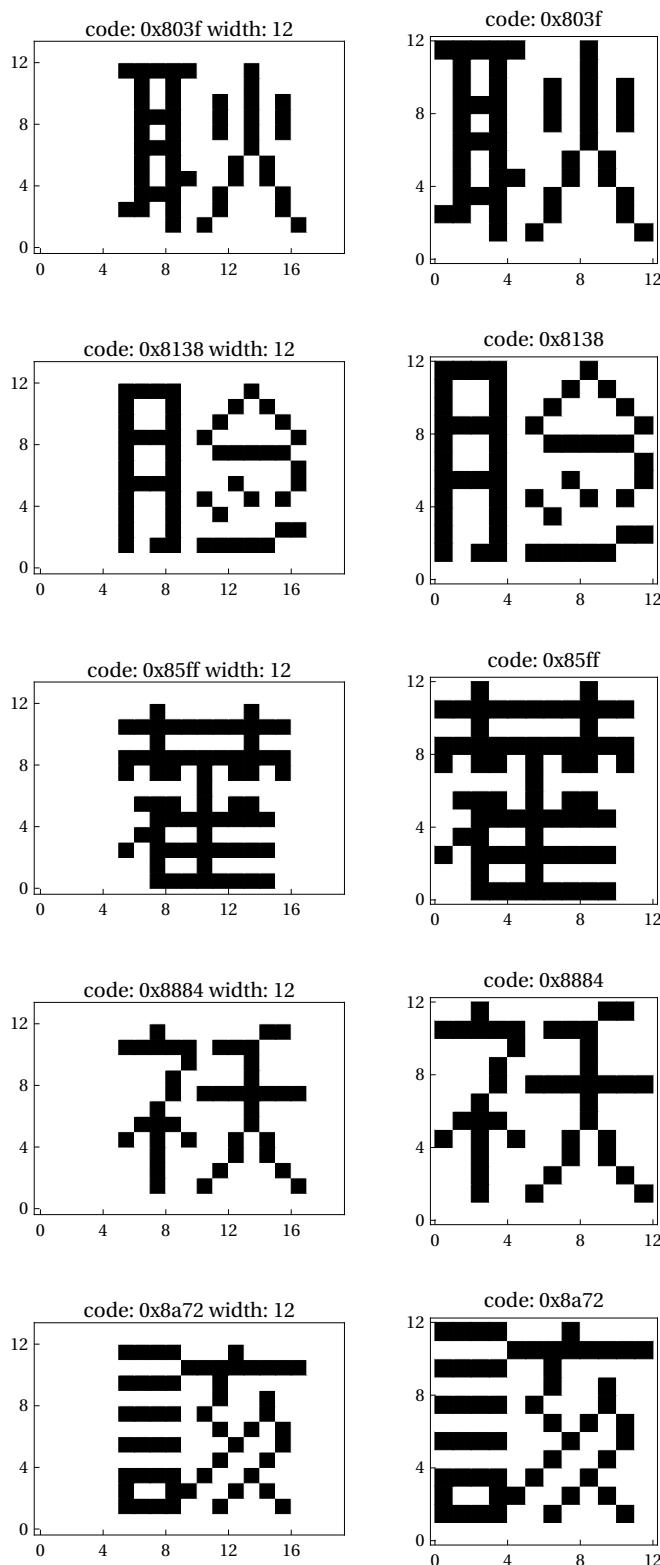


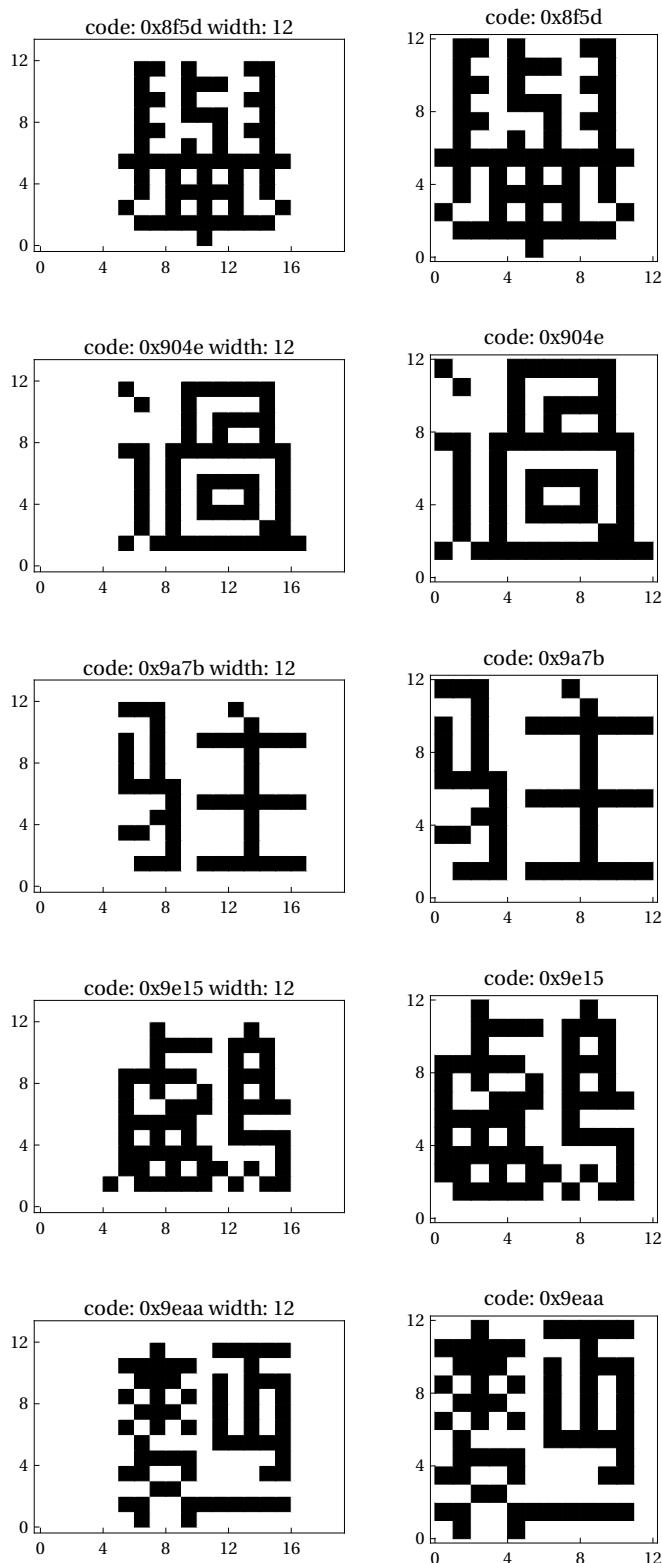


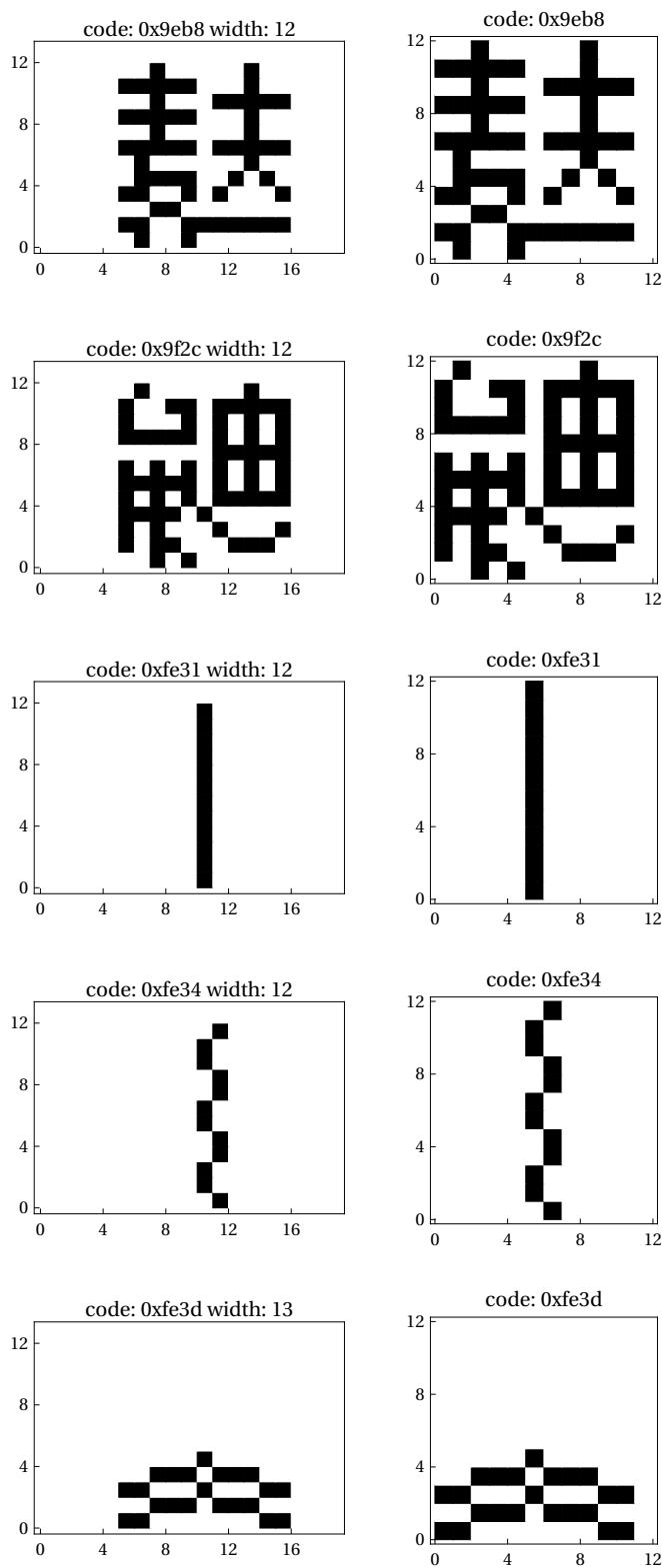


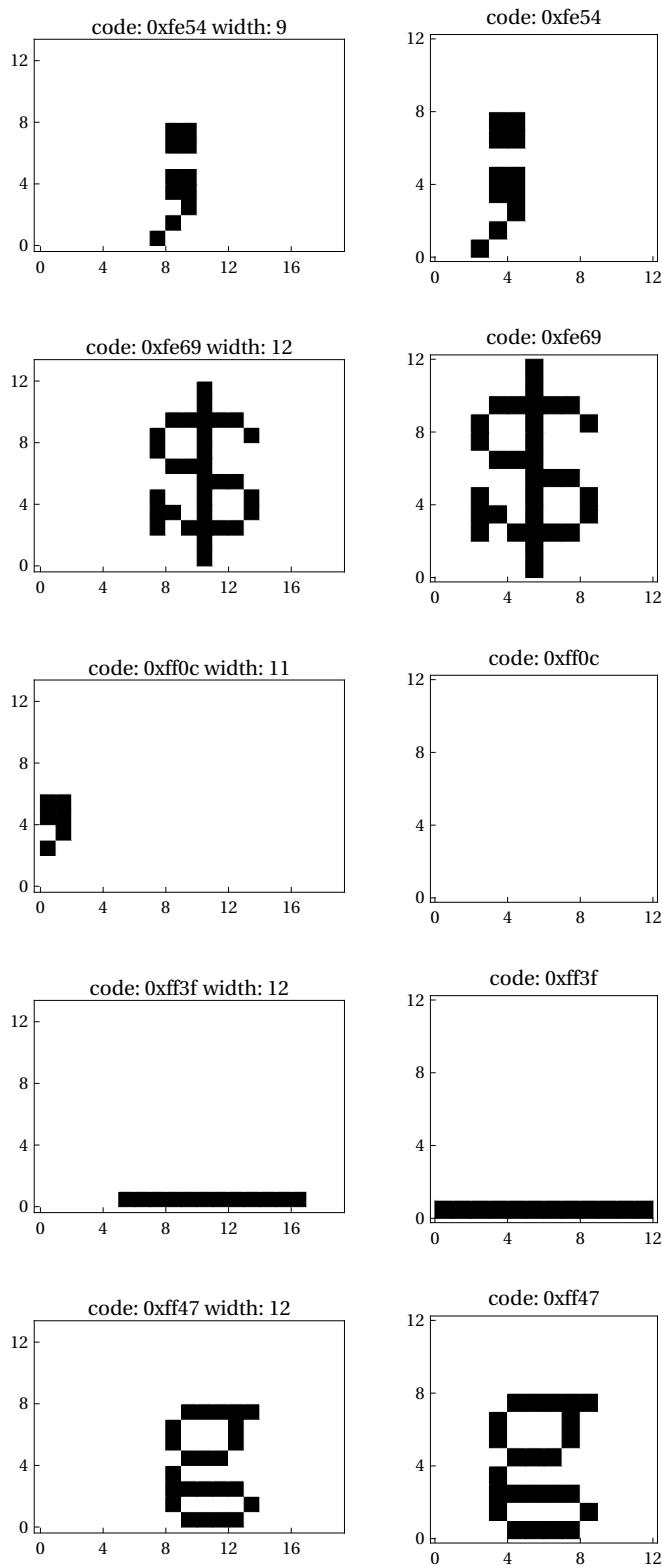


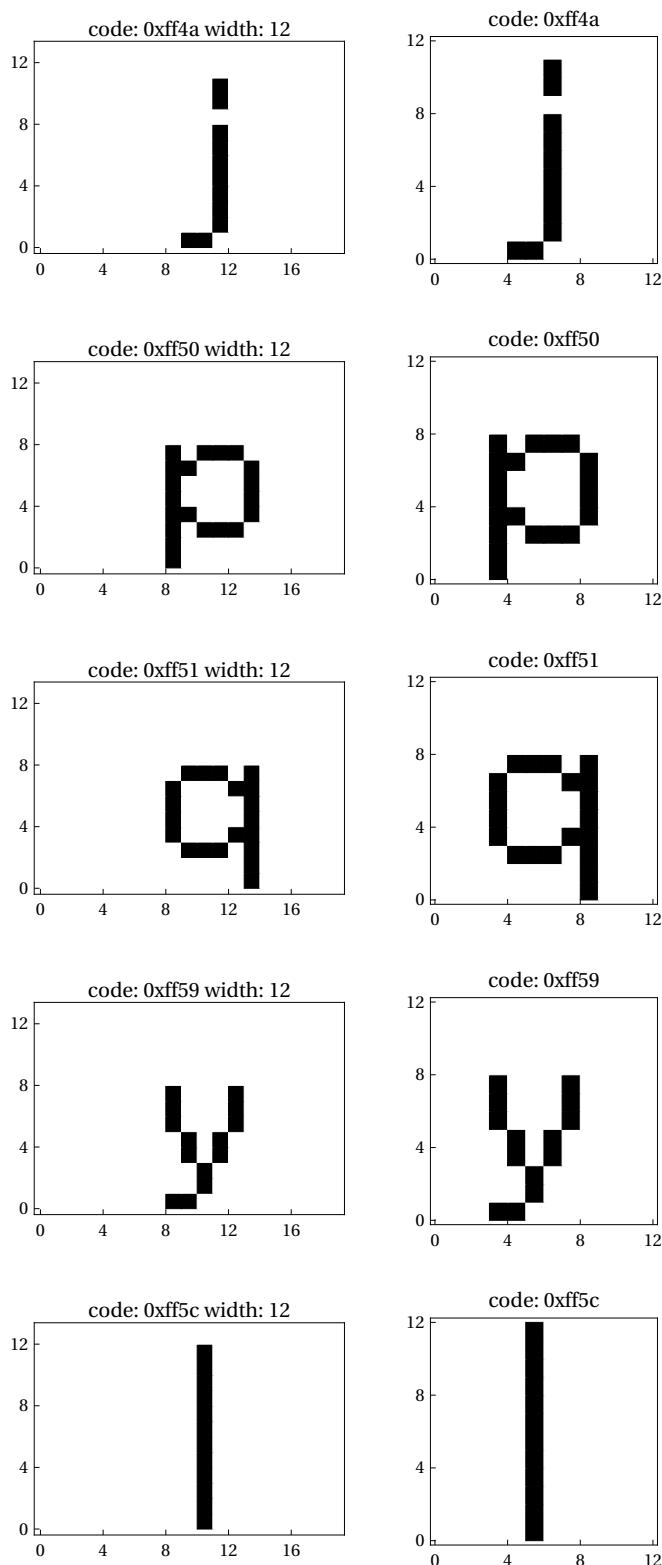


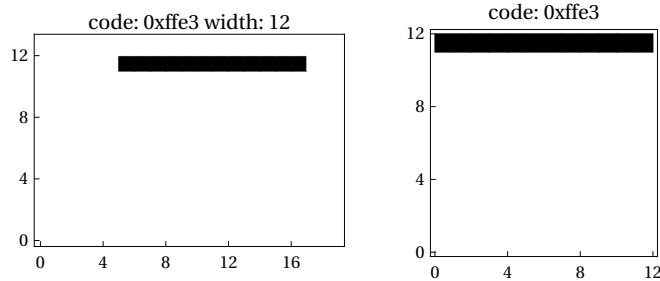












■ All glyphs

```
In[46]:= bitmapgrafix = MapThread[Graphics[Raster[Reverse[#1]], Frame -> True, AspectRatio ->
    Divide @@ Dimensions[#1], FrameTicks -> ({#, #, {}, {}} & [Range[0, 16, 4]]),
    PlotLabel -> ("code: " <> IntegerString[ToExpression[#2], 16, 4] <>
    " width: " <> ToString[#3])] &, {extbitmaps, charcodes, widths}];

In[47]:= bitmaprgrafix = MapThread[Graphics[Raster[Reverse[#1]], Frame -> True, AspectRatio ->
    Divide @@ Dimensions[#1], FrameTicks -> ({#, #, {}, {}} & [Range[0, 16, 4]]),
    PlotLabel -> ("code: " <> IntegerString[ToExpression[#2], 16, 4] <>
    " width: " <> ToString[#3])] &, {trbitmaps, charcodes, widths}];

In[48]:= MapThread[Print[GraphicsGrid[{{##}}]] &,
    Take[#, {1, 2048}] & /@ {bitmapgrafix, bitmaprgrafix}];
```

