segments

a bit of sidetracking

context 2024 meeting

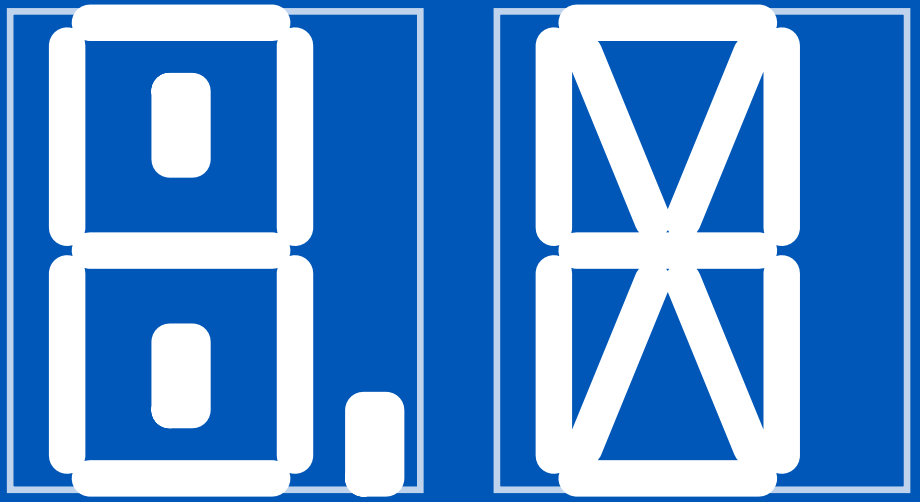
What is needed

- With digits we normally mean 0, 1, 2, 3, 4, 5, 6, 7, 8, 9 and 0.
- When computers are discussed A, b, C, d, E and F are also kind of digits.
- When numbers are to be rendered we need a . and a -.
- The E can be used for exponents.
- On dedicated devices rendering with leds can be optimized.
- Ah, so for clocks we need a : then.

Note: This was an optional talk. On the 'math day' The other talks (Ton, Frits, Willi, Bruce) were way more fun anyway!

How does it look

Here we use a bolder version, so we get for all (regular) elements in a segmented glyph :



The colon is normally a dedicated segmented display and then there is no period, which otherwise is integrated.

What are the challenges

- In Unicode we actually have the digits so let's use them.
- But we need private slots for the hexadecimal digits and other symbols.
- The period is within the bounding box of the segment so we need kerning.
- Contrary to for instance Kaktovik and Riven numerals we don't need a converter.
- Instead we just use a dynamic feature.

How is is done

- We define the private slots by name so we can access then from anywhere.
- We define a MetaFun macro that renders the glyphs.
- We register the relevant replacement glyphs in MetaFun.
- We register some font features in Lua: substitution, kerning and a ligature.
- At the TEX end the replacement glyphs can be hooked into the current font.
- We use a highlight to (temporarily) enable the feature(s).

How is is used

```
\useMPlibrary[segments]

\definefontfeature
  [default]
  [default]
  [metapost=segments]
% [metapost={category=segments,weight=2.0,offset=.2}]
```

Because dejavu is somewhat bold here we used the bold definition. You can also slant the shapes. The module defines a highlight:

```
\definehighlight
  [Digits]
  [style=\addfeature{segmentdigits}]
```

How does the code look

Here is the basic repertoire (the space is a regular one):

digits 0 1 2 3 4 5 6 7 8 9

letters A B C D E F

others - . :

extra [ ] ONE EXE

Let's have a look at:

```
1 meta-imp-segments.mkxl
```