



TECHNICKÁ UNIVERZITA V LIBERCI
Fakulta mechatroniky, informatiky
a mezioborových studií



Analogové zakázkové obvody

Milan Kolář

Ústav mechatroniky a technické informatiky



evropský
sociální
fond v ČR



EVROPSKÁ UNIE



MINISTERSTVO ŠKOLSTVÍ,
MLÁDEŽE A TĚLOVÝCHOVY



OP Vzdělávání
pro konkurenceschopnost

Projekt ESF CZ.1.07/2.2.00/28.0050
**Modernizace didaktických metod
a inovace výuky technických předmětů.**

INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ



Analogové HDL jazyky

- slouží k modelování a simulaci
- představují potencionální cestu pro analogovou syntézu (zatím není podporována automatická syntéza)
- popisují různé úrovně abstrakce elektrických i neelektrických systémů
- ve srovnání např. s Pspice je popis složitější
- nákladnější simulační programy
- v současnosti nejrozšířenější analogové HDL: VHDL-AMS, Verilog-AMS, SystemC-AMS



VHDL-AMS

1999 - první standard pro analogové a smíšené obvody:
revize VHDL-AMS 1076.1 (Analogue & Mixed Signals).

Analogový systém lze popsat soustavou diferenciálních algebraických rovnic (DAR)

- DAR obecně nemají analytické řešení;
- numerickými metodami hledáme co nejpřesnější aproximace řešení.

VHDL-AMS poskytuje možnost zápisu DAR, ale nespecifikuje způsob jejich řešení (přenecháno konkrétní implementaci simulátoru)
– je specifikován pouze výsledek.



Nové prvky VHDL-AMS

Pro popis systémů pracujících ve spojitém čase

- Modely (entita + architektura)
- Souběžné výrazy (simultaneous) – pro zápis rovnic
- Typy portů (svorek):
 - signál (klasický digitální – in, out, inout)
 - terminal (analogový uzel + energetické domény)
 - quantity (analogová proměnná)
 - generic (obecná konstanta)
- Tolerance
- A/D interakce
- D/A interakce



Množství (quantity)

nová třída objektů pro reprezentaci neznámých v soustavě DAR
(spojitý časový průběh)

Quantity

- je specifikována svým typem a počáteční hodnotou
- může být deklarována v entitě i architektuře (v deklarční části)
- nelze deklarovat uvnitř struktury „package“
- lze použít jako vstupně-výstupní element v seznamu portů

Implicitní quantity:

Q'dot – derivace

Q'inter – integrál

Q'delayed(T) – zpoždění

Q'ltf – Laplaceova transformace

Q'ztf – Z transformace



Tradiční systémy

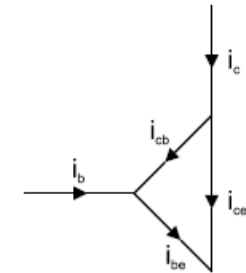
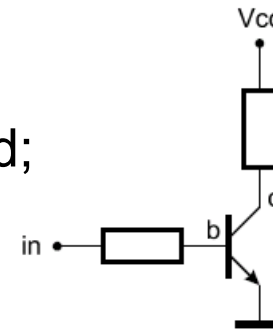
Popis systémů využívajících grafový koncepční model

(popis pomocí Kirchhoffových zákonů)

vrcholy grafu – uzly obvodu;

hrany – větve obvodu, jimiž protéká proud;

dva uzly definují větev (bránu).



Neznámé v rovnicích reprezentují tzv. **branch quantities**

– existují dva druhy:

Across quantities (reprezentují energetické veličiny)

rozdíl potenciálů mezi dvěma vrcholy (U , Θ , p);

Through quantities (představují průtokové veličiny)

odpovídají hranám (proud, tepelný tok, průtok).



Terminály

Nový objekt (např. vývod), který má nějakou přirozenou vlastnost (*nature*) a ke kterému jsou všechny hodnoty dané závislosti vztaženy.

Nature	Across	Through
Electrical	napětí	proud
Thermal	teplota	tepelný tok
Translational	rychlost	síla
Rotational	úhlová rychlost	točivý moment
Hydraulic	tlak	objemový průtok



Příklad – quantity, terminály

```

package elektricky_system is
  subtype napeti is REAL ;
  subtype proud is REAL ;
  nature elektro is -- skalární vlastnost elektro
    napeti across
    proud through
    zem reference ;
end package elektricky_system ;

...

terminal t1, t2 : elektro ;
quantity v across i1, i2 through t1 to t2 ;
  
```




Tolerance

Při řešení soustavy DAR určuje simulátoru, jak velká má být chyba u každé quantity (jak se má blížit nule).

Zavádíme **toleranční skupiny** (*tolerance group*)

– každá quantity i každý výraz náleží do nějaké skupiny.

Není-li tolerance explicitně specifikována, použije se toleranční skupina použité quantity.

```
subtype napeti is REAL tolerance "impl_napeti";  
subtype proud is REAL tolerance "impl_proud";  
i == iss*(exp(v/vt)-1.0) tolerance "male_napeti";
```



A/D a D/A interakce

Pro A/D a D/A převody

Q'ABOVE(E) ... hodnota je TRUE pro $Q > E$ a FALSE pro $Q < E$

Příklad ideálního komparátoru:

```
entity Comparator is
generic (vthresh : REAL);          -- práh
port (terminal ain, ref : electrical;
      signal dout : out BOOLEAN);
end entity Comparator;
architecture Ideal of Comparator is
    quantity vin across ain to ref;
begin
    dout <= vin'above(vthresh);
end architecture Ideal;
```

Příklad diody

Model diody

$$i_d = i_s \cdot \left(e^{(v_d - r_s i_d) / n \cdot v_t} - 1 \right)$$

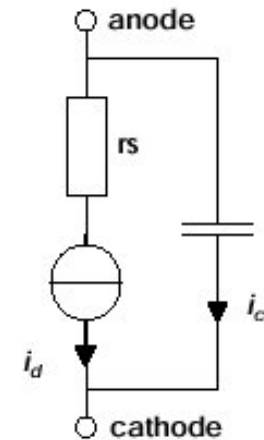
$$i_c = \frac{d}{dt} \left(t_t i_d - 2 \cdot c_j 0 \cdot \sqrt{v_j^2 - v_j v_d} \right)$$

```

library IEEE, Disciplines;
use Disciplines.electrical_system.all;
use IEEE.math_real.all;
entity Diode is
  generic (iss: REAL := 1.0e-14;
           n, af: REAL := 1.0;
           tt, cj0, vj, rs, kf: REAL := 0.0);
  port (terminal anode, cathode: electrical);
end entity Diode;

architecture Level0 of Diode is
  quantity vd across id, ic through anode to cathode;
  quantity qc: charge;
  constant vt: REAL := 0.0258;    -- thermal voltage
begin
  id == iss * (exp((vd-rs*id)/(n*vt)) - 1.0);
  qc == tt*id - 2.0*cj0 * sqrt(vj**2 - vj*vd);
  ic == qc'dot;
end architecture Level0;

```





Příklad - pružina

```
use work.types.all;
entity Vibration is
end entity Vibration;
```

```
architecture H2 of Vibration is
```

```
  quantity x1, x2, xs: displacement;
```

```
  quantity energy: REAL;
```

```
  constant m1, m2: REAL := 1.00794*1.6605655e-24;
```

```
  constant f: REAL := 496183.3;
```

```
begin
```

```
  x1'dot'dot == -f*(x1 - x2) / m1;
```

```
  x2'dot'dot == -f*(x2 - x1) / m2;
```

```
  xs == (m1*x1 + m2*x2)/(m1 + m2);
```

```
  energy == 0.5*(m1*x1'dot**2 + m2*x2'dot**2 + f*(x1-
  x2)**2);
```

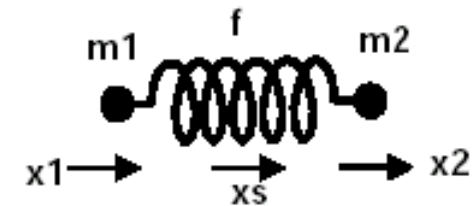
```
end architecture H2;
```

$$m_1 \ddot{x}_1 = -f \cdot (x_1 - x_2)$$

$$m_2 \ddot{x}_2 = -f \cdot (x_2 - x_1)$$

$$x_s = (m_1 x_1 + m_2 x_2) / (m_1 + m_2)$$

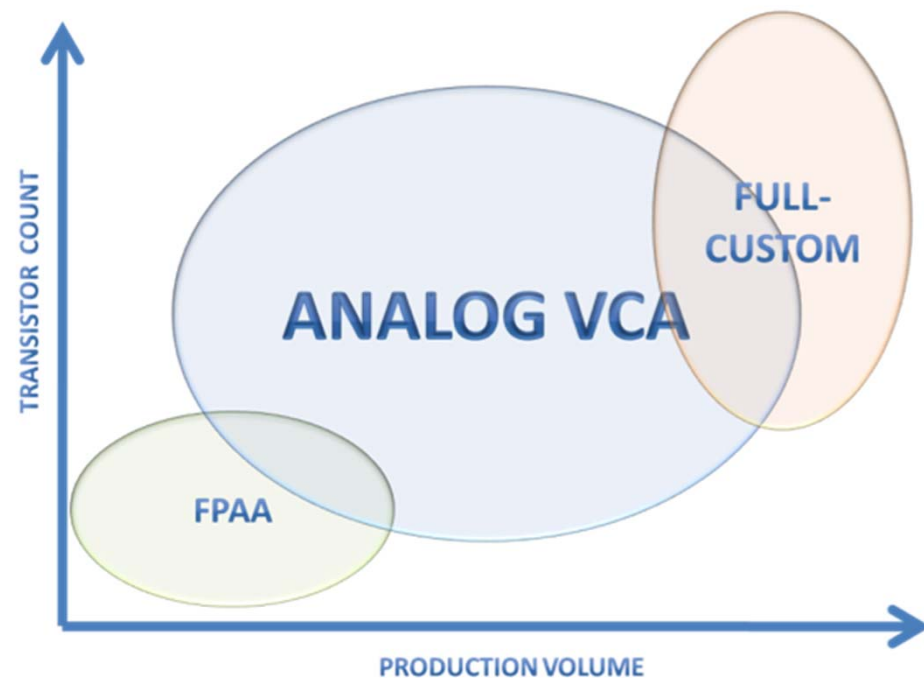
$$Energy = 0.5 \cdot (m_1 \dot{x}_1^2 + m_2 \dot{x}_2^2 + f \cdot (x_1 - x_2)^2)$$





Dělení analogových ASIC

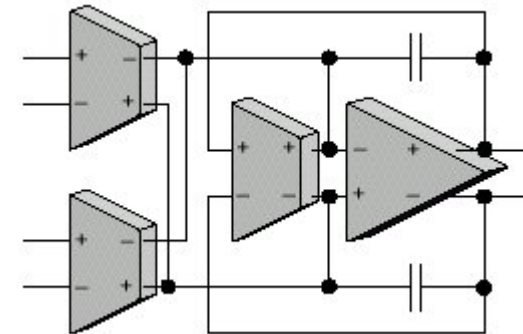
- **Zakázkové (Full-Custom)**
 - parametrizované funkční bloky
 - plně zakázkové IO
- **Via-Configurable Array (polozakázkové)**
 - analogová lineární pole
 - analogová pole obvodů
- **Programovatelné**





Analogové ASIC

- méně rozšířené
- menší hustota integrace
- obtížnější návrh, nutné zkušenosti
- návrh převážně metodou „zdola-nahoru“
- vyšší nároky na přesnost technologie



Specifika analogových technologií

- lepší je tranzistor NPN než PNP
- nelze realizovat induktory
- kapacitory zabírají velkou plochu
- snadná realizace symetrických prvků
- snadnější testování



Prvky analogových IO

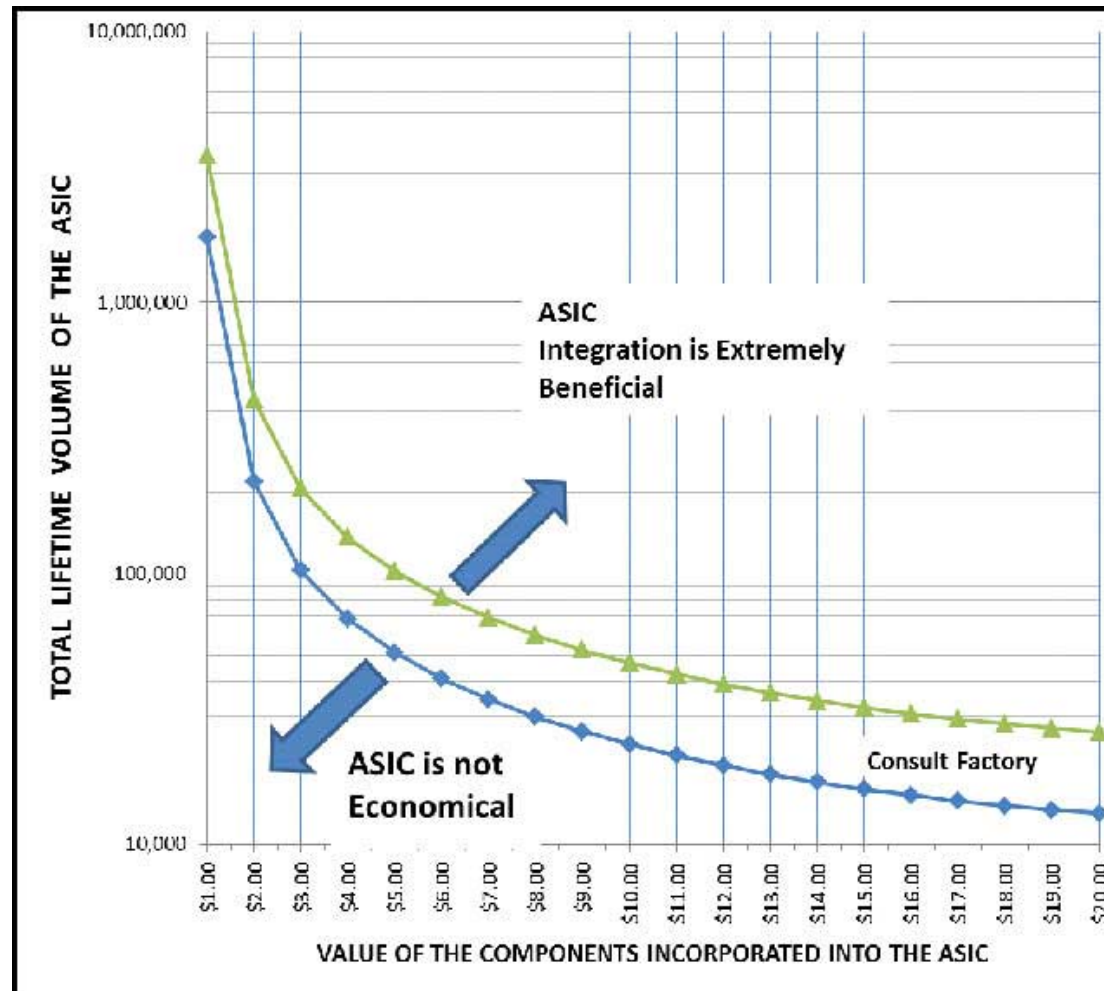
- **rezistory** - meandr izolovaných polovodičových vrstev;
- **kapacity** - relativně malé hodnoty C;
- **diody** - nejčastěji PN přechod tranzistoru;
- **tranzistory** – NPN, PNP (méně kvalitní);

Funkční bloky:

- zdroje proudu (proudová zrcadla), i jako aktivní zátěž;
- zesilovače (diferenční, napěťové, výkonové);
- napěťové referenční zdroje;
- oddělovací (vazební) obvody.



Výhodnost analogových ASIC

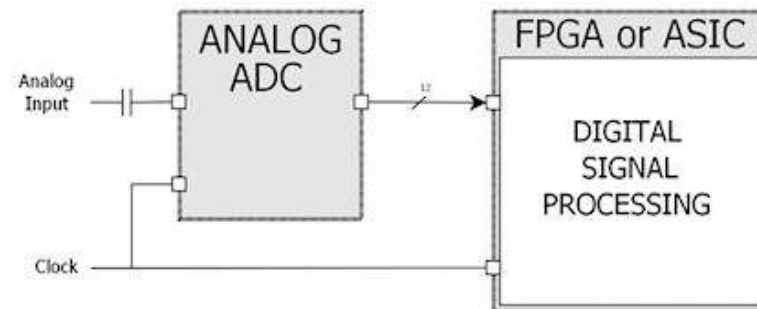


A/D převodníky a FPGA

Implementace A/D převodníku s FPGA

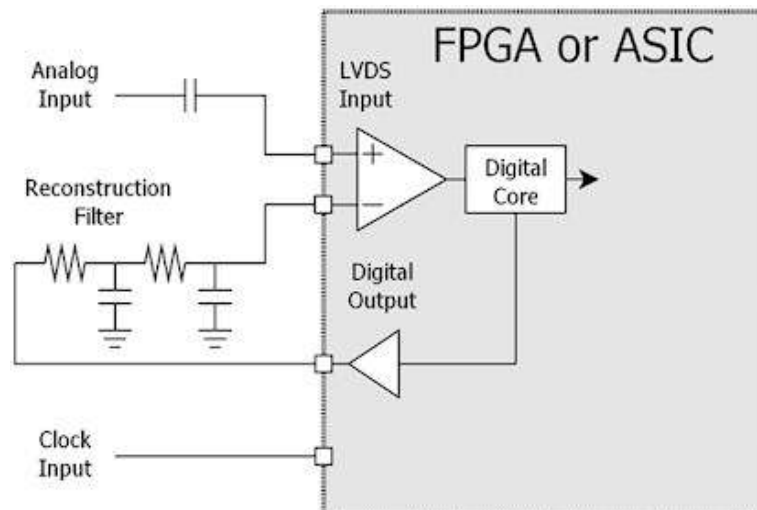
Tradiční řešení s vnějším ADC

- dobré parametry (rychlé, výkonné)
- další IO
- potřeba více pinů



Zaintegrovaný komparátor

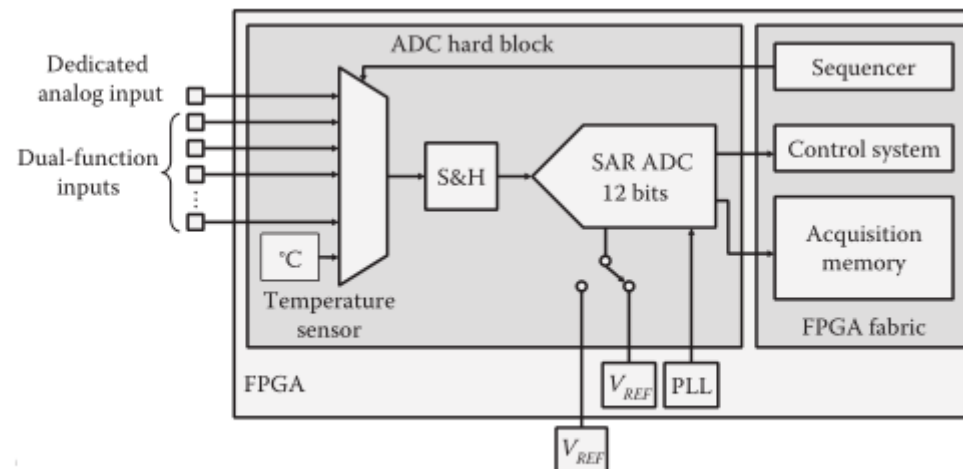
- kompaktní řešení
- zabere navíc jen 2 piny
- zvýšení MTBF
- snazší a rychlejší implementace



A/D převodník v MAX 10

12bitový A/D převodník v FPGA Intel MAX 10

Vkládá se do návrhu jako IP core (makroblok)

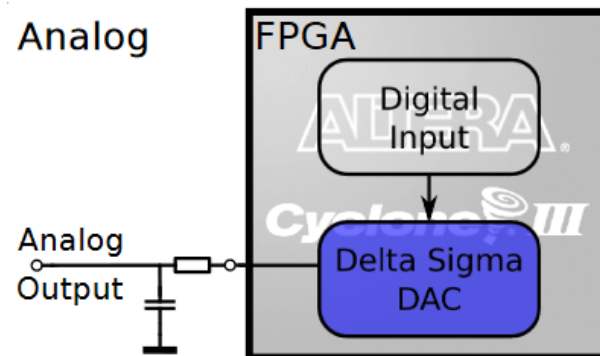


D/A převodníky a FPGA

Implementace není příliš častá.

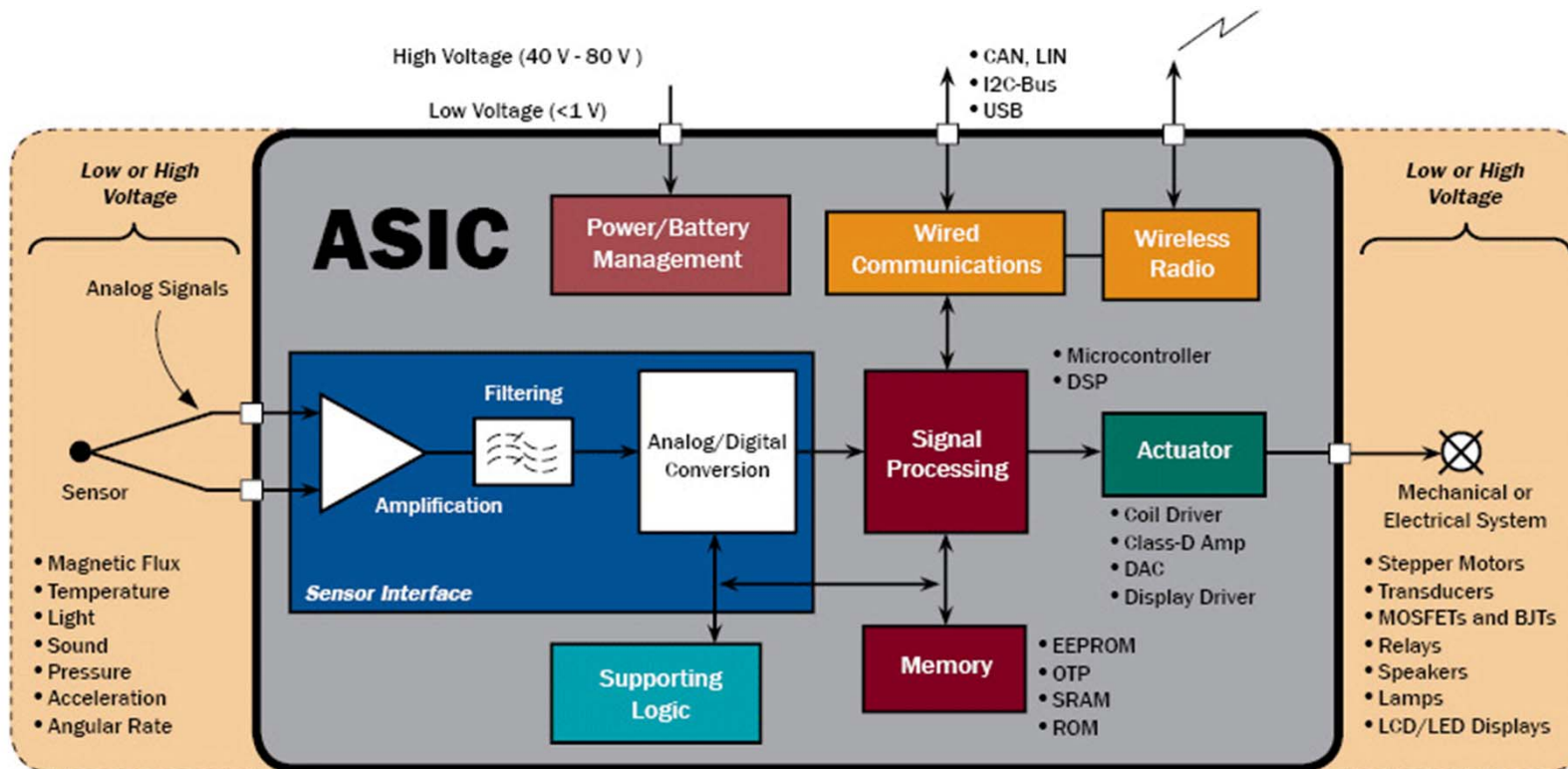
Řeší se např. nepřímými převodníky – nejčastěji principu sigma-delta modulace

- snadná integrace číslicovými technologiemi (relativně složité zapojení)
- vysoká přesnost a linearita



Typický ASIC – smíšený A/Č

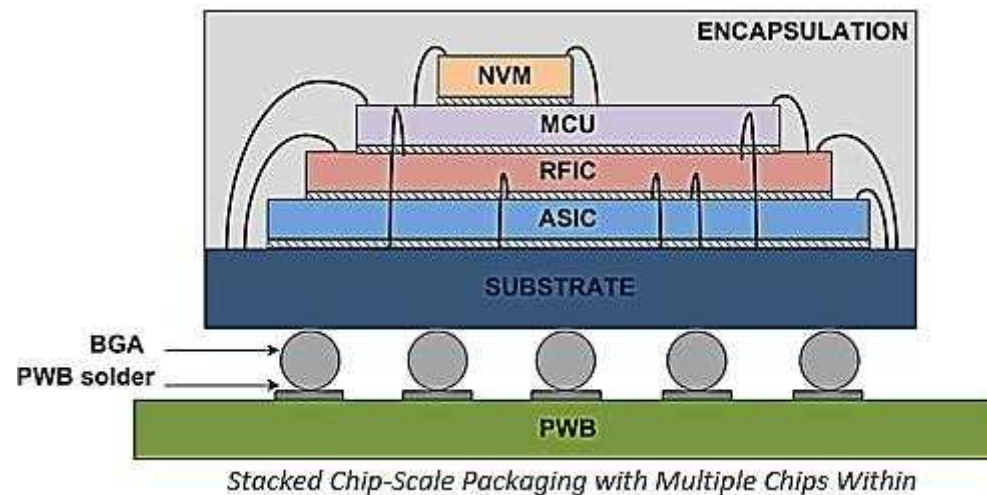
Úprava analogového signálu, digitální zpracování a komunikace





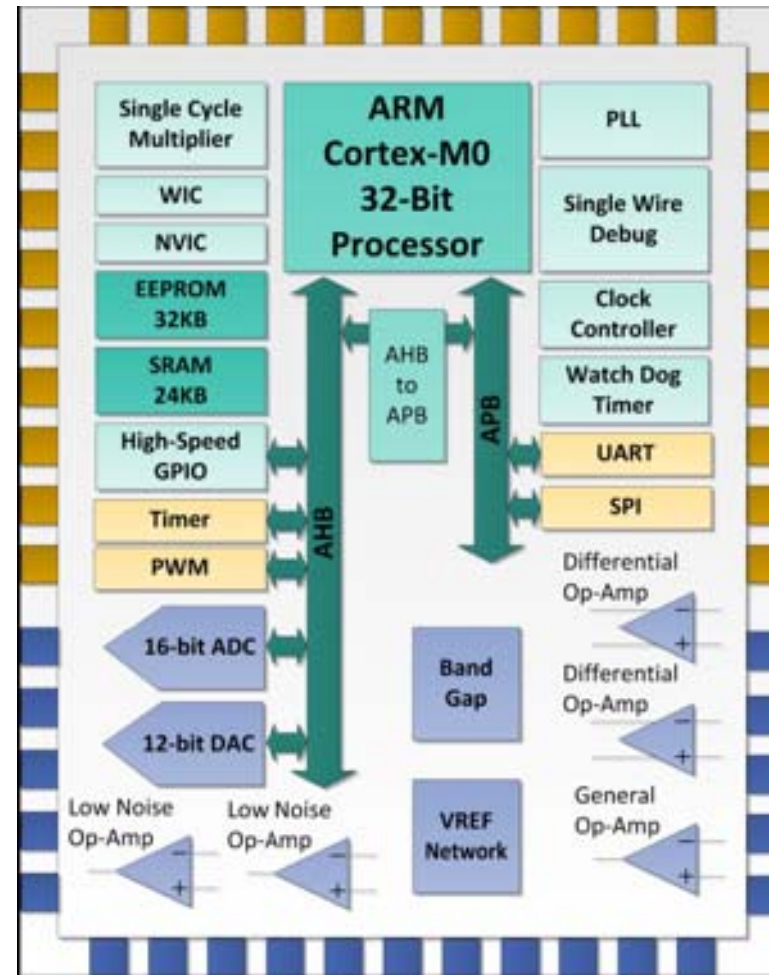
Vícevrstvé čipy

Používání více čipů v pouzdře ve vrstvách
(NVM – NonVolatile Memory, RFIC - Radio Frequency IC,
MCU – MicroController Unit)



Jiná varianta smíšeného A/Č čipu

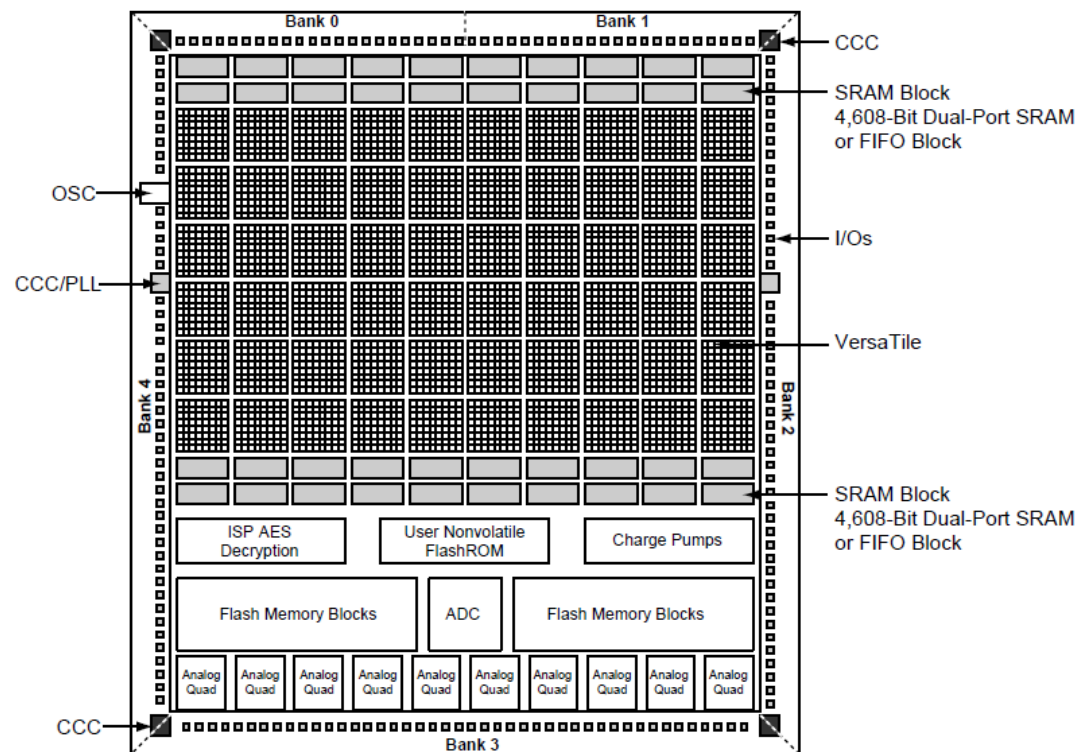
SoC firmy Triad Semiconductor
(integrovaný 32bitový ARM Cortex)



FPGA s analogovými bloky

Řada Fusion (Microsemi):

- ADC (8-/10-/12-bitové s postupnou aproximací),
- DAC (12-bitové sigma-delta),
- komparátory,
- monitory teploty,
- monitory napětí,
- monitory proudu,
- RC oscilátor,
- krystalový oscilátor,
- čítač reálného času.





Programovatelná analogová pole

Rozdílná terminologie (zavedená převážně výrobci):

- **FPAA – Field Programmable Analog Array**
- FPAD – Field Programmable Analog Device
- FPMA – Field Programmable Mixed Analog Digital Array
- EPAC – Electrically Programmable Analog Circuit
- TRAC – Totally Reconfigurable Analog Circuit

Často kombinace analogově-číslicových obvodů:

- AD a DA převodníky, napěťové reference, oscilátory, PWM obvody, generátory zpoždění, zesilovače, násobičky;
- programovatelná konfigurace bloků nebo propojovací sítě.



Programovatelné analogové funkce

Obvody využívající ke změně parametrů digitální programovatelné potenciometry, ovládání přes I2C, SPI nebo inkrementace, resp. dekrementace.

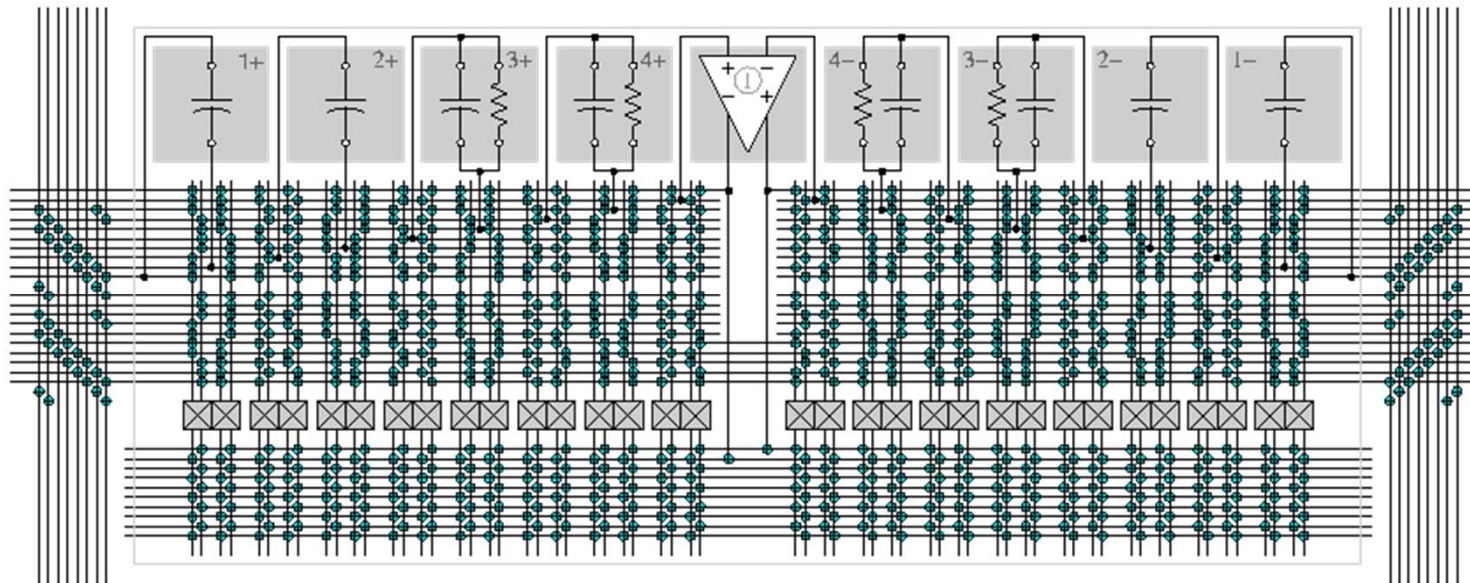
Typické programovatelné obvody:

- napěťové zesilovače,
- přístrojové zesilovače,
- oscilátory (časovače),
- analogové filtry,
- napěťové regulátory,
- zdroje proudu,
- převodníky I/U,
- Schmittovy klopné obvody.



Programovatelná analogová pole

Obecně se skládají z konfigurovatelných analogových bloků (Configurable Analog Block) – nejčastěji OZ a programovatelná kapacitní a odporová pole.





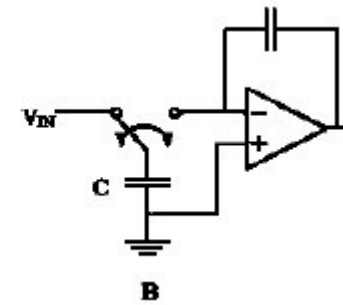
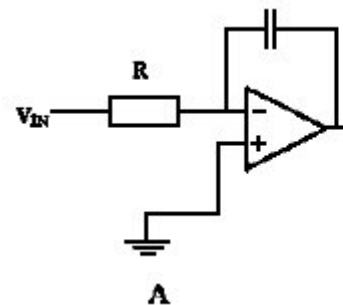
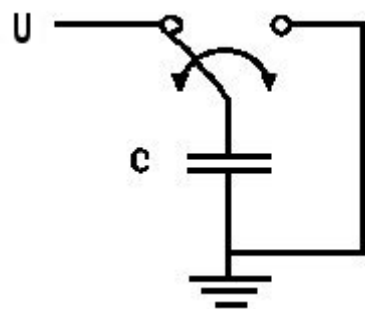
Spínané kondenzátory

Pomocí spínače a kondenzátoru realizujeme odpory

$$R = U / I = U / (\Delta Q / \Delta T) = \Delta T / C = 1 / f \cdot C$$

f ... frekvence přepínání spínače

C ... kapacita kondenzátoru

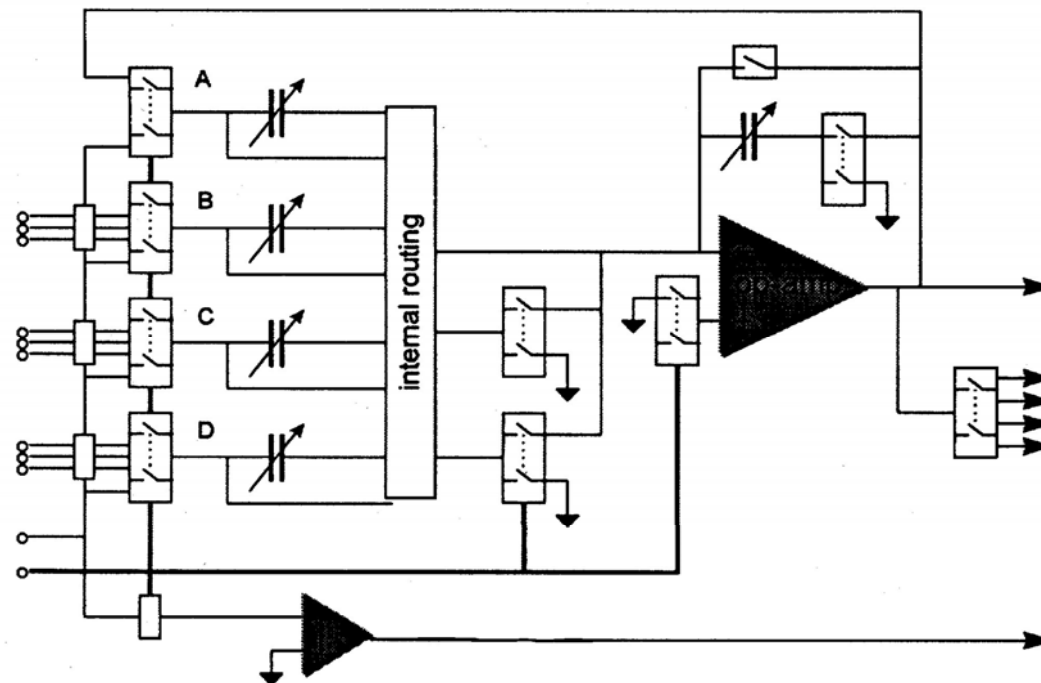




Programovatelná analogová pole

MPAA020 (Motorola) – 20 CAB

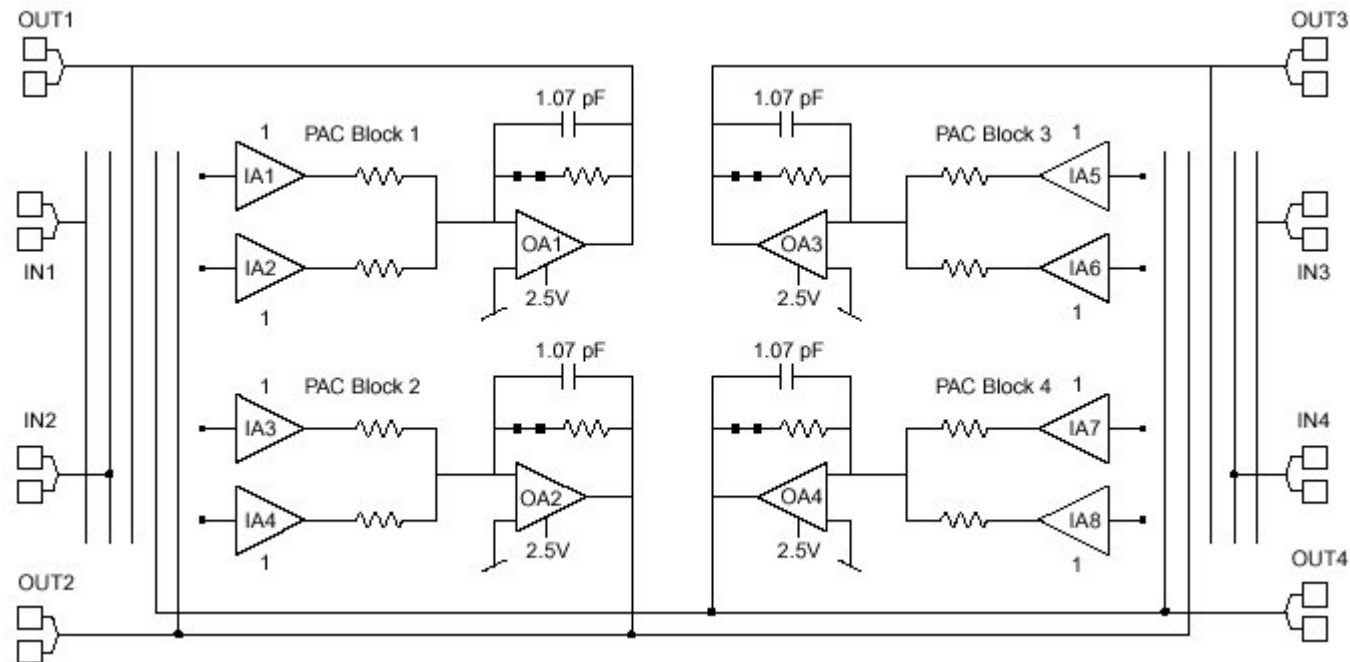
AN220E04 (Anadigm) – 4 CAB (konfigurace v ext. EEPROM)
místo odporů – spínané kondenzátory





Programovatelná analogová pole

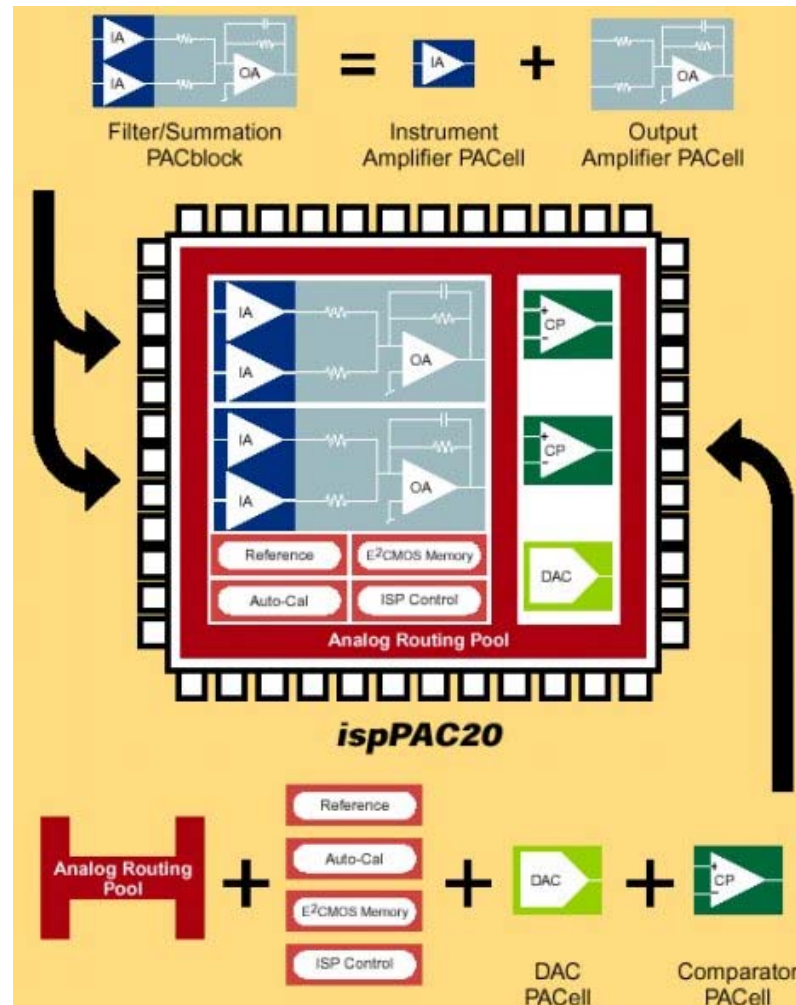
ispPAC10 (Lattice) – programovatelné v systému





Programovatelná analogová pole

ispPAC20 (Lattice)

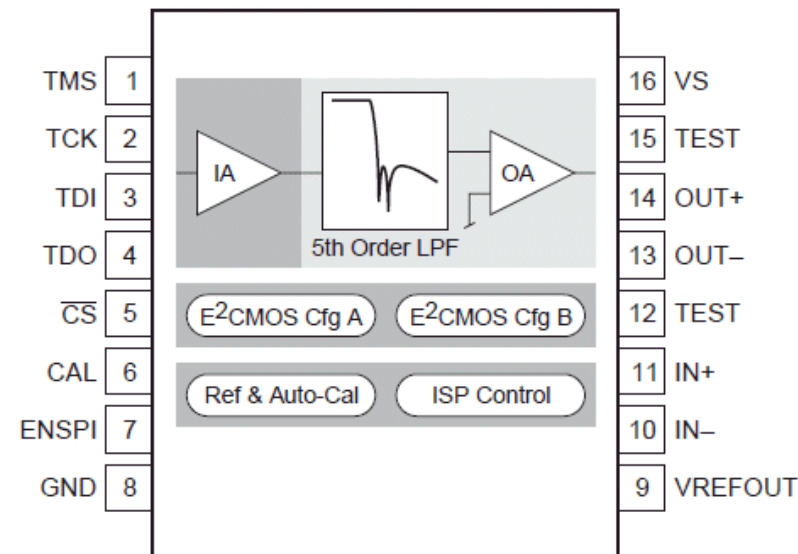




Programovatelný analogový filtr

ispPAC 80 (Lattice)

- duální dolní propust 5. řádu
- mezní frekvence 50–750 kHz
- různé typy aproximace útlumové charakteristiky
- programování přes JTAG
- EEPROM technologie

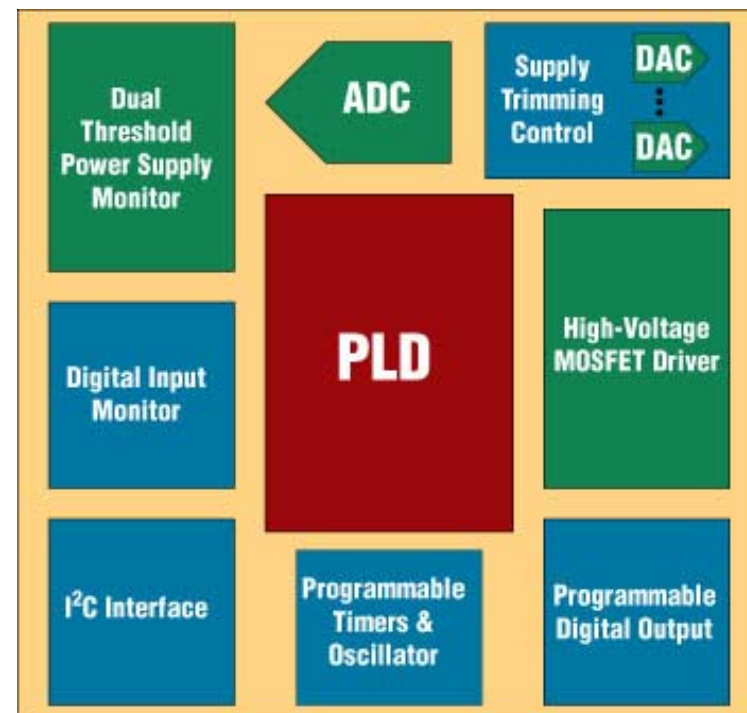




Programovatelný power manager

POWR1220AT8 (Lattice)

- monitoruje až 12 napájecích zdrojů
- 20 digitálních řídicích výstupů
- 8 analogových výstupů
- CPLD (48 makrobuněk)
- 4 časovače (od 32μs do 2s)
- MOSFET budiče
- I²C rozhraní
- programování přes JTAG





Generátor hodinových signálů

ispClock5600A (Lattice)

- vstupní/výstupní frekvence – 8 až 400 MHz
- až 20 programovatelných výstupů (různé nap. standardy)
- až 5 časových domén
- programování přes JTAG

