**SAP-1 Architecture**

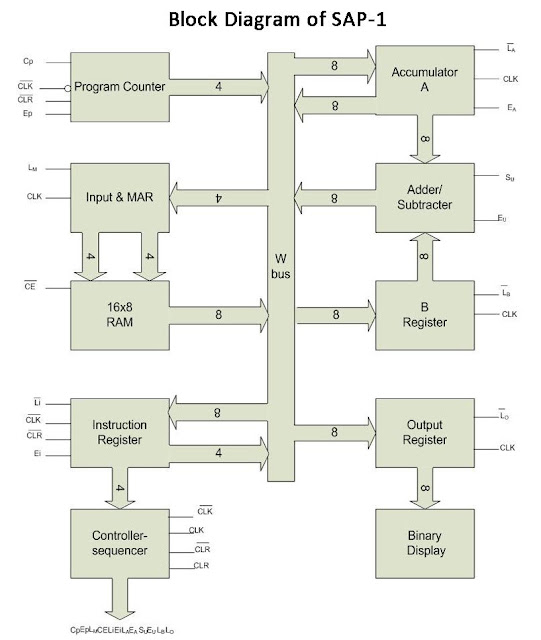
[](https://1.bp.blogspot.com/-jcMswG3yvwo/VmacKcGNGcI/AAAAAAAAAw4/GO51tD-s75g/s1600/image1.jpeg)

Figure: Architecture of SAP-1 Microprocessor/Computer

SAP1 is a basic computer architecture that was designed to demonstrate the fundamental concepts of computer design, including instruction set architecture, memory organization, and the basic operation of a computer. It has a limited instruction set, with instructions for arithmetic operations, load/store operations, and branching. The architecture was designed to be simple, making it easy for students to understand the basic workings of a computer.

SAP1 is used as an educational tool in computer science and computer engineering courses to teach students about computer architecture, assembly language programming, and computer organization. It provides a foundation for learning about more complex computer architectures and helps students understand the relationship between software and hardware in a computer system.

The components of SAP 1 are explained below:

**Program Counter**

* It counts from 0000 to 1111.
* It signals the memory address of next instruction to be fetched and executed.

**Inputs and MAR (Memory Address Register)**

* During a computer run, the address in PC is latched into Memory Address Register (MAR).

**The RAM**

* The program code to be executed and data for SAP-1 computer is stored here.
* During a computer run, the RAM receives 4-bit addresses from MAR and a read operation is performed. Hence, the instruction or data word stored in RAM is placed on the W bus for use by some other part of the computer.
* It is asynchronous RAM, which means that the output data is available as soon as valid address and control signal are applied.

**Instruction Register**

* IR contains the instruction (composed of OPCODE+ADDRESS) to be executed by SAP1 computer.

**Controller-Sequencer**

* It generates the control signals for each block so that actions occur in desired sequence. CLK signal is used to synchronize the overall operation of the SAP1 computer.
* A 12-bit word comes out of the Controller-Sequencer block. This control word determines how the registers will react to the next positive CLK edge.

**Accumulator**

* It is a 8-bit buffer register that stores intermediate results during a computer run.
* It is always one of the operands of ADD, SUB and OUT instructions.

**Adder/Subtractor**

* It is a 2's complement adder-subtractor.
* This module is asynchronous (unclocked), which means that its contents can change as soon as the input words change.

**B-register**

* It is 8-bit buffer register which is primarily used to hold the other operand (one operand is always accumulator) of mathematical operations.

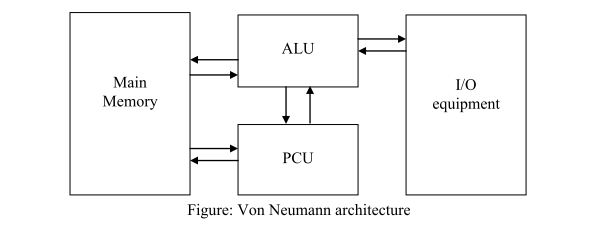
**Output Register**

* This registers hold the output of OUT instruction.

**Binary Display**

* It is a row of eight LEDs to show the contents of output register.
* Binary display unit is the output device for the SAP-1 microprocessor.

**Von Neumann Architecture**

* [](https://2.bp.blogspot.com/-YOsgFepaKDs/VmapnEPTxWI/AAAAAAAAAyI/FjHNH-fmsxQ/s1600/image1.jpeg)Program can be saved like data in the memory unit and can be accessed when needed. This approach is called ‘Stored Program Concept’ and was first adopted by John von Neumann.
* In this architecture, data and instructions are stored in a single set of main memory.
* Instruction fetch and data operation cannot occur at the same time because they share a common bus.
* The program control unit (PCU) reads program instruction, decodes instruction for ALU and determines the sequence of instruction to be executed.
* The ALU performs arithmetic and logical operations.
* It is a basic architecture of today’s computer.

[**Differences between SAP-1 and SAP-2 Architecture**](https://deeprajbhujel.blogspot.com/2015/12/differences-between-sap-1-and-sap-2.html)

|  |  |
| --- | --- |
| **SAP-1** | **SAP-2** |
| It has 8-bit bus. | It has 16-bit bus. |
| PC is 4-bit. | PC is 16-bit. |
| It does not have hexadecimal keyboard encoder. | It has hexadecimal keyboard encoder. |
| It has single input. | It has two input ports. |
| MAR receives 4-bit address from PC. | MAR receives 16-bit address from PC. |
| It does not have ROM. | It has 2 KB ROM. |
| It has 16 Byte memory. | It has 62 KB memory. |
| It does not have MDR. | It has MDR. |
| It has only adder/subtractor. | It has ALU. |
| It does not have flag. | It has 2 flags. |
| It does not have temporary register. | It has temporary register. |
| It has single register (B). | It has 2 registers (B and C). |
| It has single output port. | It has 2 output ports. |
| It has 5 instruction sets. | It has 42 instruction sets. |

[**SAP-2 Architecture**](https://deeprajbhujel.blogspot.com/2015/12/sap-2-architecture.html)

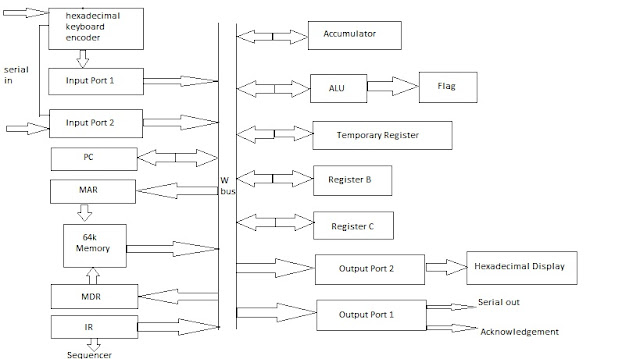
[](https://1.bp.blogspot.com/-NB7mrD1uFlM/VmagQCUG86I/AAAAAAAAAxY/kJGSBlHDONw/s1600/image4.jpeg)

Fig: Architecture of SAP-2 Microprocessor/Computer

SAP2 (Simple As Possible computer architecture with 2 instruction sets) is an extension of the SAP1 architecture, with the addition of a few more instructions to the instruction set. It is also used for educational purposes to teach computer design and architecture.

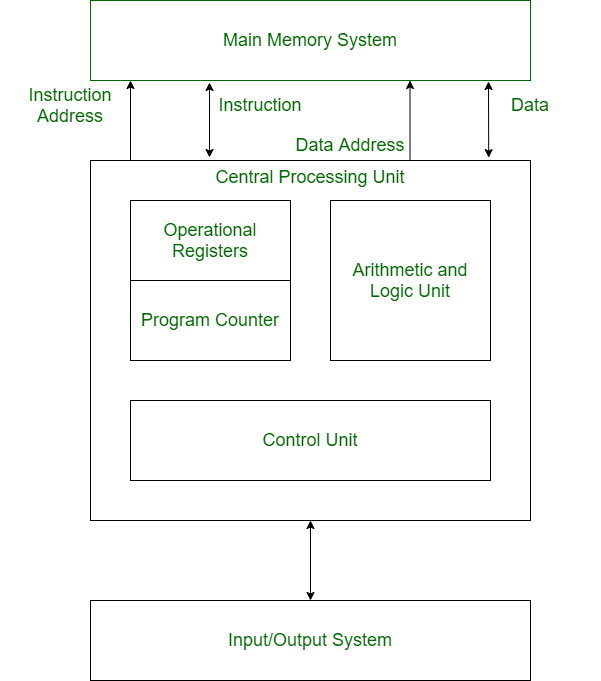
Like SAP1, SAP2 is a basic computer architecture that is designed to be simple and easy to understand. The additional instructions in SAP2 provide more functionality and allow for the execution of more complex programs.

The components of SAP 2 are explained below:

* **Hexadecimal Keyboard Encoder:**The hexadecimal keyboard encoder receives the data from outer environment and converts it into hexadecimal form. The system can understand and send them to the input port.
* **Input Ports:**The SAP-2 contains two input ports which input the data in the system in the most convenient way.
* **PC:**PC is the program counter that holds the address of the next instruction to be fetched. It initializes from 0000H to 1111H during the execution.
* **MAR:**MAR is the memory address register that stores the complete format of the address sent by the program counter. It stores the final address of the memory word that needs some computations.
* **64K Memory:**It contains 64 K memory where data and instruction reside. All the computations are performed relative to the memory.
* **MDR:**MDR is the Memory Data Register which stores the data or operand that is fetched from the memory which is needed for computation.
* **IR:**The IR is the Instruction Register that holds the complete format of the Instruction that is to be executed.
* **Control Sequencer:**It provides necessary timing signals like T0, T1, T2, ….. and control signals providing the direction for executing the program.
* **Accumulator:**The result of all the mathematical operations is stored in accumulator. It is one of the operand of ADD, OUT, SUB instruction. It is also known as processor register.
* **ALU and Flag:**The ALU perform all the arithmetic and logical calculations. The flag reflect the intermediate changes on the values during execution.
* **Temporary register, B, C:**They are the second operand of the mathematical operations. The register B and C is accessible to the programmer.
* **Output Ports:**It consists of two output ports to show the result of OUT instruction.
* **Hexadecimal Display:**Unlike Sap-1 which has binary display, Sap-2 has a hexadecimal display to show outputs in the LEDs.

**Harvard Architecture** is the computer architecture that contains separate storage and separate buses (signal path) for instruction and data. It was basically developed to overcome the bottleneck of Von Neumann Architecture. The main advantage of having separate buses for instruction and data is that the CPU can access instructions and read/write data at the same time.

**Structure of Harvard Architecture:**



*Structure of Harvard Architecture*

**Advantage of Harvard Architecture:**

Harvard architecture has two separate buses for instruction and data. Hence, the CPU can access instructions and read/write data at the same time. This is the major advantage of Harvard architecture.

| **VON NEUMANN ARCHITECTURE** | **HARVARD ARCHITECTURE** |
| --- | --- |
| It is ancient computer architecture based on stored program computer concept. | It is modern computer architecture based on Harvard Mark I relay based model. |
| Same physical memory address is used for instructions and data. | Separate physical memory address is used for instructions and data. |
| There is common bus for data and instruction transfer. | Separate buses are used for transferring data and instruction. |
| Two clock cycles are required to execute single instruction. | An instruction is executed in a single cycle. |
| It is cheaper in cost. | It is costly than Von Neumann Architecture. |
| CPU can not access instructions and read/write at the same time. | CPU can access instructions and read/write at the same time. |
| It is used in personal computers and small computers. | It is used in micro controllers and signal processing. |