Partitioning

Chapter 6

Source: Gajski, Dutt, Wu, Lin

"High−Level Synthesis"

Kluwer Academic Publishers, 1992

PARTITIONING

Used in HLS for:

Scheduling

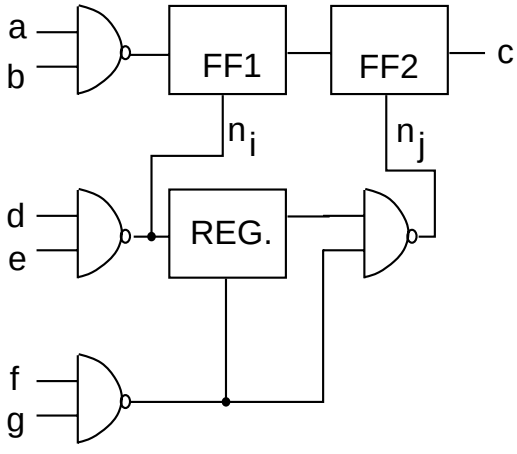
Allocation

Unit selection

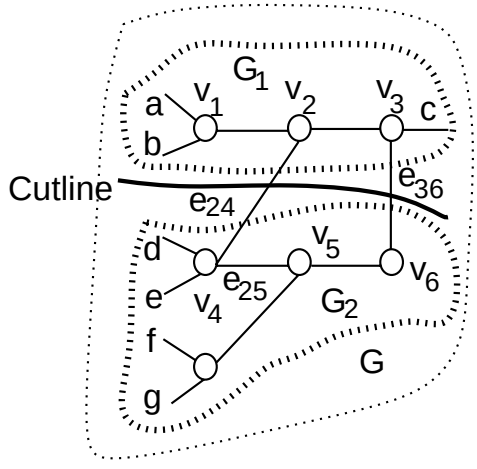
Chip partitioning

Problem decomposition
for tractability

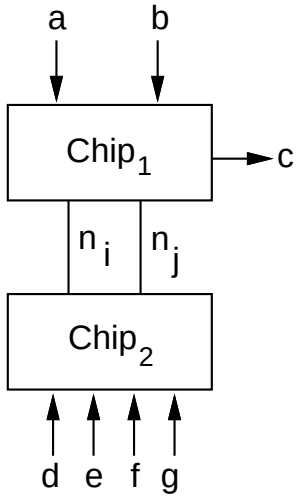
COMPONENT PARTITIONING



Design structure



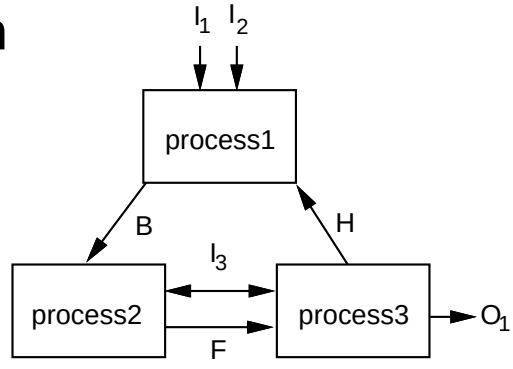
Graph model



Partitioned Design

BEHAVIORAL PARTITIONING

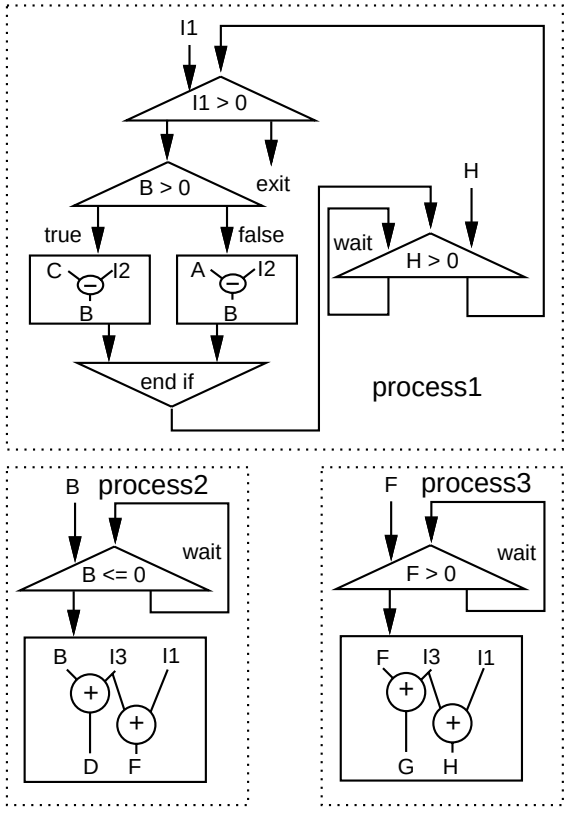
−Time utilization
−Component utilization



Inter−process communication

```
entity VHDL EXAMPLE is
        port (I1, I2, I3 : in integer;
                O1 : out integer;)
        signal B, F, H : integer;
end entity;

architecture BEHAVIOR of EXAMPLE is
begin
    process1
        var : A, C, E : integer;

        while (I1 > 0)  loop
                if (B > 0)
                    B <= C − I2;
                else
                    B <= A − I2;
                end if;
                wait until (H > 0);
        end loop;
    end process1;

    process2
        var : D : integer;
        wait until (B <= 0);
        D := I3 + B;
        F <= I3 + I1;
    end process2;

    process3
        var : G : integer;
        wait until (F > 0);
        O1 <= I3 + G;
        H <= I3 + I1;
    end process3;

end
```

Textual Description



Control/Data Flow Graph

PARTITIONING TECHNIQUES

Constructive methods

1. Random selection
2. Cluster growth
3. Hierarchical clustering

Iterative−Improvement methods

1. Min−cut partitioning
2. Simulated annealing

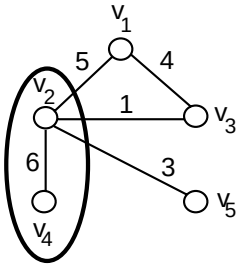
CLUSTER GROWTH ALGORITHM

Algorithm 6.1

page 185

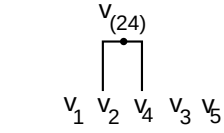
HIERARCHICAL CLUSTERING

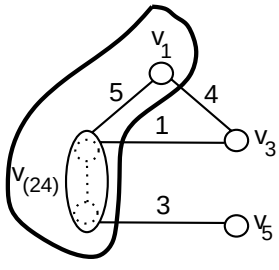
Graph | Closeness measure | Cluster tree



	v_1	v_2	v_3	v_4	v_5
v_1	–	–	–	–	–
v_2	5	–	–	–	–
v_3	4	1	–	–	–
v_4	0	**6**	0	–	–
v_5	0	3	0	0	–

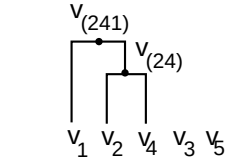
(a)

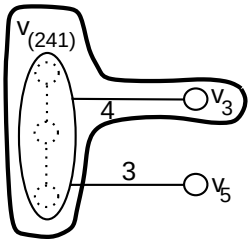




	v_1	$v_{(24)}$	v_3	v_5
v_1	–	–	–	–
$v_{(24)}$	**5**	–	–	–
v_3	4	1	–	–
v_5	0	3	0	–

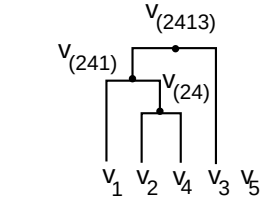
(b)

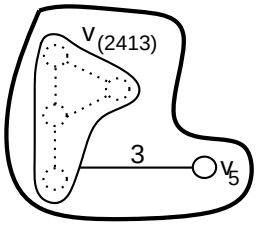




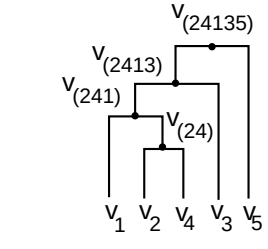
	$v_{(241)}$	v_3	v_5
$v_{(241)}$	–	–	–
v_3	**4**	–	–
v_5	3	0	–

(c)





	$v_{(2413)}$	v_5
$v_{(2413)}$	–	–
v_5	**3**	–



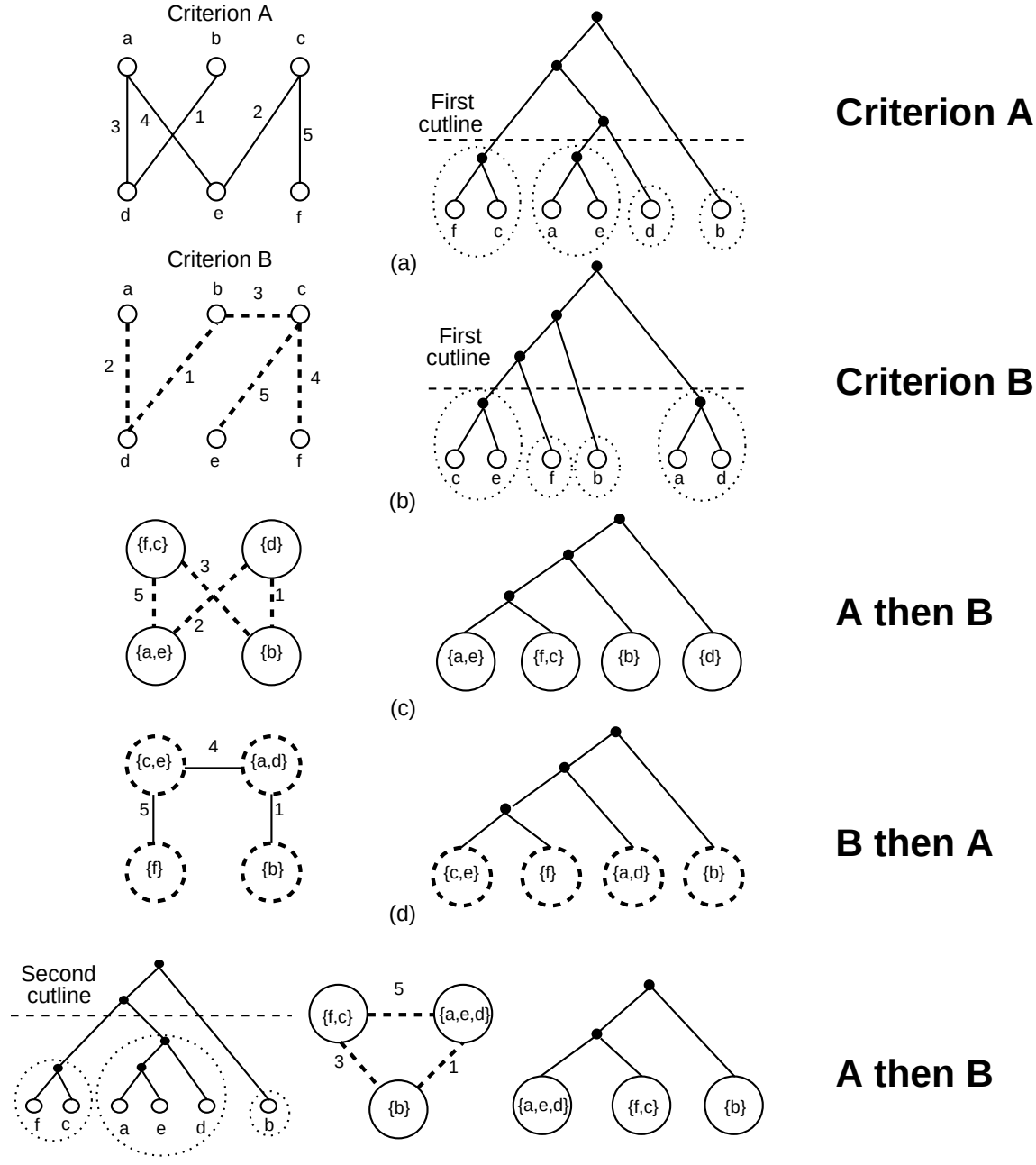
Cluster tree formation

HIERARCHICAL CLUSTERING

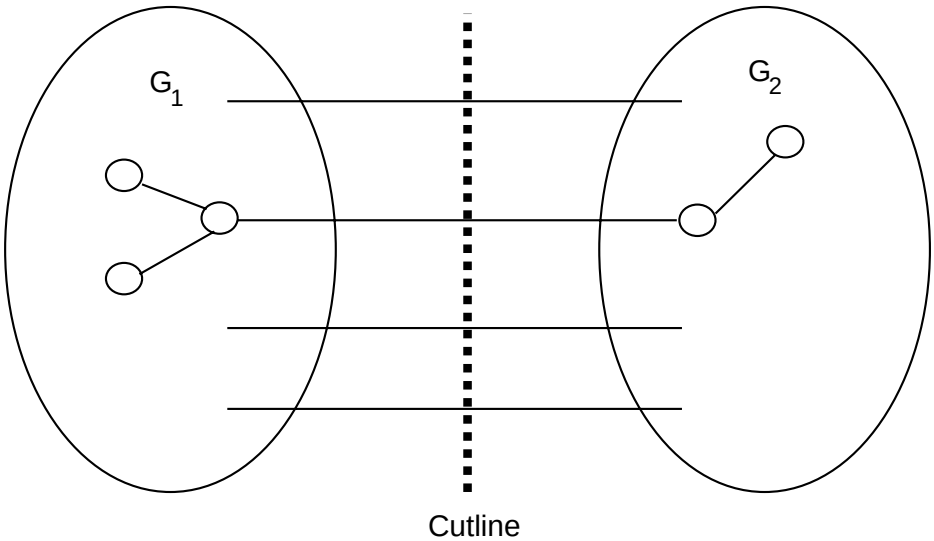
Algorithm 6.2

page 188

CLUSTERING WITH SEVERAL CRITERIA



ITERATIVE ALGORITHMS



Two−way partitioning (Kernighan−Lin)

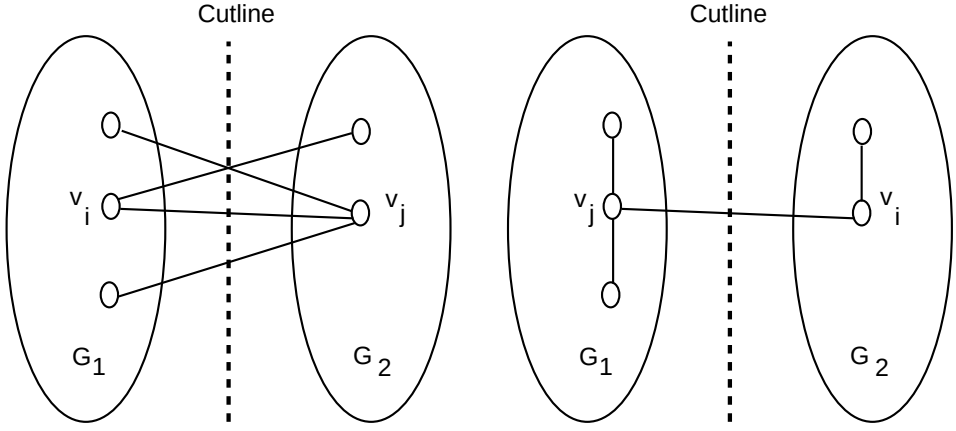
Start with 2 equal subgraphs

Each iteration, exchange
K−pairs between partitions

Continue until no further
improvement

MIN−CUT PARTITIONING

Interconnection Reduction



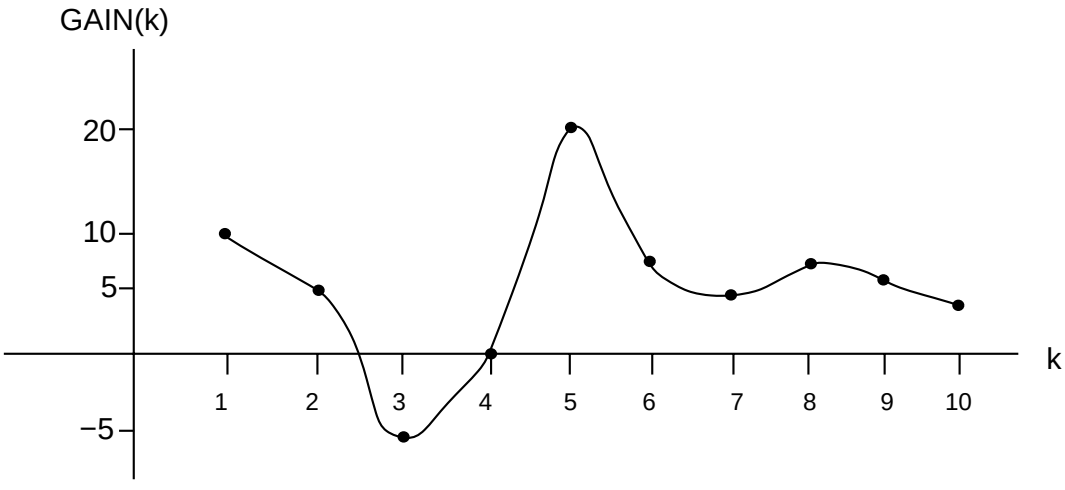
Before interchange of Vi and Vj

After interchange of Vi and Vj

MIN−CUT PARTITIONING

Algorithm 6.3
page 194

MIN−CUT SEARCH STRATEGY



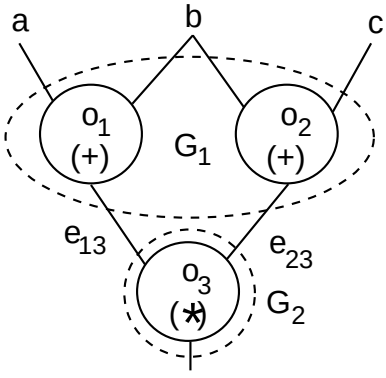
GAIN (5) is maximum.

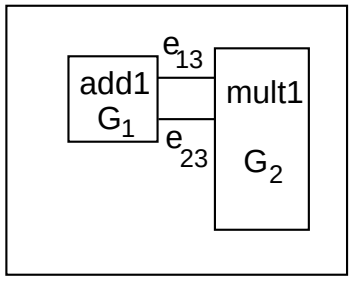
Thus, perform the first 5 exchanges

SIMULATED ANNEALING

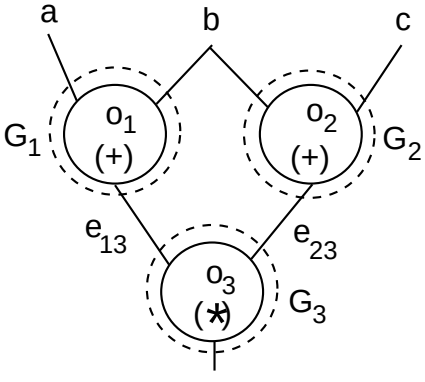
Algorithm 6.4
page 197

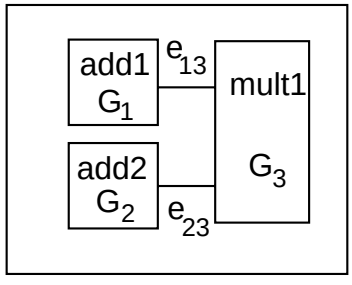
CLUSTERING EXAMPLE





Two cluster partition





Three cluster partition

CLUSTERING FOR UNIT SELECTION

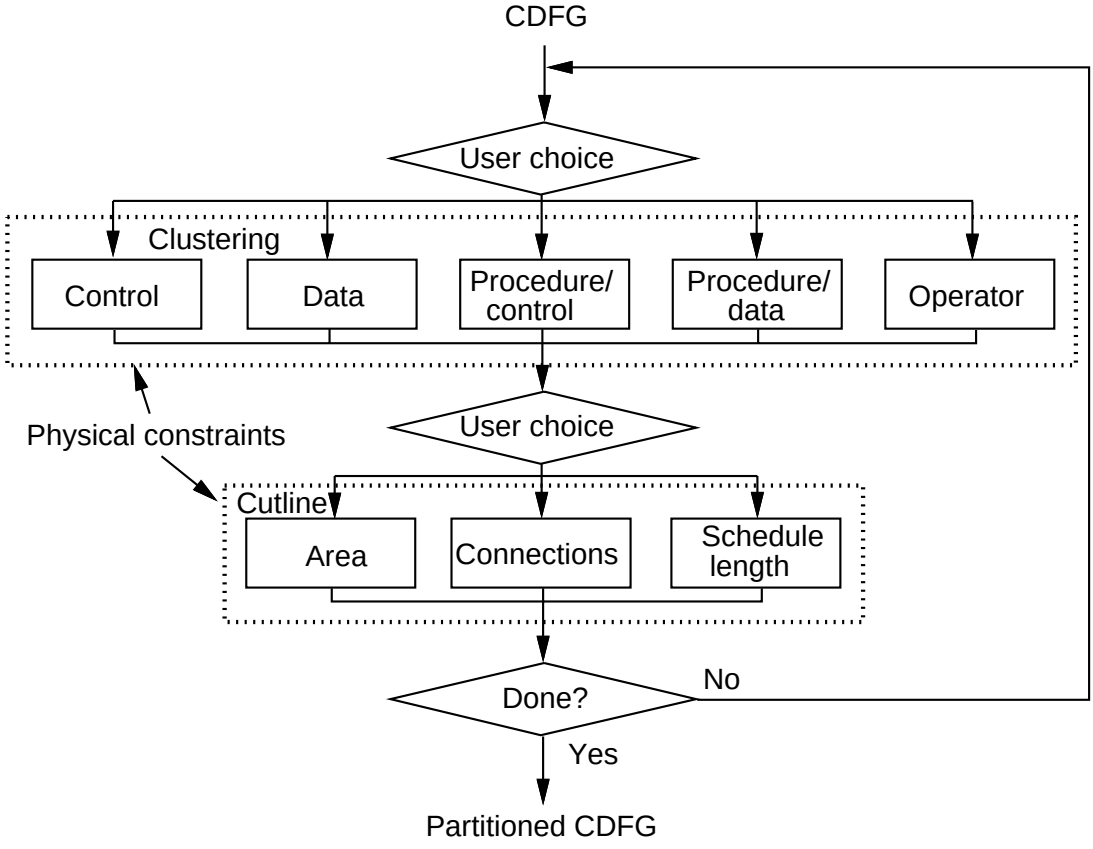
1. Functional proximity

2. Communication proximity

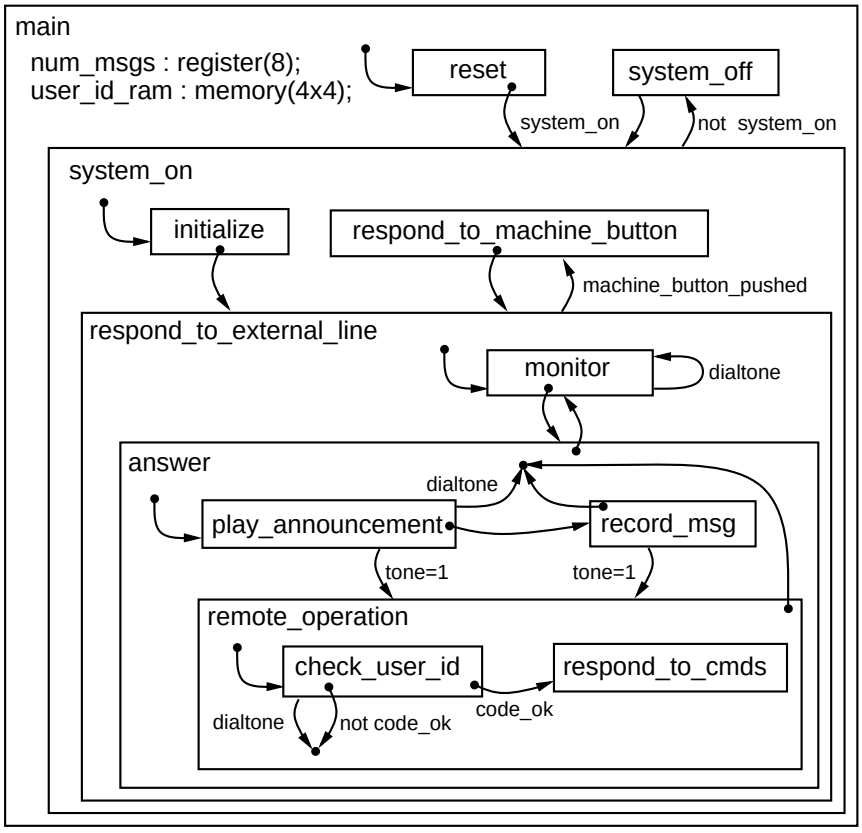
3. Potential parallelism

4. Closeness distance

PARTITIONING SCRIPTS IN APARTY



DESCRIPTION PARTITIONING

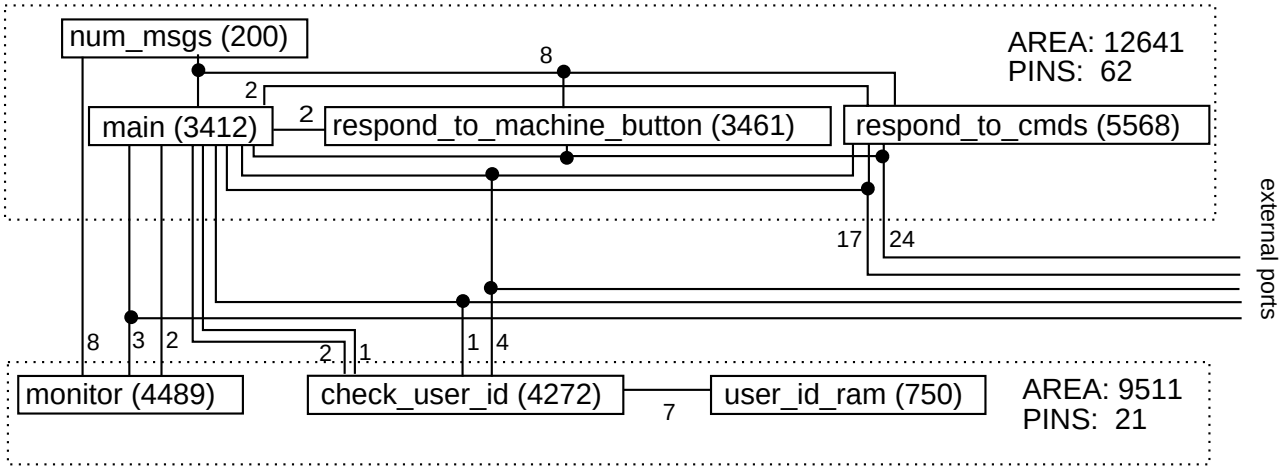


Answering machine description

```
check_user_id                    entered_code : memory(4x4);
                                 i : integer range 1 to 5;
   i := 1;
   while i <= 4 loop
      wait until button_tone;
      entered_code[i] := button;
      i := i + 1;
   end loop;

   if (entered_code[1] = user_id_ram[1]) and
       ...
      (entered_code[4] = user_id_ram[4]) then
      code_ok <= true;
   else
      code_ok <= false;
   end if;
```

Object description

PARTITIONED DESCRIPTION



FUTURE DIRECTIONS

Partitioning of CDFG

Partitioning of specifications

Estimation of quality measures

Software/Hardware partitioning