Codul VHDL furnizat implementează un multiplicator de tip Robertson, folosind mai multe subcomponente pentru a efectua operații de înmulțire pe numere în sistem binar.

Iată o descriere detaliată a câtorva dintre componentele principale din acest cod:

**ControlUnit\_Robertson**: Această componentă este o unitate de control care coordonează operațiile și secvențele din circuit. Ea gestionează semnalele de control pentru celelalte module și decide modul în care acestea interacționează pentru a efectua operația de înmulțire, sub forma unui automat cu stari finite.

Comportament specific stărilor:

RST: Inițializează anumite semnale și stabilește starea inițială (INIT).

INIT: Resetează unele semnale și trece la starea SC.

SC: Controlează semnalele în funcție de condițiile implicate de q0, count și done.

LOAD0 și LOADM: Setează anumite semnale și trece la starea ADD.

ADD: Modifică semnalele și trece la RSHIFT.

RSHIFT: Gestionează operațiile de deplasare în funcție de condițiile implicate de count și done.

VERIFY: Bazându-se pe condițiile implicate de q0, done și in\_reset, trece la diferite stări.

SUBTK: Setează anumite semnale și trece la RSHIFT.

OUTPUT: Gestionează stările semnalelor, decide următoarea stare în funcție de in\_reset.

**vect\_mux**: Acesta este un multiplexor vectorial. El are rolul de a selecta și de a direcționa diferitele căi de date în funcție de semnalele de control furnizate de unitatea de control.

**ShiftRegiste**r: Reprezintă un registru de deplasare. Este utilizat pentru a deplasa și a rearanja biții de date într-o anumită direcție în funcție de semnalele de control.

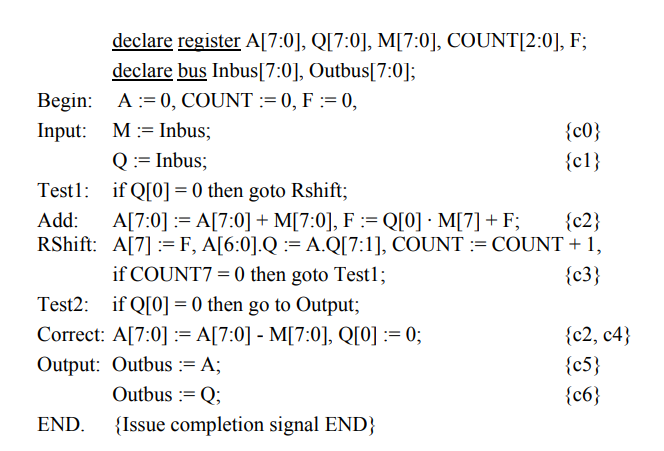
**N\_Counter**: Este un contor utilizat pentru a efectua operații de numărare în cadrul circuitului. Acesta poate gestiona secvențele și intervalele de numărare necesare pentru operația de înmulțire.

**CLA\_Parallel\_Adder**: Este un adăugător de tip carry-lookahead, folosit pentru a aduna numerele în timpul operației de înmulțire. Este responsabil pentru efectuarea adunărilor de cifre binare și pentru gestionarea depășirilor de cifre (carry-urilor) în operația de adunare.

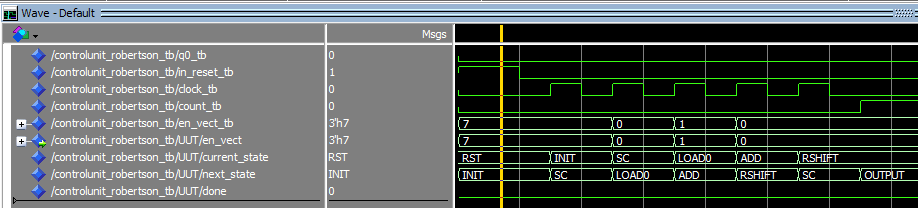
**AQ\_Block**: Această componentă este un bloc care gestionează registrele 'A' și 'Q'. Ele sunt esențiale pentru reținerea și manipularea datelor necesare în timpul procesului de înmulțire.

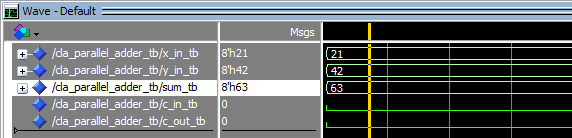
Fiecare dintre aceste componente contribuie la operația generală a multiplicatorului Robertson, interacționând și comunicând între ele conform semnalelor de control transmise de unitatea de control. Operația de înmulțire este realizată prin deplasarea și adunarea repetată a datelor în funcție de logicile implementate în aceste componente.

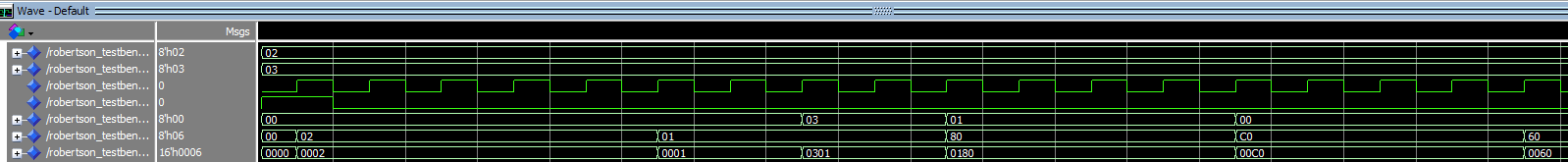
Pseudocodul din care ne-am inspirat pentru implementarea inmultitorului:



Testarea functionalitatii:

Test control unit:

Test parallel adder:

Test inmultitor Roberson complet:

2x3 face 6 .