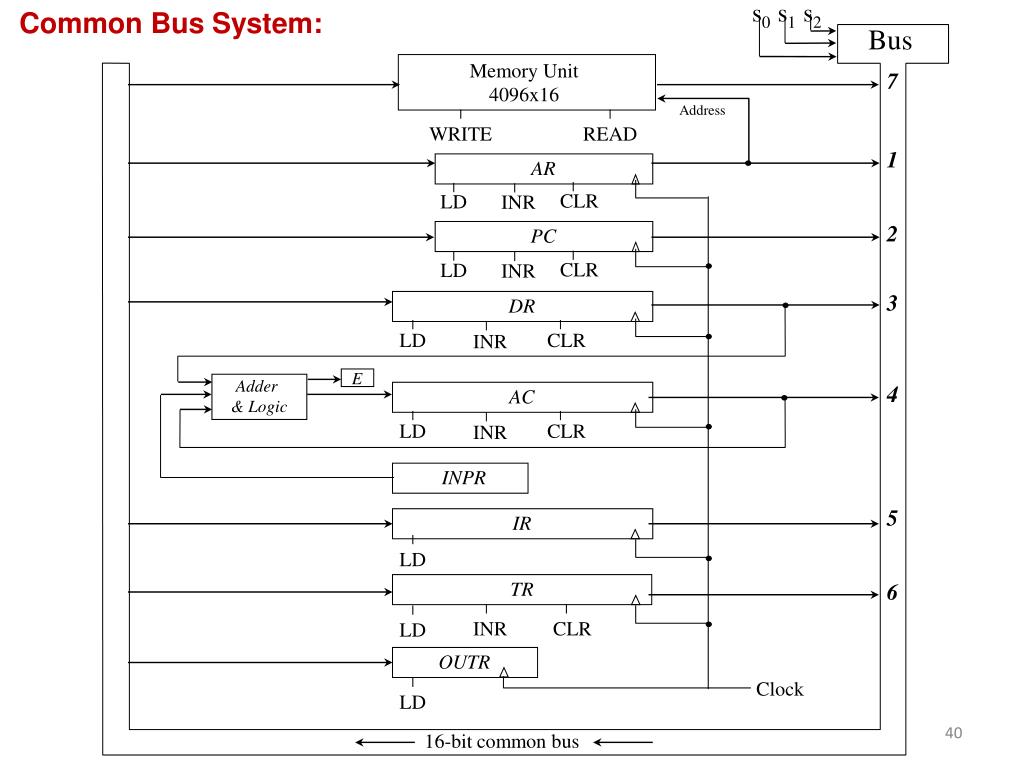
**Computer Registers**

* Computer instructions are normally stored in consecutive memory locations and are executed sequentially one at a time.
* Control reads an instruction from a specific address in memory and execute it and similarly continue this process one by one.
* Counter needed to calculate the address of next instruction after execution of current instruction.
* Need register to store instruction code after read from memory.
* Computer need processor registers for manipulating data and a register for holding a memory address.
* Some of the registers used in computer system are:
  1. Data Register (DR): it holds the operand (data) read from memory.
  2. Accumulator Register (AC): it is general purpose processing register.
  3. Instruction Register (IR): to hold the instruction read from memory.
  4. Temporary Register (TR): to hold a temporary data during processing.
  5. Address Register (AR): hold a memory address, 12-bit width address.
  6. Program Counter (PC):
     + hold the address of the next instruction to be read from memory after the current instruction is executed.
     + Instruction words are read and executed in sequence unless a branch instruction is encountered.
     + A branch instruction calls for a transfer to a nonconsecutive instruction in the program.
     + The address part of a branch instruction is transferred to PC to become the address of the next instruction.
     + To read instruction, memory read cycle is initiated, and PC is incremented by one (next instruction fetch)
  7. Input Register (INPR): receive an 8-bit character from an input device.
  8. Output Register (OUTR): hold an 8-bit character for an output device.

|  |  |  |  |
| --- | --- | --- | --- |
| **Register Symbol** | **No. of bits** | **Register Name** | **Register Function** |
| DR | 16 | Data Register | Hold memory operand |
| AR | 12 | Address Register | Hold address for memory |
| AC | 16 | Accumulator | Processor register |
| IR | 16 | Instruction Register | Hold instruction code |
| PC | 12 | Program Counter | Hold address of next instruction |
| TR | 16 | Temporary Register | Hold temporary data |
| INPR | 8 | Input Register | Hold input character |
| OUTR | 8 | Output Register | Hold output character |

**Common Bus System**

* The basic computer has eight registers, a memory unit, and a control unit.
* Paths must be provided to transfer information from one register to another register and between memory and register.
* A more efficient scheme for transferring information in a system with many registers is to use a common bus.
* The connection of the registers and memory of the basic computer to a common bus system.
  + The output of seven registers and memory are connected to the common bus.
  + The specific output is selected by MUX (S0, S1, S2)
    - Memory (7), AR(1), PC(2), DR(3), AC(4), IR(5), TR(6).
    - When LD (load input) is enable, the particular register receives the data from the bus.
    - Control inputs: LD, INC, CLR, Write, Read
  + **Control Variables:**
    - Various control variables are used to select:
      * The path of information; and
      * The operation of the registers.
  + **Selection Variables:**
    - Used to specify a register whose output is connected to the common bus at any given time.
    - To select one register out of 8, we need 3 selection variables.
    - For example, if **S2S1S0 = 011**, the output of DR is directed to the common bus.
  + **Load Input:**
    - Enables the input of a register connected to the common bus.
    - When LD = 1 for a register, the data on the common bus is write into the register during the next clock pulse transition.
  + **Increment Input (INR):**
    - Increments the content of the register.
  + **Clear Input (CLR):**
    - Clear the content of a register to zero.
  + When the contents of AR or PC (12-bit) are applied to the 16-bit common bus, the four most significant bits are set to zero.
  + When AR, or PC receives information from the bus, only the 12 least significant bits are transferred to the register.
  + Both INPR and OUTR use only 8 least significant bit of the bus.



**Fig: Common Bus System**