

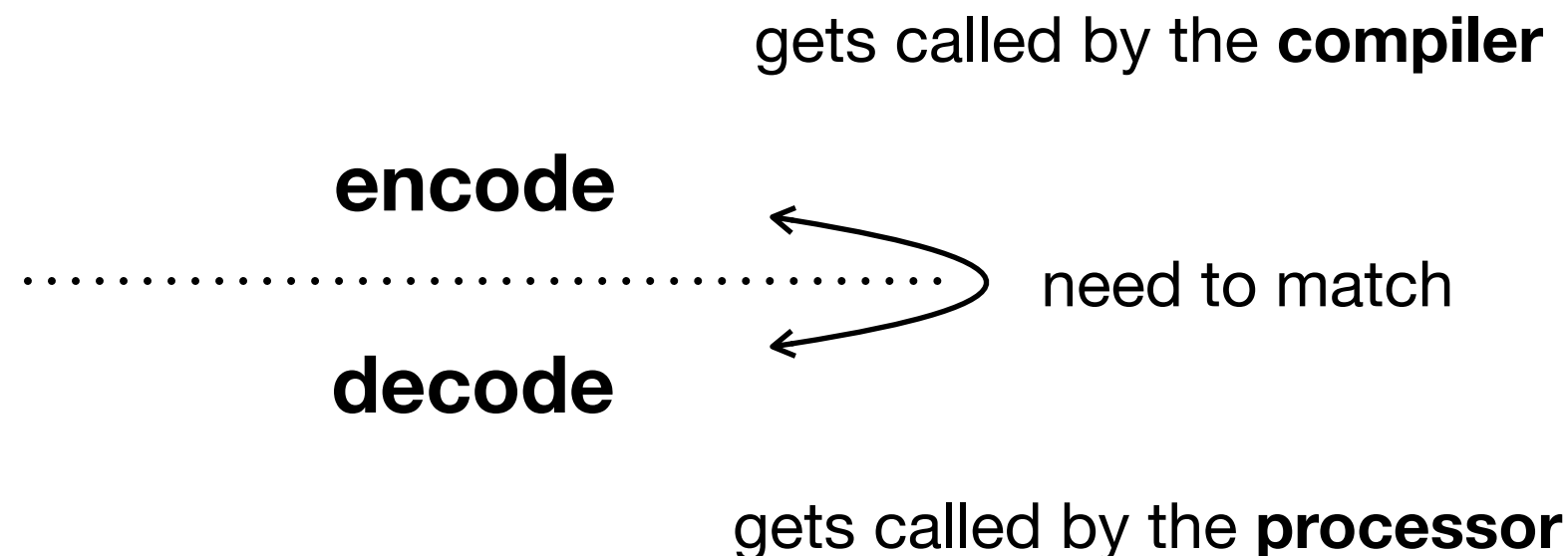


**.\_.inselfie**

- The semantics of the bitwise operators `<<` and `>>` on the `uint64_t` type is that of a *logical shift*.
- In selfie there are library functions that perform shift operations (see [here](#)):
  - `leftshift()`:  $n \ll b = n * 2^b$
  - `rightshift()`:  $n \gg b = n / 2^b$

# Language Operators

- The compiler has to recognize the operator and generate code for it.
- The processor has to understand the instructions encoded by the compiler and executes them.
- We first expand the processor by implementing a new machine instruction that can then be used by the compiler to generate code.



# ...in selfie

- The semantics of the bitwise operators `<<` and `>>` on the `uint64_t` type is that of a *logical shift*.
- In selfie there are library functions that perform shift operations (see [here](#)):
  - `leftshift()`:  $n \ll b = n * 2^b$
  - `rightshift()`:  $n \gg b = n / 2^b$