

```
ENTITY midesp IS
  GENERIC(numbits: positive := 8);
  PORT(ei, ed, cero, reloj : IN bit;
        sel: IN bit_vector(1 DOWNTO 0);
        par: IN bit_vector(numbits-1 DOWNTO 0);
        sal: OUT bit_vector(numbits-1 DOWNTO 0));
end midesp;

---

ARCHITECTURE compo OF midesp IS
  SIGNAL estado: bit_vector(numbits-1 DOWNTO 0);
BEGIN

  PROCESS(cero, reloj)
    VARIABLE siguiente: bit_vector(numbits-1 DOWNTO 0);
  BEGIN
    CASE sel IS
      WHEN "00" => siguiente:= estado;
      WHEN "01" => siguiente:= ei & estado(numbits-1 DOWNTO 1);
      WHEN "10" => siguiente:= estado(numbits-2 DOWNTO 0) & ed;
      WHEN OTHERS => siguiente:= par;
    END CASE;
    IF cero = '0' THEN
      estado <= (OTHERS => '0');
    ELSIF (reloj'event AND reloj = '1') THEN
      estado <= siguiente;
    END IF;
  END PROCESS;

  sal <= estado;

END compo;

---

ARCHITECTURE rtl OF midesp IS
  SIGNAL estado, siguiente: bit_vector(numbits-1 DOWNTO 0);
BEGIN

  sal <= estado;
  WITH sel SELECT
    siguiente <= estado WHEN "00",
      ei & estado(numbits-1 DOWNTO 1) WHEN "01",
      estado(numbits-2 DOWNTO 0) & ed WHEN "10",
      par WHEN OTHERS;
  estado <= (OTHERS => '0') WHEN cero = '0' ELSE
    siguiente WHEN (reloj'event AND reloj = '1') ELSE
    estado;

END rtl;
```