## Step 2

```
- Is the readability problem solved?

- What kind of parallelism can still be expressed?

- Who is the public targeted by this "language"?

- Is this language extensible enough to support new features?
  What is the price for the developer?
```

- Not completely : it is now more easy the operations which require to turn on or off one PIN, but the more complex register states have become less understandable due to the lack of binary operations. However, the rest is more readable.
- The kind of register parallelism permitted by binary operators is not available anymore. What is left is some kind of algorithmic parallelism, for example by ligthing up common cathodes on the 7seg.
- This language is useful for people who want to code but not do the electric system and wiring part. Probably beginners in code with teachers doing the wiring and code-preparing part.
- This language is made of a few C++ constants and functions that interact in a more user-readable way with the Arduino system. It is infinitely extensible, since it is still C++.
- The developper has to read and understand a documentation that he has not written himself. Furthermore, as we saw, some things such as memory parallelization are now hidden to him. But in a general way, he gains readability and ease of use due to functions that are specific to the targeted system (an Arduino).