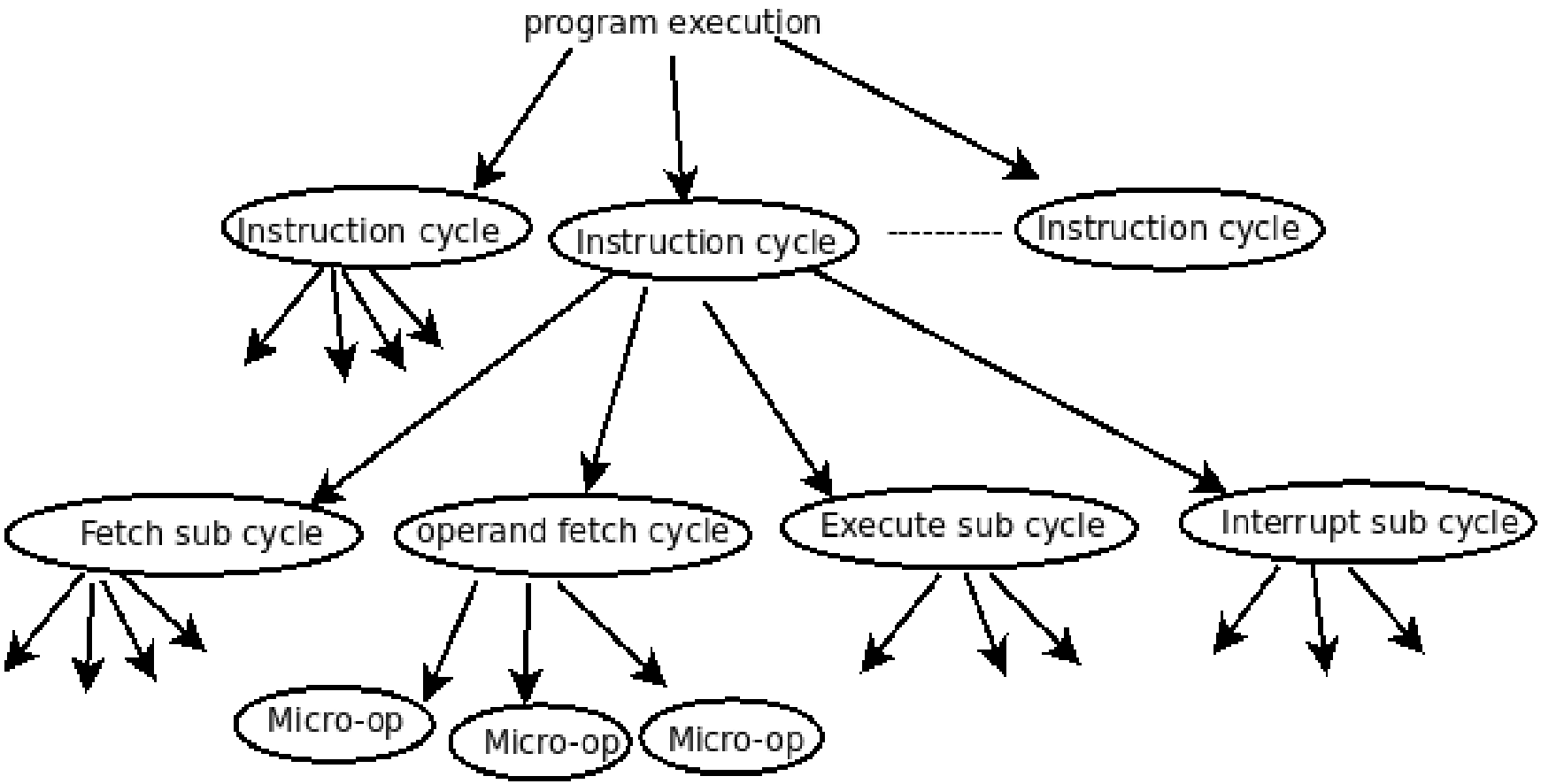
CPU control unit:Micro-operation

- Program execution is nothing but a sequence of instruction cycle execution.
- Each instruction has four sub-instruction cycles: fetch, indirect operand fetch, execution and interrupt.
- Each sub instruction cycle has micro-operations.

Micro-operations



Fetch sub instruction cycle

- In this cycle, instructions are fetched from memory to processor registers.
- Different micro-operations performed in different processor time units or cpu clock pulses.

T1: MAR <---PC

T2: MBR <--- Memory

PC <--- (PC)+1

T3: IR <---MBR

Indirect sub instruction cycle

- After the instruction decode, the processor knows which operation cpu has to perform and how many operands needs for instruction execution.
- In instruction msb bit for in which addressing mode operand fetch from memory.
- If MSB bit is 0, then operand fetched from memory in direct addressing mode.
- If MSB bit is 1, then operand fetched from memory in indirect addressing mode.

T1: MAR <--- (IR(address))

T2: MBR <--- Memory

T3: IR(address) <--- (MBR(address))

Execution sub instruction cycle

- Some of the micro-operations for ADD R1, X instruction execution.
- Two operands, one from register and another from memory.

T1: MAR <--- (IR(address))

T2: MBR <--- memory

T3: R1 <--- (R1) + (MBR)

Interrupt sub instruction cycle

- The processor checks if any interrupt occurred or not. If occurs then these micro-operations are performed.

T1: MBR <--- (PC)

T2: MAR <--- Save_address

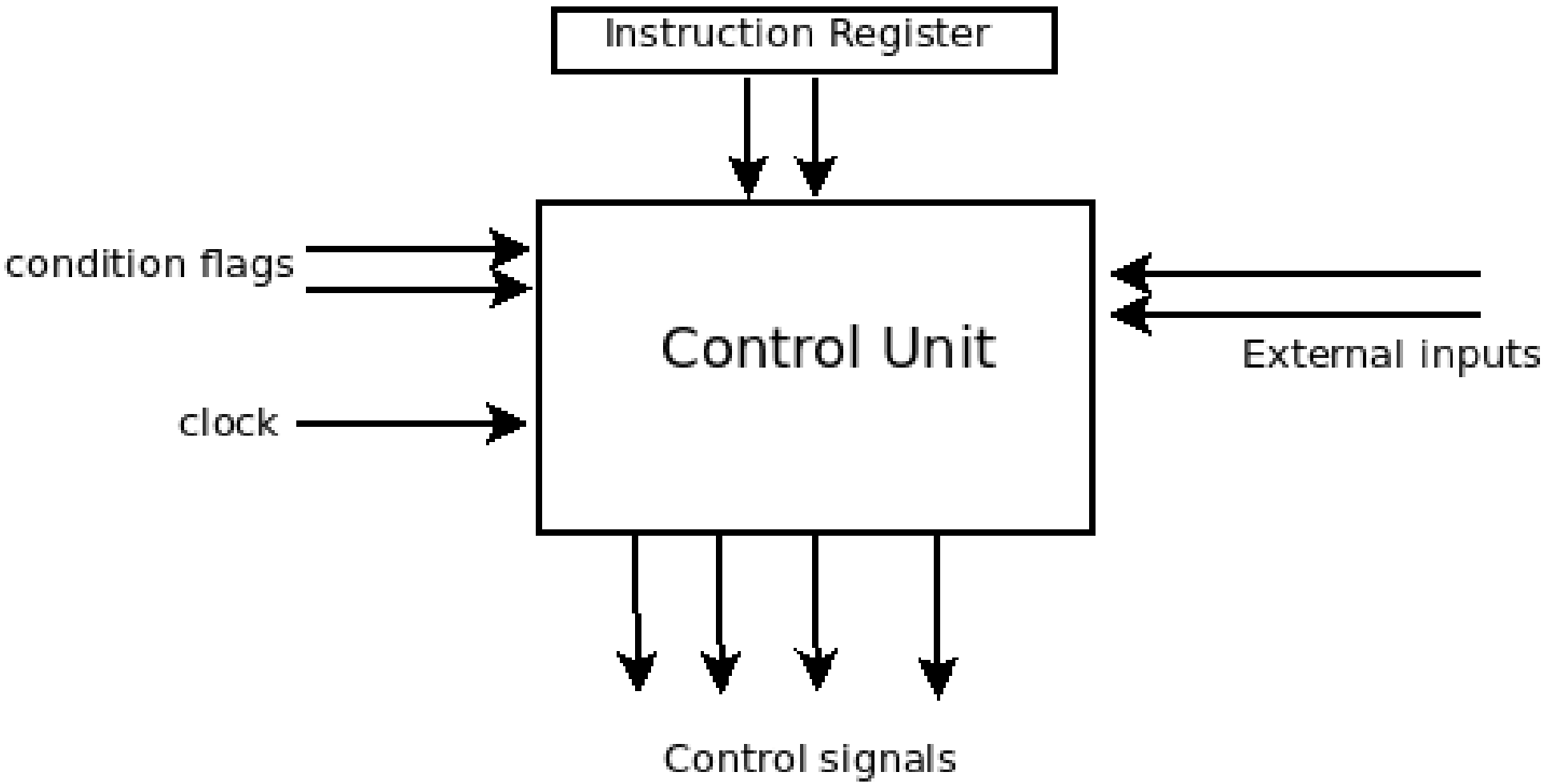
PC <--- Routine_address

T3: Memory <--- (MBR)

Control unit design

- The control unit generates control signals to activate various components in the cpu, like registers, internal bus, ALU, and paths between various components in the cpu.
- Instruction register, external inputs, condition codes and clock are inputs to control unit.
- IR: in this register instruction loaded from memory, decoded by the processor. After instruction decode processor knows which operation has to perform.

Control unit design



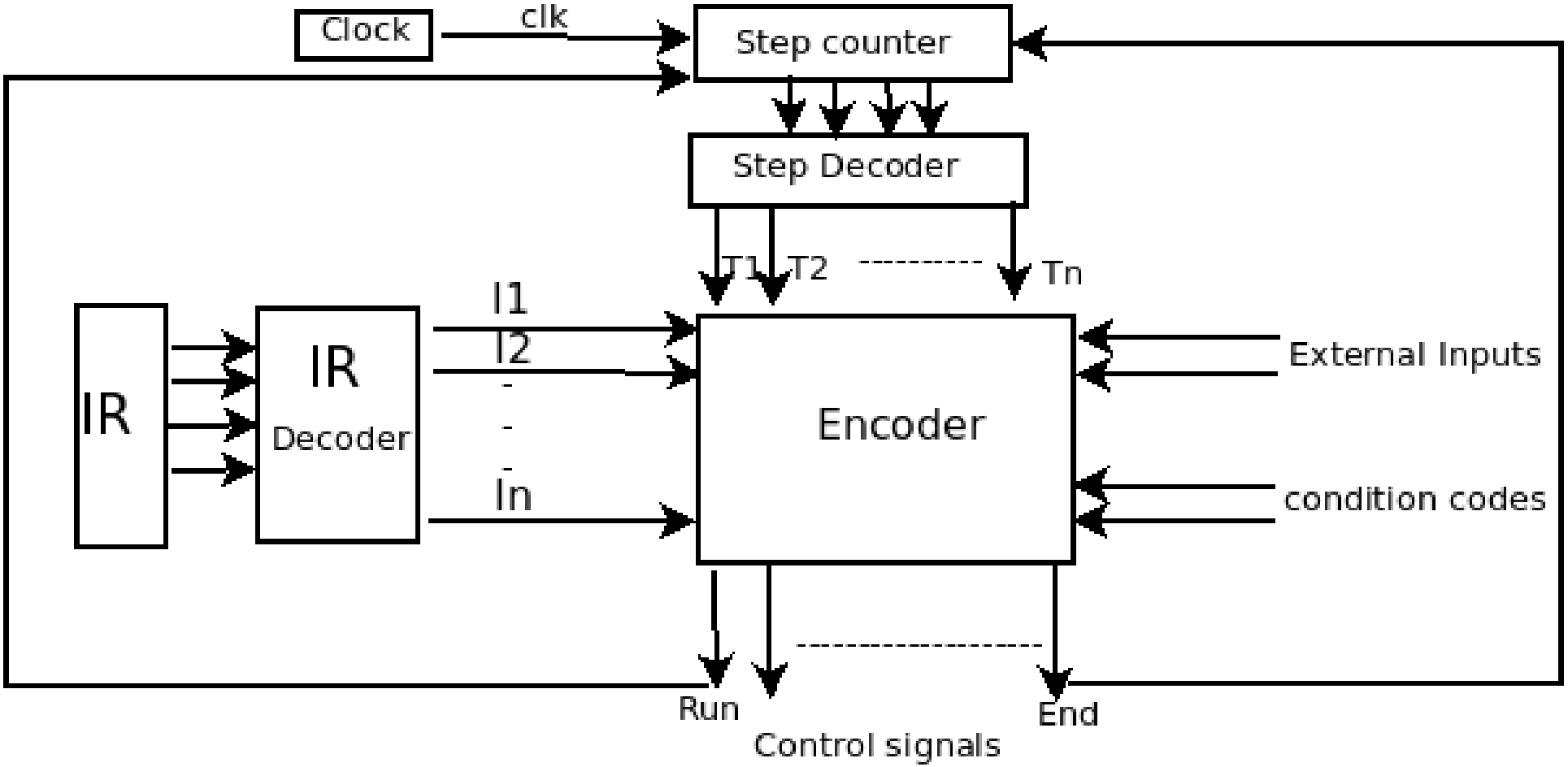
Control unit design

- Condition codes gives status of previous operation. For example, increment and skip if zero(ISZ).
- Instruction effective address is incremented and compared with zero, if ZF is set to 1 then processor skips next instruction execution.
- External inputs: control signals are generated from memory to processor like WMFC.
- Clock: in a processor clock cycle, one micro-operation or set of micro-operations are performed.

Hardwired control unit

- The processor has to generate control signals in the proper sequence for executing instructions.
- For hardwired control design, hardware components like AND gates, OR gates, encoders, and decoders are used.
- A counter is used to keep track of the control steps. Inputs for the hardwired control unit are the clock, instruction register, condition codes, and external inputs like WMFC and interrupts.
- Each step in this sequence is completed in one clock cycle. The step decoder provides a separate signal line for each time unit in the control sequence.
- Instruction register opcode bit inputs for the IR decoder. The output of the instruction decoder consists of a separate line for each machine instruction.

Hardwired control unit



Hardwired control unit

- For any instruction loaded in the IR register, one of the output lines from I1 through In is set to 1 and the other lines are set to 0, which means not active.
- The encoder combines all the input signals and generates separate control signals like Yin, PC Out, ADD, Run, End, etc.

Zin = T1+T6.ADD+T4.BR+-----

- The End control signal starts a new instruction fetch cycle by resetting the control step counter to its starting value.
- When the RUN control is called and set to 1, it means the step counter is incremented by one at the end of every clock cycle. When RUN reaches 0, the step counter stops counting.

Microprogrammed control unit

- The microprogrammed control unit also generates a sequence of control signals for instruction execution. It operates on the basis of a complex instruction set computer (CISC).
- The programme which is used to generate control signals is called "Microprogrammed". This microprogrammed is placed on the processor chip. It is a special type of memory, also known as control memory.
- Each micro-program contains a set of microinstructions. Each microinstruction or control word contains different bit patterns. Contorl words contain binary 1’s and 0’s and, based on control word bits, there are different control signals generated. A micro-routine is nothing but a sequence of microinstuctions.

Microprogrammed control unit

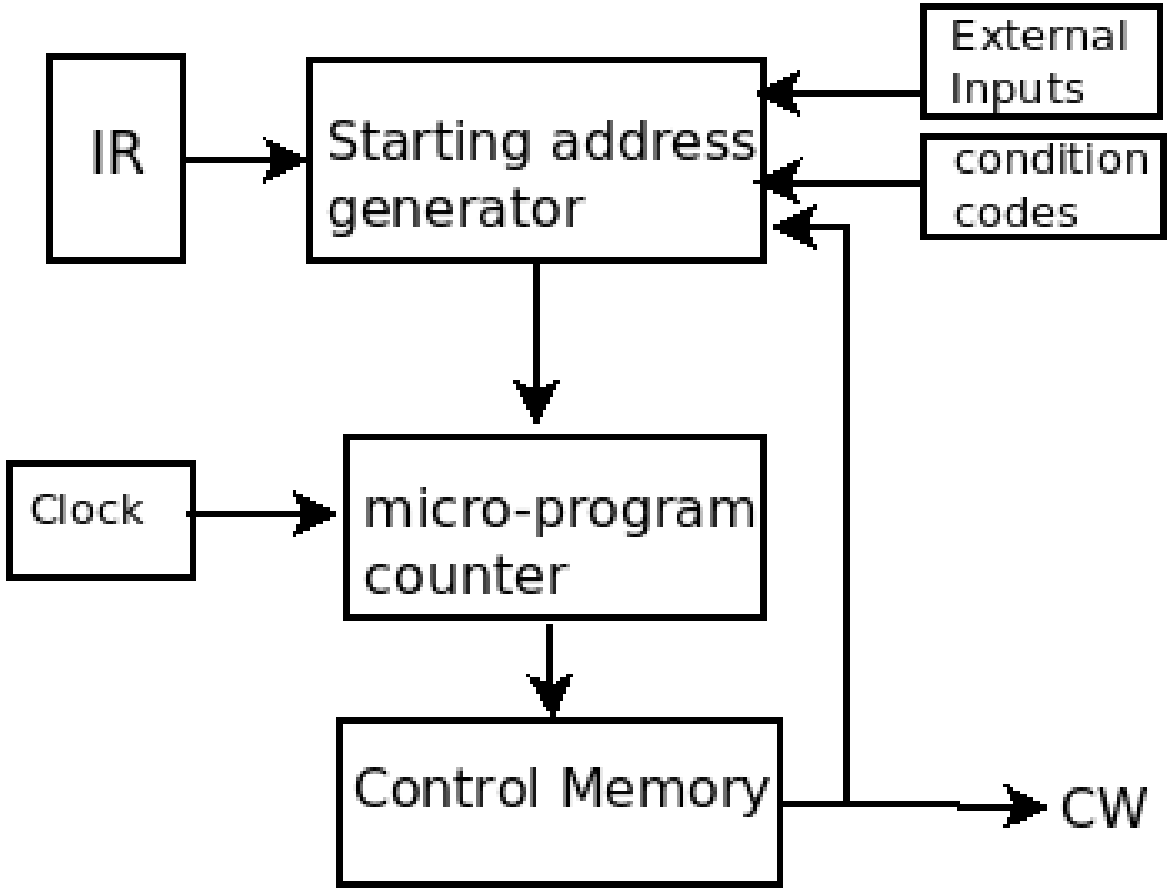


Figure 4 Microprogrammed control unit

Microprogrammed control unit

- The instruction register, which contains the currently executed instruction, external inputs from the system bus such as WMFC, interrupts, and clock are all inputs to the microprogrammed control unit.
- The Statring address generator fetches instructions from the instruction register and instructions are decoded. After that, the first micro-routine or instruction starting address is loaded into the micro-program counter .
- Based on this micro-program counter address control word generated from control memory, with this control signal, the micro-routine starts execution.
- The micro-program counter increments automatically with the clock so that it can read the sequence of control words of a micro-routine.
- If all micro-routines are successfully completed, then the end bit is set to 1, which means the microprogrammed control unit can start the next instruction execution.