Main Function:

main void begin (){

}

All DataTypes:

    main void begin ()

{

NUM ~int1 = 10 ;

ftNUM ~float = 5.1 ;

senBIT ~char = 'A' ;

Sentence ~String = "Ali Gauhar" ;

}

Functions, Single line comment and input statement

-> function signature

void ~addFunction (NUM ~num1, NUM ~num2) ;

->  main function

main void begin ()

{

~addFunction ( 5, 10 ) ;

}

-> function definition

func void ~addFunction (NUM ~num1, NUM ~num2) {

NUM ~X ;

in\_put <- ~C ;

NUM ~Y ;

in\_put <- ~D ;

NUM ~Z = ( ~G + ~H );

out\_put <- "Addition Done" ;

-> since the output can only be string therefore type casting

-> needed but here the typ casting functionality has not been

-> embedded therefore only printing the signal

}

While loop:

main void begin ()

{

WhileLP ( ~A & ~B ){

    out\_put <- "Hello";

    out\_put <- "Good Morning";

}

}

For Loop:

main void begin ()

{

ForLP ( NUM ~int = 5 ; ~int <= ~X ; ~int ++ ){

    out\_put <- "Hello";

    out\_put <- "Good Morning";

}

}

If else statement:

main void begin ()

{

TR ( ~A ){

    out\_put <- "Hello";

    out\_put <- "Good Morning";

}

FL{

    out\_put <- "No Hello";

    out\_put <- "No Good Morning";

}

}

Switch Cases:

main void begin ()

{

choose ( ~X ) {

From ~A :

      in\_put <- ~ABC;

      Stop;

From ~B :

      in\_put <- ~DEF;

      Stop;

default :

       in\_put <- ~XYZ;

       Stop;

}

}

Expression:

main void begin ()

{

NUM ~x = (~A + ~B) \* ~C / ~D;

}