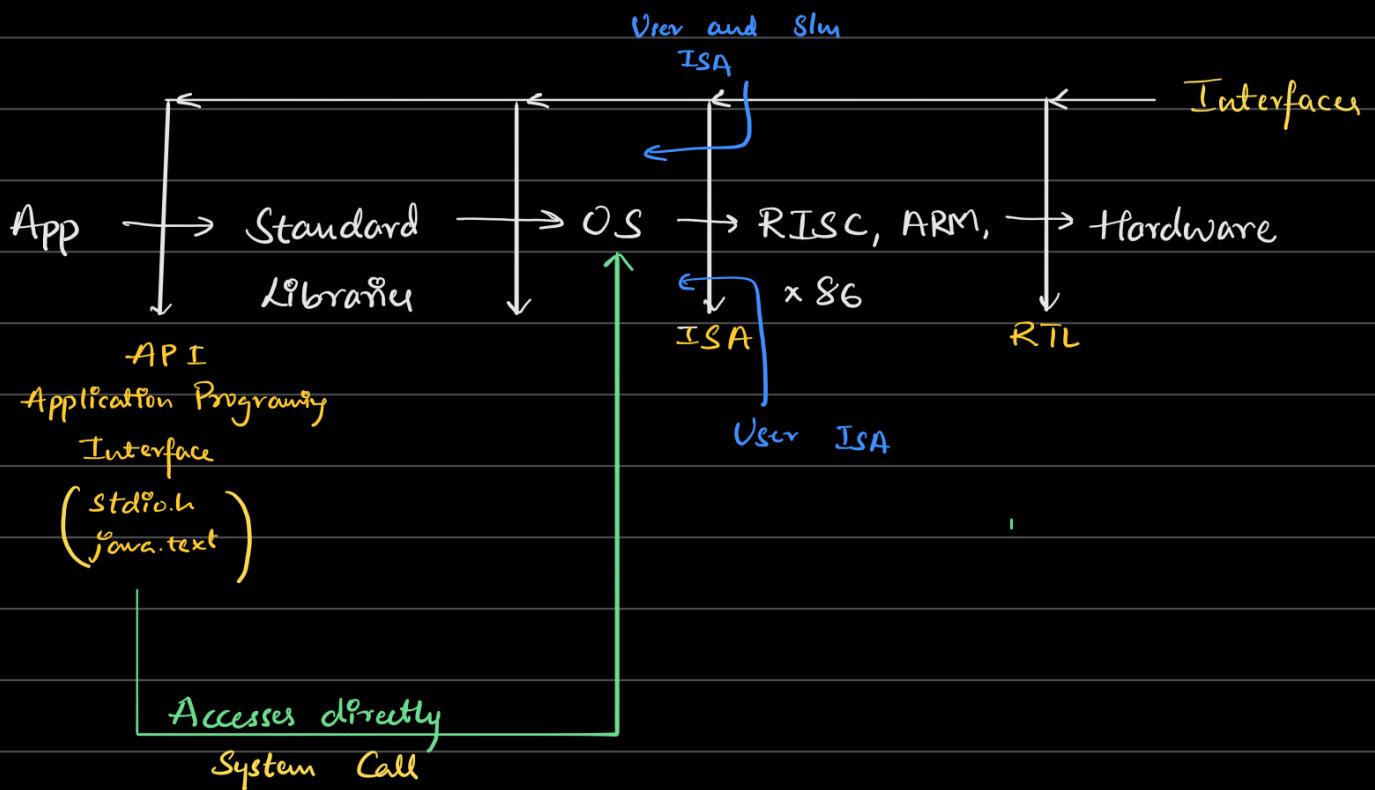


DO2TO1 ABI - Application Binary Interface.

r0, a0, a1, sp, lr - registers in ARM.

What is Interface?

↳ Interface are the means by which we can convert one form of information to another.



- Some parts of the ISA are available to the user directly and some are OS exclusive, these are called User ISA and System ISA.
- Application uses System Call via ABI to access OS directly i.e., access the registers directly and convert them.

ABI - Application Binary Interface.

or System Call Interface.

Register Allocation for RISC-V CPUs

Width of registers is denoted by **XLEN**

It is 32 bit for RV32

64 bit for RV64

There's two ways to load the data.

i. Directly load info. to registers.

ii. Load onto the memory bytes and then to the register.

There are again multiple ways to load this in memory.

One memory address make up 1 byte of information.

i.e. 8 bits.

$m_0 \rightarrow 8 \text{ bits}$

$m_1 \rightarrow 8 \text{ bits}$

MSB ↑ Enters into memory by the order of LSB to MSB

$m_8 \rightarrow 8 \text{ bits}$, LSB in groups of 8 bits.

$m_n \rightarrow 8 \text{ bits}$

This type of memory system is called Little Endian. RISC-V belongs to the little-endian memory addressing system.

In Big Endian, the MSB to LSB fills from m_8 to m_0 . i.e., vice versa of the current one.

One doubleword requires 8 bits of data.

→ Consider an array M of 3 doublewords, stored in memory addresses $m[0]$ to $m[2]$.
Our purpose is to load one of these doublewords into the register.

To access the dw starting from 16th bit ($m[16]$) we need the base address $m[0]$ of the memory to be stored in another register.

Instruction size for the RV ISA are 32 bits only, regardless of whether it is RV32 or RV64.

D02T03

The instructions which operate on signed or unsigned integers are called as 'Base Integer Instructions' (RV64I) (BII)

Any CPU core which implements RV64I has to have the 47 instructions.

→ Instructions which only deal with registers are called as R-Type BI.

→ Instructions which deal with registers and an immediate constant are called as I-Type BI.

→ Instructions which deal with only source registers and an immediate are called as S-Type BI.

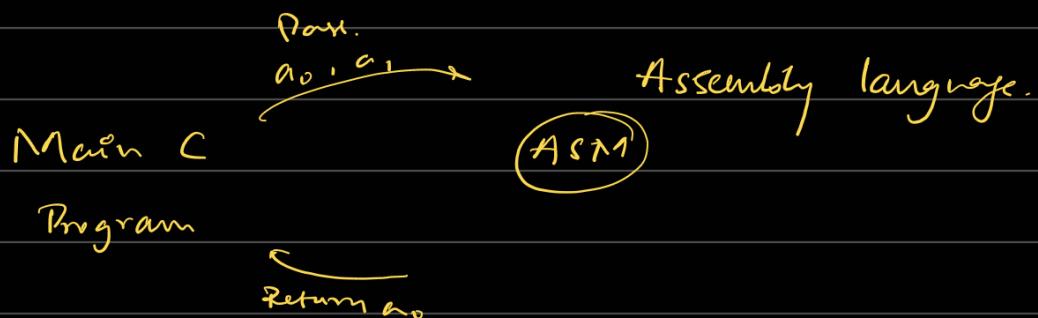
Conclusion:-

As we can observe, registers are allocated 5 bits.

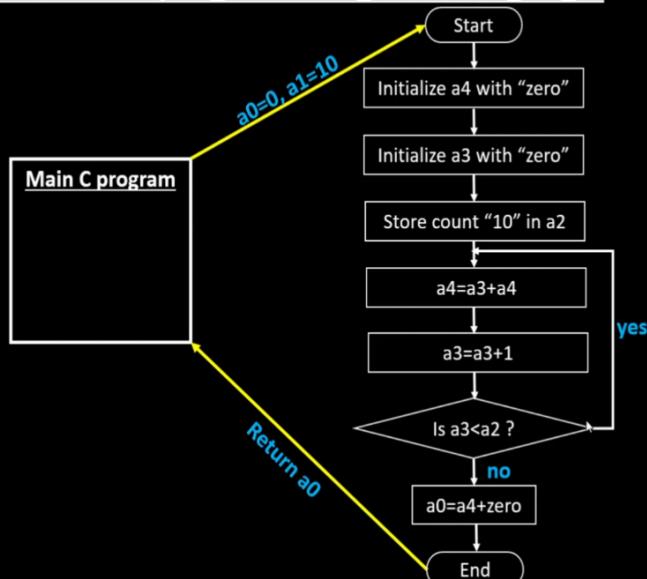
That is why we have 32 registers = 2^5 in RV

Register	ABI name	Usage	Saver
x0	zero	Hard-wired zero	-
x1	ra	Return address	Caller
X2	sp	Stack pointer	Callee
x3	gp	Global pointer	-
x4	tp	Thread pointer	-
x5-x7	t0-2	Temporaries	Caller
x8	s0/fp	Saved register/frame pointer	Callee
x9	s1	Saved register	Callee
x10-11	a0-1	Function arguments/return values	Caller
x12-17	a2-7	Function arguments	Caller
x18-27	s2-11	Saved registers	Callee
x28-31	t3-6	Temporaries	Callee

D02 2021 (22)



Lab to re-write C program using ASM language



Q) why rewrite $a_0 = a_2 + 0$ to return a_0 ? why not directly return a_2 ?