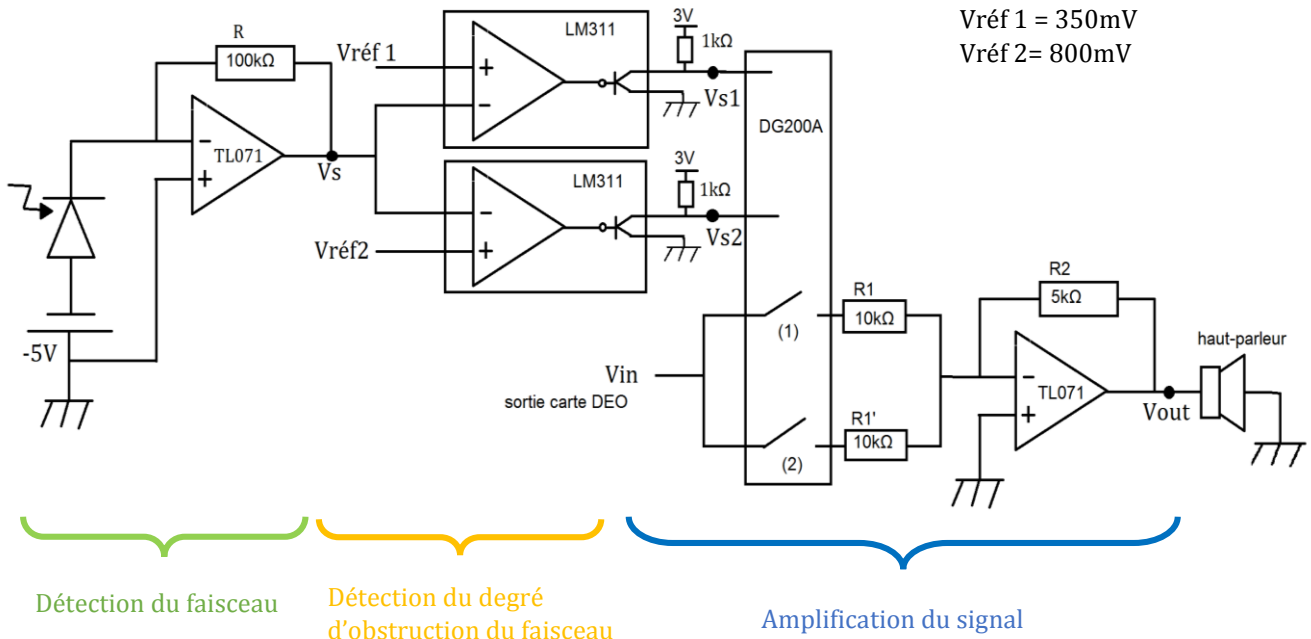
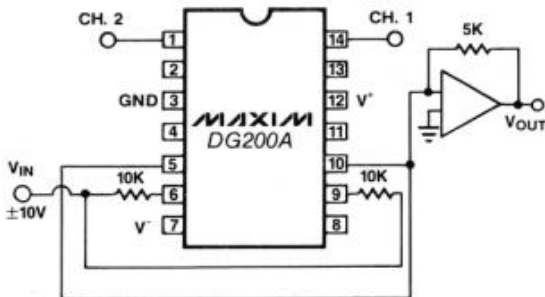


# Harpe laser : dossier technique

## Circuit électronique global pour un faisceau :



## Circuit électronique de la partie amplification :



## Pour plusieurs faisceaux :

On utilise des composants tels que l'AO **TL084** ou le composant **LM339**, qui regroupent plusieurs composants en un. En effet, le TL084 est équivalent à quatre composants TL071. Or on a besoin de deux TL071 par faisceaux il est utilisable pour 2 faisceaux. On utilise également le **LM339** qui remplace deux composants **LM311**.

**LM339, LM339A... D, DB, N, NS, OR PW PACKAGE**  
**LM2901... D, N, NS, OR PW PACKAGE**  
(TOP VIEW)

1OUT	1	14	OUT3
2OUT	2	13	OUT4
V <sub>CC</sub>	3	12	GND
2IN-	4	11	4IN+
2IN+	5	10	4IN-
1IN-	6	9	3IN+
1IN+	7	8	3IN-

**LM311... D, P, PS, OR PW PACKAGE**  
(TOP VIEW)

EMIT OUT	1	8	V <sub>CC</sub> +
IN+	2	7	COL OUT
IN-	3	6	BAL/STRB
V <sub>CC</sub> -	4	5	BALANCE

**TL084, TL084A, TL084B**  
**D, J, N, NS, OR PW PACKAGE**  
(TOP VIEW)

1OUT	1	14	4OUT
1IN-	2	13	4IN-
1IN+	3	12	4IN+
V <sub>CC</sub> +	4	11	V <sub>CC</sub> -
2IN+	5	10	3IN+
2IN-	6	9	3IN-
2OUT	7	8	3OUT

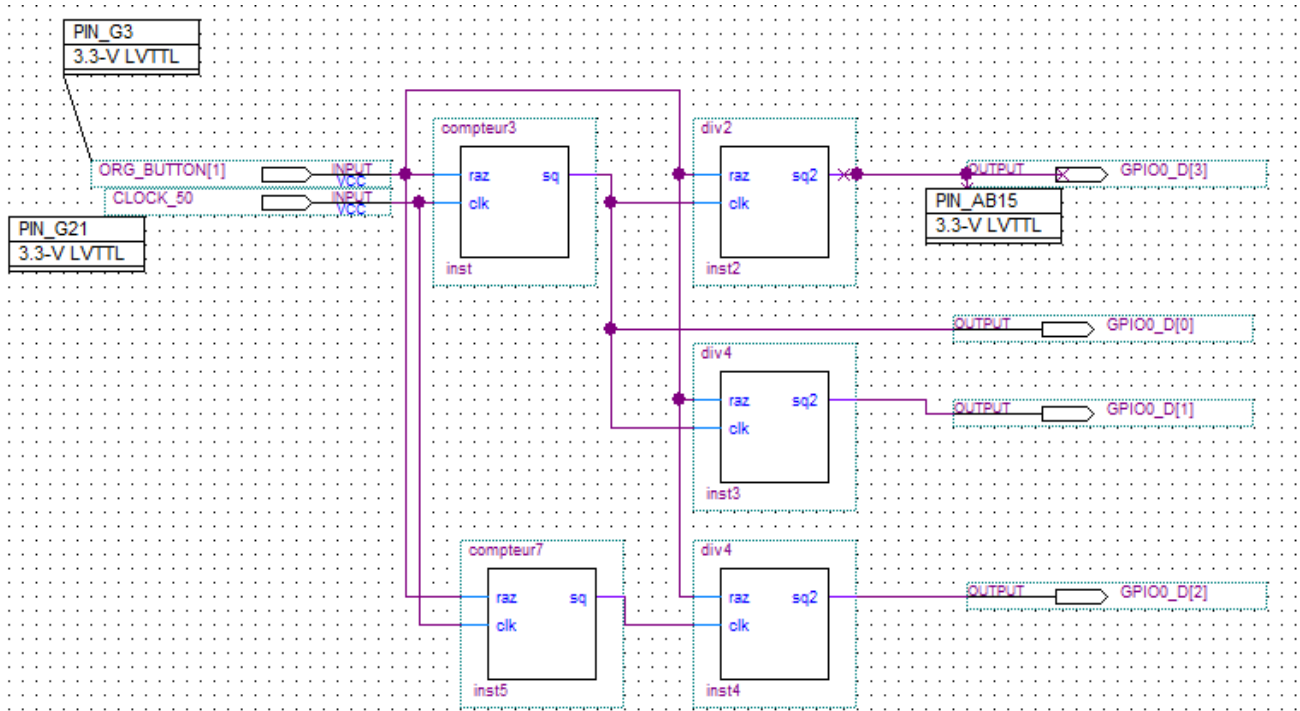
**PIN CONNECTIONS**

TL071 (Top View)

Source : datasheetcatalog.com

Partie relative à la carte DEO :

Block diagramme



sortie	fréquence	Note/son	sortie	fréquence	Note/son
D0	1525Hz	Fa / sol octave (6)	D2	1017Hz	si octave (4)
D1	254Hz	Do octave (2)	D3	150Hz	Ré octave (2)

Programmes VHDL utilisés sur la carte DEO

Division de la fréquence du signal d'entrée par  $2^{15}$  :

```

library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.STD_LOGIC_UNSIGNED.ALL;
use IEEE.STD_LOGIC_ARITH.ALL;

entity compteur3 is
port(
  raz, clk : in std_logic;
  sq : out std_logic
);
end compteur3;

architecture Behavioral of compteur3 is
  signal total : STD_LOGIC_VECTOR(15 DOWNTO 0);
begin
  cpt : process(raz,clk)
  begin
    if raz = '0' then
      total<="0000000000000000";
    elsif(clk'event and clk='1')then

      total <= total+1;
    end if ;
  end process ;

  sq <= total(14);
end Behavioral;

```

Division la fréquence du signal d'entrée par  $2^{13}$  :

```

1  library IEEE;
2  use IEEE.STD_LOGIC_1164.ALL;
3  use IEEE.STD_LOGIC_UNSIGNED.ALL;
4  use IEEE.STD_LOGIC_ARITH.ALL;
5
6  entity compteur7 is
7  port(
8  raz, clk : in std_logic;
9  sq : out std_logic
10 );
11 end compteur7;
12
13 architecture Behavioral of compteur7 is
14 signal total : STD_LOGIC_VECTOR(15 DOWNTO 0);
15 begin
16 cpt : process(raz,clk)
17 begin
18   if raz = '0' then
19     total<="0000000000000000";
20   elsif(clk'event and clk='1')then
21
22     total <= total+1;
23   end if ;
24 end process ;
25
26 sq <= total(12);
27 end Behavioral;

```

Division la fréquence du signal d'entrée par 6 :

```

entity div4 is
port(
  raz, clk : in std_logic;
  sq2 : out std_logic
);
end div4;

architecture Behavioral of div4 is
  signal total :STD_LOGIC_VECTOR(3 DOWNTO 0) ;
begin
  cpt : process(raz,clk)
  begin
    if raz = '0' then
      total<="0000";

    elsif(clk'event and clk='1')then
      total <= total+1;

      if(total="0101") then
        total<="0000";
      end if;

    end if ;

  end process ;

  sq2<='0'when total<"0011" else
    '1';
end Behavioral;

```

Division la fréquence du signal d'entrée par 10 :

```

entity div2 is
port(
  raz, clk : in std_logic;
  sq2 : out std_logic
);
end div2;

architecture Behavioral of div2 is
  signal total :STD_LOGIC_VECTOR(3 DOWNTO 0)
begin
  cpt : process(raz,clk)
  begin
    if raz = '0' then
      total<="0000";

    elsif(clk'event and clk='1')then
      total <= total+1;

      if(total="1001") then
        total<="0000";
      end if;

    end if ;

  end process ;

  sq2<='0'when total<"0101" else
    '1';
end Behavioral;

```