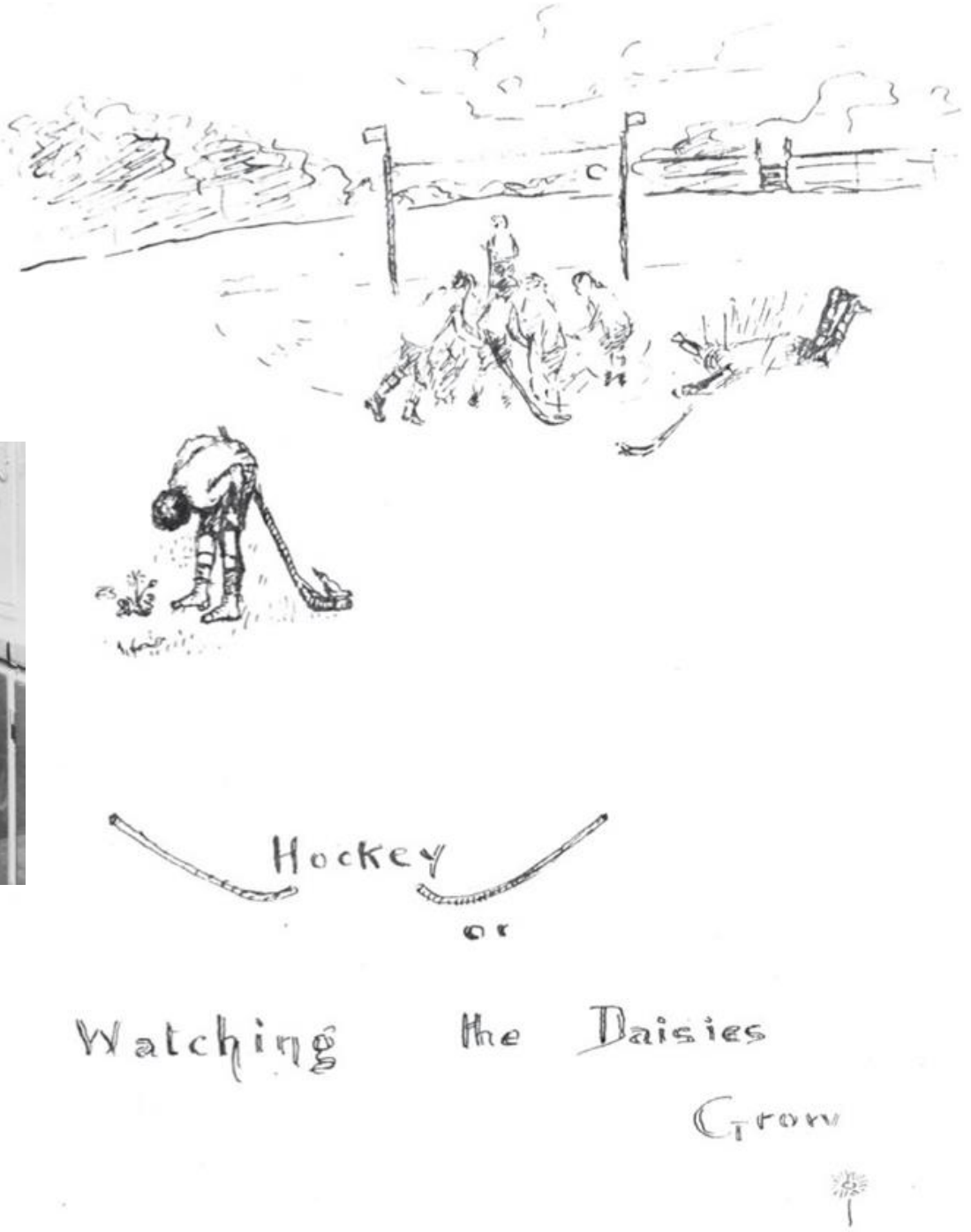
The “Turing Tax”

Discussion exercise

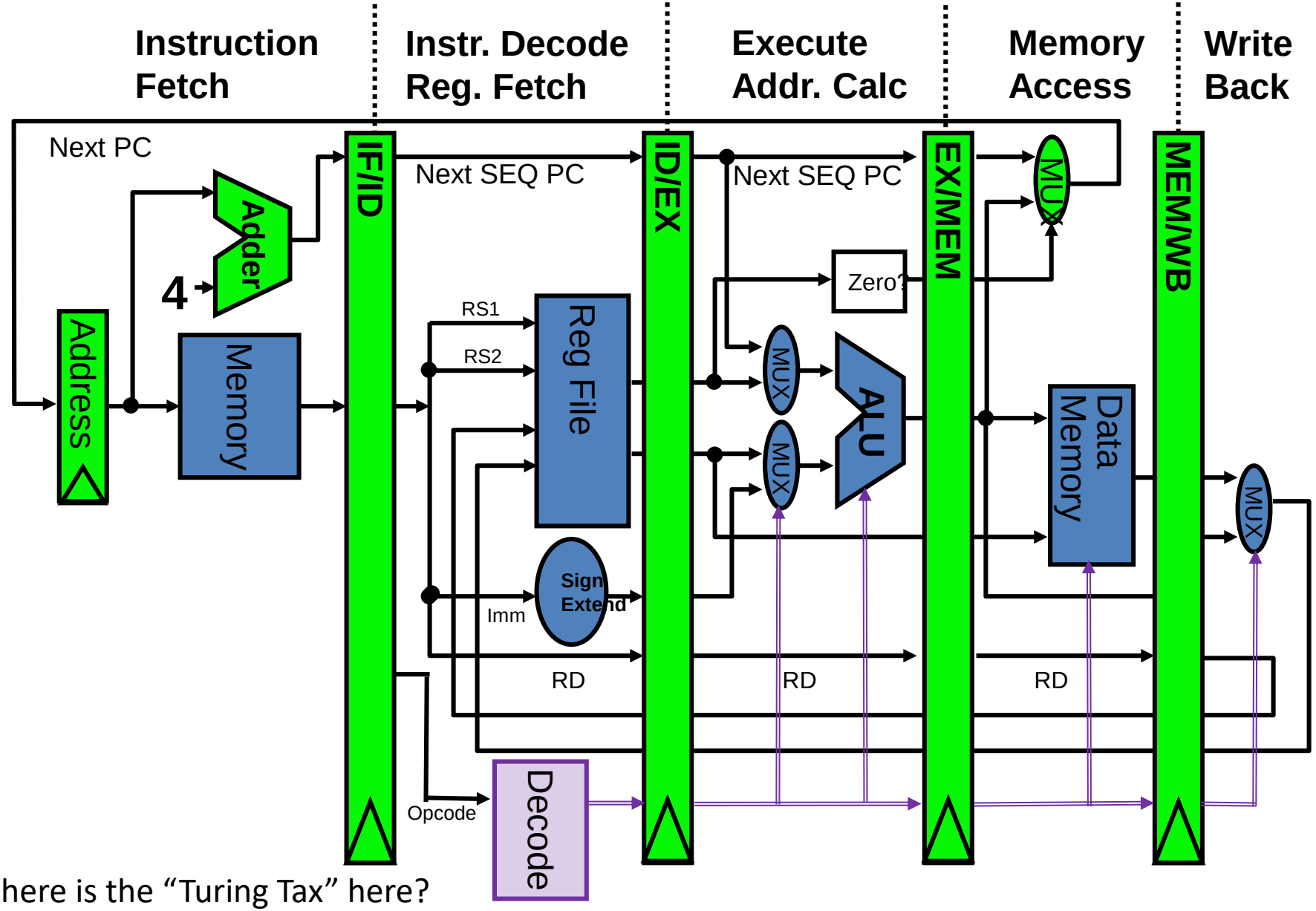
Alan Turing and colleagues working on the Ferranti Mark I Computer in 1951. How intelligent was it? Photograph: Science & Society Picture Library/Getty Images



Turing tax

- Alan Turing realised we could use digital technology to implement any computable function
- He then proposed the idea of a “universal” computing device – a *single* device which, with the right program, can implement any computable function *without further configuration*
- The “Turing Tax” is a term for the overhead (performance, cost, or energy) of universality in this sense
- That is, the performance difference between a special-purpose device and a general-purpose one
- **One of the fundamental questions of computer architecture is to how to reduce the Turing Tax**

Pipelined MIPS Datapath with early branch determination



- Where is the "Turing Tax" here?
- That is – which bits are overhead due to the general-purpose nature of the processor, in contrast to a special-purpose digital design?

Turing tax: instructions

- Instruction fetch
  - Store instructions
  - Fetch them
  - Decode them
  - Maintain PC
  - Handle branches

  - Predict branches
  - Handle branch mis-predictions

Turing tax: data routing

- Forwarding is used to avoid stalls
- Forwarding is switched by multiplexors
- Which are determined by instruction decode

- We might not need all forwarding paths
- We might not need to switch them
- We might place the producer and consumer adjacently, so the wires can be shorter

Turing tax: register access

- Instructions use registers to pass values from one operation to the next
- Each time a register is used, we have to look the value up in the register file

- In a special-purpose machine, we'd use a piece of wire!

Turing tax: configurable ALU

- In our MIPS pipeline, the ALU function is controlled by a signal derived from decoding the instruction
- The ALU is a multipurpose unit – that can add, subtract, multiply etc

- In a special-purpose design we would only have the units we need
- and we'd have just the right number of each kind

Turing tax: avoidance?

What can we do to avoid the Turing Tax?

Caches are “Turing Tax”

Discuss!

The Turing Tax is irrelevant for most applications

Discuss!