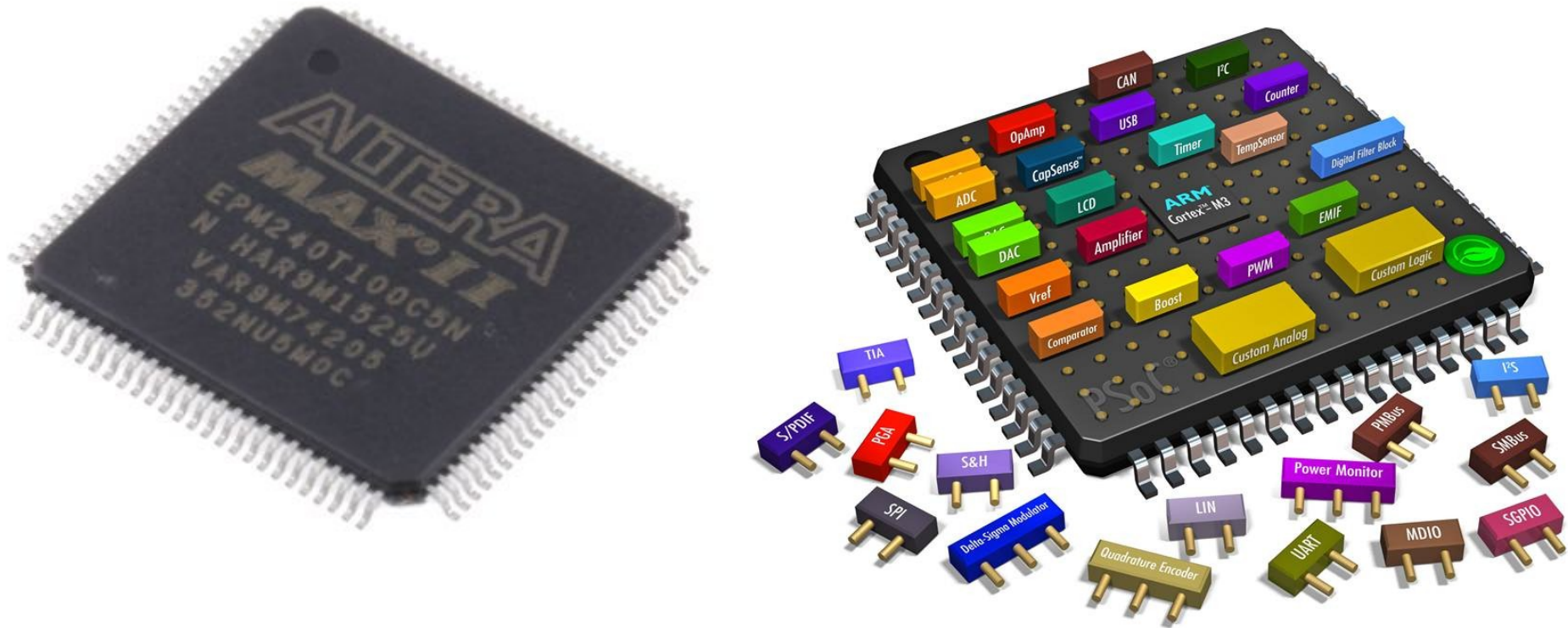


# Újrakonfigurálható eszközök



## 4. Verilog példaprogramok EPM240-hez

# Tartalom

---

- C-M240 fejlesztői kártya, felhasznált kivezetések
- 15-fdiv-LED: LED villogtatás frekvencia leosztással
- 16-fdiv-div: Két LED villogtatása frekvencia-osztással
- Project05: LED futófény
- Project06: Gombok és számkijelzés
- Project07: Bináris számláló
- Project08: 8-bites léptetőregiszter

# C-M240 fejlesztői kártya

Gyártó: Shenzhen 21EDA Electronic Technology    CPLD: Altera MAX II EPM240T100C5N

Órajel: 50 MHz    VDD: 3,3 V    Perifériák: 8-digit kijelző, 8 LED, 4 +1 nyomógomb, 1 csipogó

Fejlesztői környezet: **ALTERA QUARTUS Prime (Intel FPGA)**



# C-M240 – felhasznált kivezetések

## Bemenetek

**CLK\_50M** az órajel bemenet  
**DEV\_CLK** a reset gomb

CLK_50M	input	PIN_12
K1 gomb	input	PIN_29
K2 gomb	input	PIN_28
K3 gomb	input	PIN_27
K4 gomb	input	PIN_26
DEV_CLK	input	PIN_44

## Kimenetek

BELL	output	PIN_50
D30 LED8	output	PIN_51
D31 LED7	output	PIN_52
D32 LED6	output	PIN_53
D33 LED5	output	PIN_54
D34 LED4	output	PIN_55
D35 LED3	output	PIN_56
D36 LED2	output	PIN_57
D37 LED1	output	PIN_58

# C-M240 – felhasznált kivezetések

## Számjegyek

DIGIT1 (jobb szélső)	output	PIN_86
DIGIT2	output	PIN_87
DIGIT3	output	PIN_96
DIGIT4	output	PIN_89
DIGIT5	output	PIN_97
DIGIT6	output	PIN_91
DIGIT7	output	PIN_92
DIGIT8 (bal szélső)	output	PIN_95

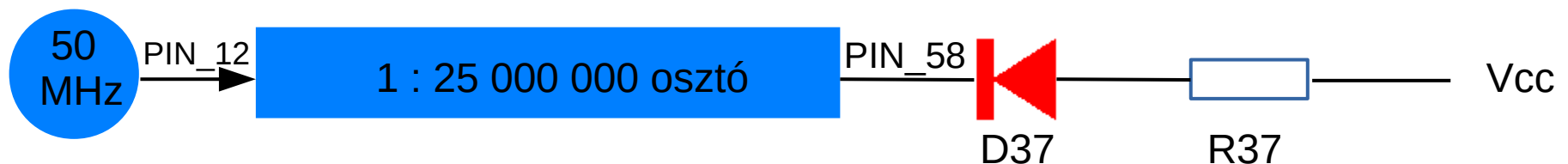
## Szegmensek

DATA0	output	PIN_85	A
DATA1	output	PIN_84	B
DATA2	output	PIN_83	C
DATA3	output	PIN_82	D
DATA4	output	PIN_81	E
DATA5	output	PIN_78	F
DATA6	output	PIN_77	G
DATA7	output	PIN_76	DP

# LED villogtatás frekvencia leosztással

## Projekt: 15-fdiv-LED

```
// Egy frekvenciaosztót használunk a D37 LED (pin 58) villogtatásához
module ledwater (clk_50M, led_out);
  input clk_50M;           // 50MHz-es órajel (pin 12 bemenet)
  output led_out;         // LED (pin 58 kimenet)
  reg [24: 0] count;      // 25,000,000-szoros osztó
  reg div_clk;           // a leosztott órajel
  reg led_out;           // LED kimenet
  always @ (posedge clk_50M)
  begin
    if (count == 25000000)
      begin
        div_clk <= ~div_clk; // 25,000,000 ciklust számlál
        count <= 0;         // 0.5 s periodusidő (2 Hz).
                             // számláló törlése
      end
    else
      count <= count + 1;   // számláló léptetése
      led_out <= div_clk;   // másodpercenként felvillan a LED
    end
  end
endmodule
```



# Két LED villogtatása frekvencia-osztással

- Az előző projektet kibővítjük egy második számlálóval, melynek kimenete 4 Hz-en ketyeg. Ezzel egy másik LED-et villogtatunk.

## Projekt: 16-fdiv-div

```
module ledwater (clk_50M, led_out, f_led_out);  
input clk_50M; // 50MHz-es órajel (pin 12 bemenet)  
output led_out; // LED2 (pin 57 kimenet)  
output f_led_out; // LED1 (pin 58 kimenet)  
reg [24: 0] count; // 25,000,000-szoros osztó  
reg [24: 0] f_count; // 12,500,000-szoros osztó  
reg div_clk, f_div_clk; // A leosztott órajelek  
reg led_out, f_led_out; // LED kimenetek
```

```
// A count frekvenciaosztót használjuk a D36 LED (pin 57) villogtatásához  
always @ (posedge clk_50M)  
begin  
    if (count == 25000000)  
        begin // 25,000,000 ciklust számlál  
            div_clk <= ~div_clk; // 0.5 s periodusidő (2 Hz).  
            count <= 0; // számláló törlése  
        end  
    else  
        count <= count + 1; // számláló léptetése  
        led_out <= div_clk; // másodpercenként felvillan LED2  
    end
```

# Két LED villogtatása frekvencia-osztással

- Az előző projektet kibővítjük egy második számlálóval, melynek kimenete 4 Hz-en ketyeg. Ezzel egy másik LED-et villogtatunk.

## Projekt: 16-fdiv-div

```
// Az f_count frekvenciaosztót használjuk a D37 LED (pin 58) villogtatásához
always @ ( posedge clk_50M )
begin
    if ( f_count==12500000 ) // 12,500,000 ciklust számlál
        begin
            f_div_clk<=~f_div_clk; // 0.25 s periódusidő (4 Hz).
            f_count<=0;
        end
    else
        f_count<=f_count+1; // számláló léptetése
        f_led_out<=f_div_clk; // félmásodpercenként felvillan LED1
end

endmodule
```



# Project05: LED futófény

- Az alábbi projektben a C-M240 kártya 8 db LED-jén állítunk elő oda-vissza pásztázó futófényt. Ebben is van frekvenciaosztó, csak másképp használjuk.
- Az 50 MHz-es rendszer órajelet egy 26 bites számlálóval leosztjuk, melynek felső 4 bitje vezérli a kijelzés ütemeit (0 – 15. ütem).

```
// LED chaser 8 db LED felhasználásával
// Az effekteket egy frekvenciaosztó segítségével ütemezzük
module chaser (clk_50M, dataout);
    input clk_50M;           // 50MHz rendszeróra bemenet (pin 12)
    output [7: 0] dataout;  // a 8 db LED kimenet (pins 51
    reg [7: 0] dataout;
    reg [25: 0] count;      // Frekvenciaosztó számláló regisztere

    // számláló léptetése
    always @ (posedge clk_50M)
        count <= count + 1;
```

# Project05: LED futófény

```
always @ (posedge clk_50M)
begin
  case (count [25:22]) // count[25:22] a számláló legfelső 4 bitje,
                      // ez az ütemezés kulcsa  $T = 2^{27}/50\,000\,000$  s
  // Egyidejűleg csak egy LED világít
  0: dataout <= 8'b11111110;
  1: dataout <= 8'b11111101;
  2: dataout <= 8'b11111011;
  3: dataout <= 8'b11110111;
  4: dataout <= 8'b11101111;
  5: dataout <= 8'b11011111;
  6: dataout <= 8'b10111111;
  7: dataout <= 8'b01111111;
  8: dataout <= 8'b01111111;
  9: dataout <= 8'b10111111;
  10: dataout <= 8'b11011111;
  11: dataout <= 8'b11101111;
  12: dataout <= 8'b11110111;
  13: dataout <= 8'b11111011;
  14: dataout <= 8'b11111101;
  15: dataout <= 8'b11111110;
  endcase
end
endmodule
```

## Felhasznált kivezetések

clk_50M	Pin 12
LED D37	Pin 58
LED D36	Pin 57
LED D35	Pin 56
LED D34	Pin 55
LED D33	Pin 54
LED D32	Pin 53
LED D31	Pin 52
LED D30	Pin 51

# Project06: gombok és a számkijelzés

- Az alábbi projektben a lenyomott nyomógomb sorszámát íratjuk ki a **C-M240** kártya 7-segmenses számkijelzőjén. A kiírás a nyomógomb sorszámának megfelelő helyiértéken történik (jobb szélről kezdve).
- A nyomógombok pergésmentesítése azáltal megy végbe, hogy csak a tiszta lenyomások jelentenek eseményt, a felengedések és az együttes lenyomások nem.
- **Bemenetek:**

clock_50MHz	Pin 12
K1	Pin 29
K2	Pin 28
K3	Pin 27
K4	Pin 26

## Kimenetek:

Seg A	Pin 85	Digit 1	Pin 86
Seg B	Pin 84	Digit 2	Pin 87
Seg C	Pin 83	Digit 3	Pin 96
Seg D	Pin 82	Digit 4	Pin 89
Seg E	Pin 81	Digit 5	Pin 97
Seg F	Pin 78	Digit 6	Pin 91
Seg G	Pin 77	Digit 7	Pin 92
DP	Pin 76	Digit 8	Pin 95

# Project06: gombok és a számkijelzés

```
module project06(clk_50M, key, segment, digit);
input clk_50M; // 50MHz rendszer órajel (Pin 12)
input [3:0] key; // K1, K2, K3, K4 nyomógombok állapota
output reg [3:0] digit; // számjegyek vezérlése
output reg [7:0] segment; // a DP és a gfedcba szegmensek
reg [3:0] key_temp; // átmeneti tároló

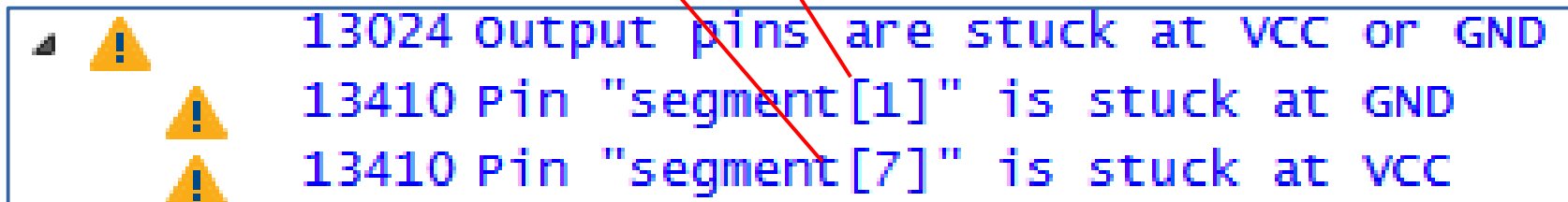
always@ (posedge clk_50M)
begin
    key_temp<=key; // a nyomógombok állapotának rögzítése
    case (key_temp)
        4'b1110: segment<=8'b1111_1001; // szegmens kód K1 lenyomáskor 1
        4'b1101: segment<=8'b1010_0100; // szegmens kód K2 lenyomáskor 2
        4'b1011: segment<=8'b1011_0000; // szegmens kód K3 lenyomáskor 3
        4'b0111: segment<=8'b1001_1001; // szegmens kód K4 lenyomáskor 4
    endcase
end

always@ (posedge clk_50M)
begin
    case(key_temp)
        4'b0111: digit<=4'b0111; // K4 lenyomáskor a 4. számjegy
        4'b1011: digit<=4'b1011; // K3 lenyomáskor a 3. számjegy
        4'b1101: digit<=4'b1101; // K2 lenyomáskor a 2. számjegy
        4'b1110: digit<=4'b1110; // K1 lenyomáskor a 1. számjegy
    endcase
end
endmodule
```

# Néhány szó a figyelmeztetésekről

- Az alábbi figyelmeztetések azt jelzik, hogy bizonyos kimenetek mindig alacsony vagy magas szinten maradnak. Esetünkben ez nem hiba: a B szegmens itt mindig világít, a tizedespont soha.

```
always@ (posedge clk_50M)
begin
  key_temp<=key; // a nyomógombok állapotának rögzítése
  case (key_temp)
    4'b1110:segment<=8'b1111_1001; // szegmens kód K1 lenyomásakor 1
    4'b1101:segment<=8'b1010_0100; // szegmens kód K2 lenyomásakor 2
    4'b1011:segment<=8'b1011_0000; // szegmens kód K3 lenyomásakor 3
    4'b0111:segment<=8'b1001_1001; // szegmens kód K4 lenyomásakor 4
  endcase
end
```



```
▲ ! 13024 Output pins are stuck at VCC or GND
! 13410 Pin "segment[1]" is stuck at GND
! 13410 Pin "segment[7]" is stuck at VCC
```

# Project07: bináris számláló

- Az alábbi projektben egy 21-bites frekvencia előosztót és egy 8-bites bináris számlálót definiáltunk.
- A 8-bites számláló állapotát a C-M240 kártyán található 8 db LED segítségével folyamatosan kijelezzük.

```
module binary_counter (clk_50M, led_array);  
input clk_50M; // 50MHz órajel (Pin 12 bemeneten)  
output [7:0] led_array; // LEDek vezérlése (Pin 51 - Pin 58)  
reg [7:0] led_array;  
reg [20:0] f_divider; // frekvencia-osztó (1 : 2 097 152)  
reg [7:0] counter; // 8-bites bináris számláló  
  
always @ (posedge clk_50M)  
    f_divider <= f_divider +1; // A frekvenciaosztó léptetése  
  
always @ (posedge f_divider[20])  
begin  
    counter = counter + 1; // A 8-bites számláló léptetése  
    led_array = ~counter; // és állapotának kijelzése  
end  
endmodule
```

# Project08: 8-bites léptetőregiszter

- Léptetőregisztert a nem blokkoló értékadások használatával alakíthatunk ki. Például:

```
always @ (posedge clk)
begin
  s_reg[0] <= din;
  s_reg[1] <= s_reg[0];
  s_reg[2] <= s_reg[1];
  s_reg[3] <= s_reg[2];
  s_reg[4] <= s_reg[3];
  s_reg[5] <= s_reg[4];
  s_reg[6] <= s_reg[5];
  s_reg[7] <= s_reg[6];
end
```

ahol **clk** az órajel,  
**din** az adatbemenet,  
[7:0]**s\_reg** pedig a léptetőregiszter.

- A léptetést így is írhatjuk:

```
s_reg <= {s_reg[6:0],din};
```

{... , ... } a konkatenáció (összefűzés)

- A következő projektben 1 Hz-es órajellel léptetünk egy 8-bites regisztert, melynek adatbemenete a K1 gombbal vezérelhető.

# Project08: 8-bites léptetőregiszter

```
module shifter (clk_50M,din,led_array);
input clk_50M; // 50MHz rendszer órajel (Pin 12 bemeneten)
input din; // adatbemenet (K1 gomb, PIN_29)
output [7:0] led_array; // LED kimenetek (PIN_51 - PIN_58)
reg [7:0] led_array;
reg [24: 0] f_count; // 25,000,000-szoros osztó
reg div_clk; // A leosztott órajel
reg [7:0] s_reg; // 8-bites shift regiszter

always @ (posedge clk_50M)
begin
if (f_count == 25000000)
begin
div_clk <= ~div_clk; // 25,000,000 ciklust számlál
f_count <= 0; // 0.5 s időközönként állapotot vált
// számláló törlése
end
else
f_count <= f_count + 1; // számláló léptetése
led_array <= ~s_reg;
end

always @ (posedge div_clk)
begin
s_reg[0] <= din;
s_reg[1] <= s_reg[0];
s_reg[2] <= s_reg[1];
s_reg[3] <= s_reg[2];
s_reg[4] <= s_reg[3];
s_reg[5] <= s_reg[4];
s_reg[6] <= s_reg[5];
s_reg[7] <= s_reg[6];
end
endmodule
```

A 8 bites shift regiszter állapotát a C-M240 kártya 8 LED-je segítségével folyamatosan kijelezzük.

A bitenkénti negáció a LED-ek negatív logikájú vezérlése miatt szükséges.