Proiectul in care-mi desfasor activitatea din cadrul companiei Continental comanda si controleaza cutiile de viteze ale masinilor AUDI.

Este vorba de un proiect cu distributie multicore, in scop de safety si CPU Load Optimization. Manipularea cutiilor de viteze cu software-ul corespunzator se face folosind TCU ( Transmission Control Unit ), alimentate cu o tensiune de minim 7V si maxim 28V.

Daca TCU-ul este alimentat cu mai putin de 7V, DTC-ul de undervoltage se va seta. In mod similar, daca tensiunea de alimentare depaseste cei 28V, DTC-ul de overvoltage isi va face aparitia. DTC-ul (Diagnostic Trobule Code) este folosit pentru a identifica problemele aparute in masina.

In orice proiect automotive, folosim servicii de UDS (Unified Diagnostic Services) care sunt diferite intre ele datorita protcolului utilizat (de exemplu, $11, $19, $23, $34). Prin aceste servicii putem “comunica” cu unitatea de control a transmisiei (TCU).

Pentru a observa DTC-urile setate, trimitem serviciul cu numarul de protocol 19, astfel: $19 02 2F. Raspunsul primit va fi alcatuit de positive response 59 02 2F + codul de 3 bytes + stauts\_byte. Spre exemplu, 59 02 2F xx yy zz tt. Prin “xx”,“yy”,”zz” identificam orice DTC consultand informatiile necesare.

In functie de ultimul byte putem spune daca DTC-ul este active,inactive,s-a healuit sau daca a ramas memorat. Prin “tt” intelegem multimea valorilor {2F,2E,28,2C}.

Totodata, o alta metoda de a detecta daca DTC-ul este activ sau inactiv presupune vizualizarea activ\_bit-ului (daca este pe “1” sau pe “0”) din tabela disponibila de DTC-uri. In aceste conditii, trebuie sa folosim Trace pentru a vizualiza valorile necesare.

In ceea ce priveste proiectul DL382, avem 2 generatii.

In fiecare dintre cele 2 generatii, cu acronimele Gen1, respectiv Gen2, avem 2 metode pentru comanda cutiei de viteze automate: **SBC** (Shift by Cable) si **SBW** (Shift by Wire).

Despre **SBW**, putem spne ca este sistemul prin care modurile de transmisie sunt antrenate/schimbate intr-un automobile prin comenzi electronice, fara a avea vreo legatura mecanica intre maneta schimbatorului de viteze si transmsie.

In mod traditional, schimbarea transmisiei a fost realizata prin legaturi mecanice pentru a pune vehiculul in pozitii cum ar fi Park (P), Reverse (R), Neutru (N), Drive (D), folosind un maner montat pe coloana de directive sau un schimbator de viteze in apropierea consolei centrale.

Prin metoda **SBW**, se elimina spatiul de rutare necesar pentru a acoperi legaturile mecanice dintre schimbatorul de viteze si transmisie. In plus, asigura deplasarea fara effort prin apasarea unui buton sau butoane. Prin aceasta optimizare se elimina orice efort din partea soferului care-si selecteaza viteza.

In cadrul lucrarii mele de diploma, o sa ma axez pe reducerea timpului necesar pentru citirea blocurilor de memorie EEPROM, tinand cont de momentul in care se actioneaza cheia (KL15 ON).

Avand in vedere ca este vorba despre un proiect multicore, imi doresc sa folosesc si sa implic in rezolvarea problemei, toate core-urile disponibile si anume: core0, core1.

In prezent, citirea blocurilor de memorie EEPROM se face doar pe core-ul 0. Astfel, timpul necesar este unul mai mare.

In toate aplicatiile si proiectele din domeniul IT, se urmareste dezvoltarea cat mai curata si eficienta a paralelismului. Aici, avem cateva exemple: Thread-uri, Pipe-uri etc. Stim foarte bine ca, desi se spune despre solutiile amintite anterior ca se executa in paralel, nu este vorba despre un paralelism “curat”.

In cadrul proiectului meu de diploma, imi doresc sa “echilibrez” cele 2 core-uri. Din “calupul” mare, sa impart in 2 parti aproximativ egale din punct de vedere al memoriei si sa le dispun simultan pe cele 2 core-uri.

Astfel, voi ajunge la un timp mai mic decat cel initial.

Totusi, in practica, nu e totul atat de simplu. Trebuie sa tin cont si de Switch Off Path Test (SOPC), care este obligatoriu prezent alaturi de citirea blocurilor de memorie EEPROM. Acesta este un element de siguranta, permitand masinii sa se opreasca in orice moment in care a detectat o eroare aparuta.

--->--------------------------------------- de ilustrat

--------------------timestamp

( poze )

**Introducere**

**Context**

Într-o lume aflată într-un proces de evoluţie continuu, cu toţii suntem conştienţi că principalul mijloc de deplasare este automobilul. Fiind un element esenţial al vieţii noastre cotidiene, automobilul, constituie un motiv bine întemeiat pentru rapiditatea cu care avansează industria automotive.

Totodată, necesitatea dezvoltării unor noi tehnologii în acest vast domeniu este foarte ridicată. Atât dezvoltarea noilor soluţii, cât şi testarea lor, reprezintă elemente demne de o atenţie sporită, deoarece chiar şi o greşeală minoră poate fi fatală.

**Situaţie actuală**

Durata inițializării unității de control a transmisiei, adică a **TCU-**ului, depinde într-o proporție covârșitoare de citirea datelor din EEPROM.

În cadrul companiei Continental, divizia Powertrain Transmission, există un proiect cu distribuţie multicore care foloseşte o unitate de control a transmisiei, **TCU**, pentru a manipula cutia de viteze a maşinilor Audi.

În trecut, folosind un proiect mai vechi, era utilizată o unitate de control a transmisiei cu un singur nucleu pentru a “manevra” cutia de viteze a maşinilor Audi.

Atât în proiectul “părinte”, cu singur core, cât şi în proiectul recent (cel multicore), citirea blocurilor de memorie EEPROM se face pe un singur core.

Dacă în proiectul vechi, soluţia de citire pe un singur core era singura disponibilă, pe cel multicore se doreşte distribuirea blocurilor de memorie EEPROM într-un mod cât mai avantajos pe 2 core-uri.

**Temă propriu-zisă**

Având în vedere că în noul proiect unitatea de control a transmisiei, **TCU-ul**, dispune de un microcontroler cu 3 nuclee, îmi doresc să îmbunătăţesc considerabil durata de citire a blocurilor de memorie EEPROM. Astfel, din momentul în care s-a acţionat cheia de pornire a maşinii (KL15 => ON), să dispun blocurile într-un mod avantajos şi benefic pe 2 core-uri.

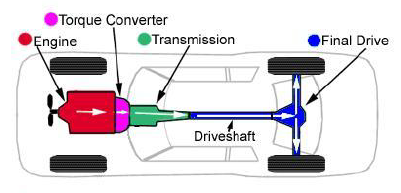
Ţinând seama de faptul că microcontrolerul “primeşte” informaţii de la chipul memoriei EEPROM prin intermediul interfeţei SPI, rezultă că funcţionalitatea interfeţei trebuie “transferată“ şi pe al-2lea core ce urmează a fi utilizat (pe core-ul 0, utilizat iniţial, interfaţa a fost implementată).

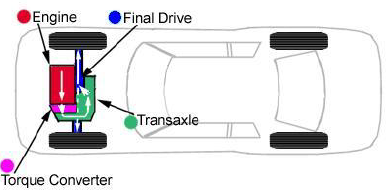
**Prezentarea sistemului de transmisie a puterii**

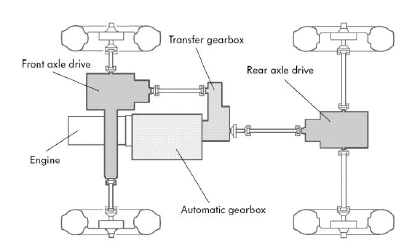
* **Sistemul de tracțiune**

Sistemul de tracţiune reprezintă ansamblul componentelor ce fac posibilă aducerea şi menţinerea unui automobil în stare de mişcare.

În funcţie de tipul automobilului, desprindem mai multe categorii de tracţiune:

* Tracţiune spate, întâlnită cu precădere la maşinile sport (Lamborghini, Mercedes , BMW, Ferrari)
* Tracţiune faţă, întâlnită la majoritatea maşinilor mici (Dacia, Renault, Ford, Chevrolet, VW, Opel)



* Tracţiune integrală, 4x4, întâlnită la SUV-uri (BMW X5, BMW X6, Audi Q7, VW Touareg)
* **Elementele unui sistem de transmisie a puterii**
* **Ambreiajul**

Ambreiajul este un organ de mașină care prin cuplare transmite un moment de putere a unei mișcări de rotație, sau întrerupe această mișcare de rotație prin decuplare. Activarea ambreiajului se poate face atât în timp ce acesta se rotește sau în timp ce stă neînvârtit.



* **Cutia de viteze**

Cutia de viteze este un ansamblu de roți dințate care servește la transformarea forței și transmiterea mișcării de rotație la rotile autovehiculului.

Raportul de transmisie este raportul dintre viteza de rotație a roților dințate ce compun cutia de viteze și viteza de rotație a roților automobilului.

Cutiile de viteze se împart în mai multe categorii, în funcție de:

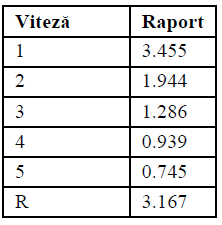
1. Raportul de transmisie:

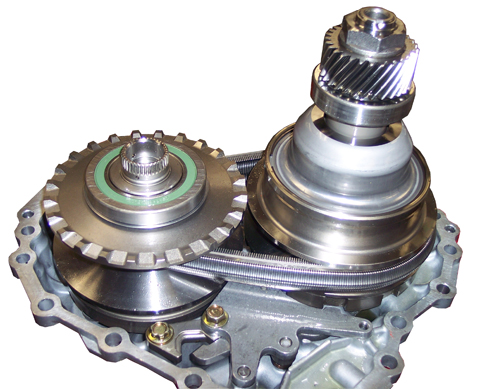
* cu raport de transmisie fix
* cu raport de transmisie variabil (CVT)

1. Gradul de automatizare:

* manuală
* CVT (Continuously Variable Transmission)
* Automată

Ȋn cazul cutiei de viteze manuale, pentru fiecare viteză a cutiei este cuplată o roată dințată a acesteia pentru care există un raport de transmisie fix, ca în imaginea de mai jos:



**Cutiile de viteze CVT (Continuously Variable Transmission)**, spre deosebire de cutiile manuale, au capacitatea de a parcurge un număr infinit de rapoarte de transmisie. CVT oferă eficiență sporită în ceea ce privește consumul de carburant, deoarece îi permite motorului să funcționeze la cea mai eficientă turație.

**Cutiile de viteze automate** sunt cutiile care realizează schimbarea treptelor de viteză fără intervenția conducătorului automobilului. Mai mult, decizia de schimbare a treptelor de viteză este luată de asemenea automat, pe baza informațiilor provenite de la diferiți senzori. Realizarea unei trepte de viteză se face prin intermediul mai multor mecanisme planetare.



**Cutiile de viteze automate cu dublu ambreiaj**, din punct de vedere cinematic, sunt de fapt compuse din două cutii de viteze manuale, dispuse în paralel. Practic, în aceeași carcasă avem două cutii de viteze, fiecare cu propriul ambreiaj, o cutie conținând treptele impare (1, 3, etc.) iar a doua treptele pare (2, 4, etc.). Acest principiu permite ca vitezele să fie schimbate fără să se piardă din putere. Ȋn timp ce unul din ambreiaje transmite puterea, celălalt este pregătit pentru a cupla următoarea viteză, care este preselectată. Astfel, schimbul vitezei se realizează într-o fracțiune de secundă.

**Moduri de comunicaţie ale TCU-ului**

Privind interacțiunea TCU-ului cu celelalte elemente, desprindem 2 moduri prin care ele comunică între ele:

1. **Controller Area Network (CAN)**

* Modul autonom, poate organiza singur transmisia și recepția datelor conform specificațiilor
* Chipul modului de CAN trimite si recepționează în mod general și standard date cu 11 biți de identificare
* Are implementat și modul extins de date, cu 29 biți de identificare
* Se folosesc 2 pini pentru a interfața cu magistrala de transmisie
* Pinii respectivi trebuie să fie legați între ei printr-o rezistență de 150 Ω, deoarece fiecare fir are o tensiune diferită, rezultând o cădere de tensiune între cele două fire
* Modul folosit pentru comunicarea facilă între calculator și microcontroller
* Programul scris în limbajul C este compilat și apoi se generează un fișier cu extensia .hex
* Acest fișier .hex este încărcat într-un program ce se folosește de transmisia prin CAN și flash-uiește microcontroller-ul cu programul dorit

1. **FlexRay**

* Este un protocol de comunicații în domeniul automotive, dezvoltat de Consorțiul Flexray
* Acesta este conceput pentru a fi mai rapid și mai fiabil decât CAN și TTP
* Diferența de calitate față de celelalte protocoale se resimte în preț, fiind mult mai costisitor
* Prima dată a fost folosit in producția seriei de vehicule BMW X5
* Utilizarea completă a fost introdusă în 2008 la noul BMW Seria 7
* Pe langă vehiculele amintite anterior, a mai fost folosit la producerea:
  + - Audi A4
    - Audi A6
    - Audi A7
    - Audi A8
    - Lamborghini Huracan
    - Mercedes-Benz S-Class

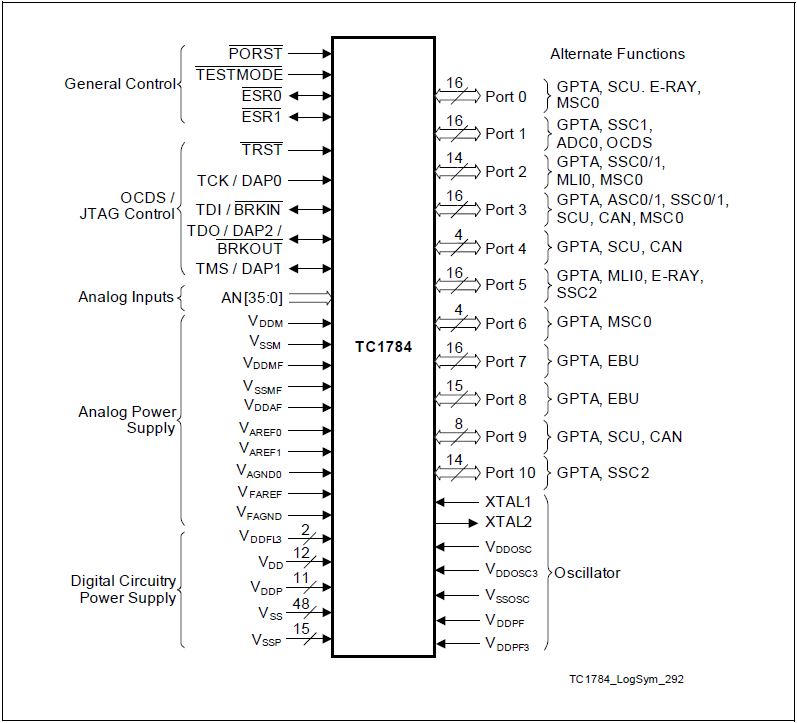
Interfața pentru rețeaua FlexRay, **VN7600**, este ideală pentru dezvoltarea, simularea sau testarea rețelelor FlexRay. Posibilitățile de conectare prin USB fac ca aceasta să fie aplicabilă în mai multe locuri diferite (aici putem aminti ca locuri aplicabile laboratorul sau chiar pe durata unui test-drive).

În plus, interfețele oferă acces la CAN.

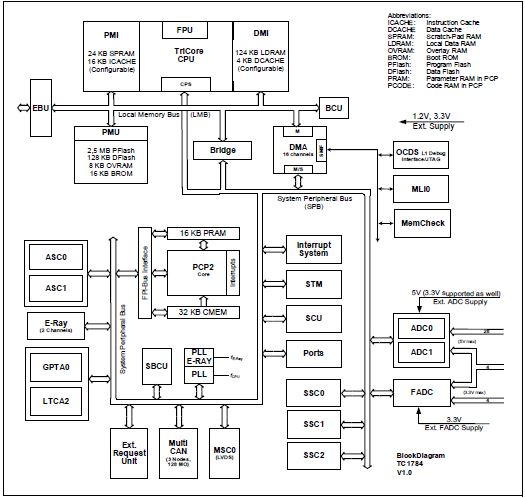


În cadrul proiectului desfășurat de către compania Continental, au existat 2 generații:

1. Generația 1- după multe căutari succesive ale unui microcontroller satisfăcător, după înlocuiri multiple, s-a ajuns la ideea de a folosi un microcontroller de la Infineon și anume Infineon Tricore [IFX] TC1784

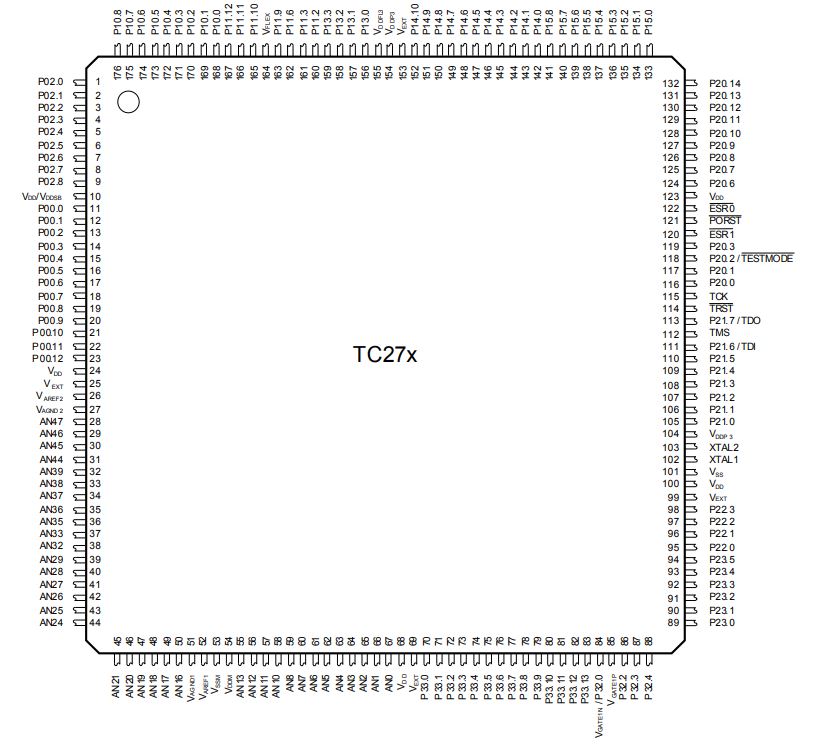


În continuare, o să ilustrez schema bloc a microcontroller-ului Infineon TC1784:

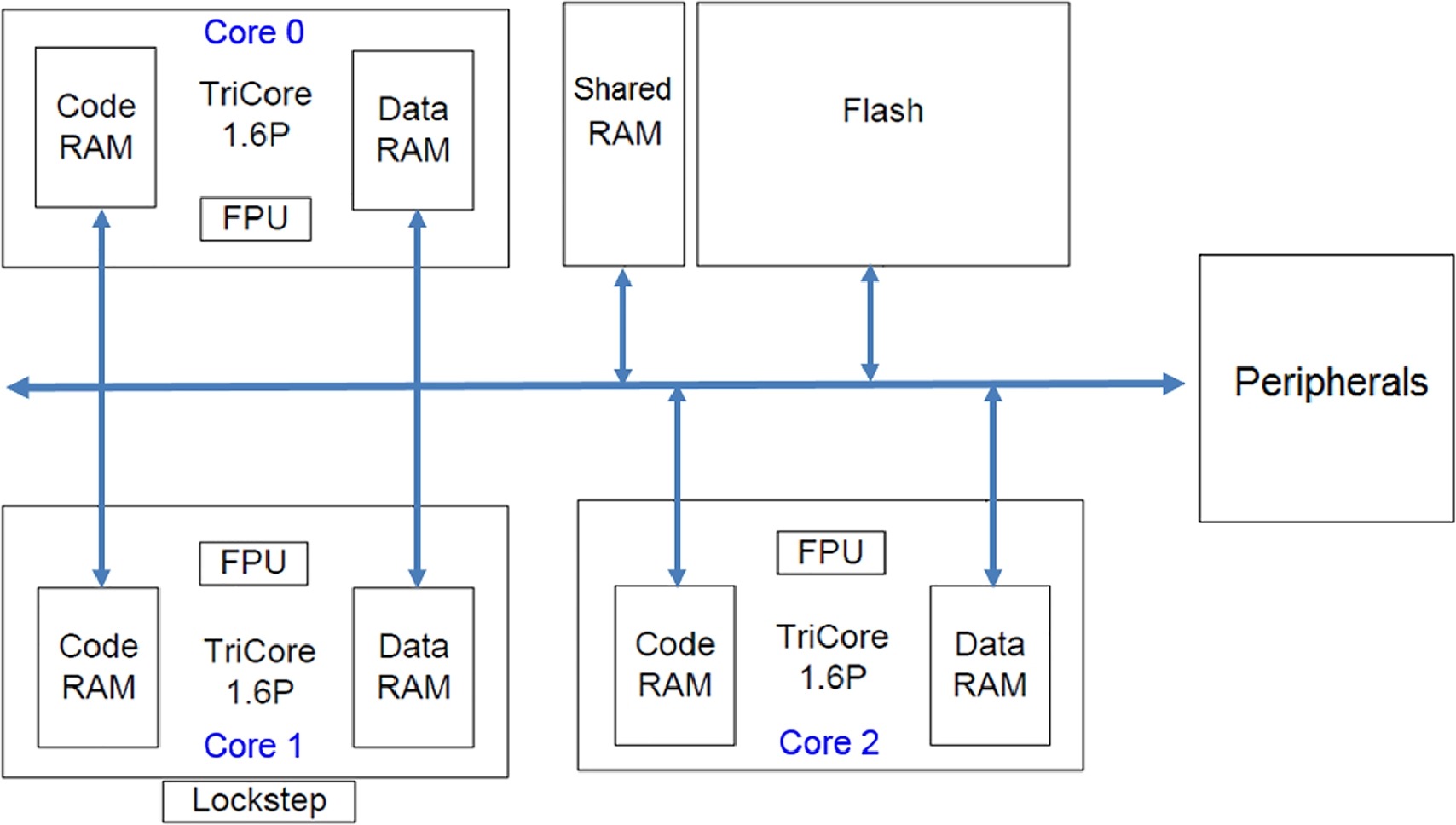


* Din punct de vedere al construcției acestuia, desprindem următoarele caracteristici:
  + - Frecvența maximă a CPU-ului: 180 MHz
    - Frecvența maximă a PCP-ului: 180 MHz
    - Frecvența maximă a sistemului: 90MHz
* Structural și privind CPU-ul, microcontroller-ul TC1784 include un CPU de mare performanță și un procesor de control periferic-**PCP**.
* În ceea ce privește PCP-ul, acesta este un procesor de control periferic foarte flexibil optimizat pentru controlul întreruperilor, “descărcând” astfel CPU-ul.
* Totodată, TC1784 include un controller DMA (Direct Memory Access) cu 16 canale DMA independente.
* TC1784 include un sistem programabil de întrerupere cu următoarele caracteristici:
* Răspuns rapid de întrerupere
* Sisteme independente de întrerupere pentru CPU și PCP
* Noduri de cereri de servicii programabile (SRN)
* Fiecare SRN poate fi mapat la sistemul de întreruperi al CPU-ului sau PCP-ului
* Schemă flexibilă de prioritizare a întreruperilor cu 255 de nivele de prioritate întreruperi per întrerupere sistem

1. Generația 2, a folosit un alt microcontroller de la Infineon și anume Aurix TC277:



În continuare, o să prezint schema bloc simplificată a micrcontroller-ului Aurix TC277:



* Tehnologie TriCore, **core0** **core1 core2**, cu o frecvență de până la 200 MHz per core
* Suportă operațiile cu virgulă flotantă și virgule fixă cu toate core-urile
* Temperatură: -40°C ... 145°C
* Alimentat la 5.5V sau 3V
* În ceea ce privește conectivitatea, amintesc:
  + - FlexRay
    - SPI
    - CAN
    - LIN
    - UART
    - Ethernet