12. What is interrupt cycle? Elaborate this concept with the help of RTL language.

Ans:- Interrupt cycle is a process by which a computer retrives a program instruction from its interrupt service routine (ISR) from its memory, which determines what actions the instruction requires, and carries out those actions. The interrupt cycle is initiated after the last execute phase if the interrupt flip flop R is equal to 1. This flip flop is set to 1 when any input interrupt or output interrupt occurs (i.e. when any input/output devices need to feed data or display the data). The interrupt happens only after T_0, T_1 and T_2 cycles of any instruction program (i.e the interrupt is handled only after decode cycle has been completed). The condition for setting flip flop R is given as:

$$T_0' T_1' T_2' \cdot IEN \cdot (FGI + FGO) : R \leftarrow 1$$

When the R flip flop is set to 1, the interrupt cycle is starte Whenever R flip flop is 1 the control will allways go through this cycle until it turns to 0. We will denote the phase signals of interrupt cycles as RT_0, RT_1 and RT_2. At first the interrupt cycles stores the return address which i the program counter into memory location 0. It is done with the help of a temporary register and then the program counter is set to 0. It is then incremented to 1 in nex phase and the IEN flag is cleared along with the sequence counter and then the memory location 1 contains

branch instruction which sends the control to the interrupt service routine where our interrup is handled and finaly the control goes back to memory location '0' where our return address is stored. It is shown in RTL as follows.

RT_0 : $AR \leftarrow 0$, $TR \leftarrow PC$.

RT_1 : $M[AR] \leftarrow TR$, $PC \leftarrow 0$.

RT_2 : $PC \leftarrow PC+1$, $IEN \leftarrow 0$, $R \leftarrow 0$, $SC \leftarrow 0$.