**Ligjerata 6**

**CPU** në një mikrokontroller zakonisht është një bërthamë e vetme e një MPU (micro processor unit).

**CPU përbëhet nga komponentat**:

* Harduerike:
  + ALU – Arithmetic logic unit.
  + CU – Control Unit.
  + BIL – Bus Interface Logic.
  + Registers.
* Softuerike:
  + Instruksionet.
  + Modelet e adresimit.

Diagram

Description automatically generated

**Control Unit**: Menaxhon punën e CPU, duke e mbajtur atë gjatë gjithë kohës brenda një cikli të quajtur “*instrucation cycle”* ky cikël në vete përfshinë tri faza:

* *Fetch* – CU udhëzonë BIL që të marrë instruksionin nga memorie me adresën që gjendet në regjistrin *PC* dhe ta vendosë atë në regjistrin *IR*.
* *Decode* – Pasi që instruksioni të jet i vendosur në *IR*, cikli kalon në fazën e dekodimit, në këtë fazë CU urdhëron dekodimin e instruksionit që gjenden në *IR*, për të fituar një rezulatat të dekoduar të cilin CU e përdorë për udhëzimin (lëshimin e control signals) e pjesëve të tjera të CPU në bazë të instruksionit.
* *Execute* – Pasi që instruksioni të jetë dekoduar CU e përdorë për të udhëzuar(lëshonë control signals) pjesët e tjera të CPU për të i përmbushur kërkesat e instruksionit, në fazën e fundit të ekzekutimit CU kujdeset për rritjen e vlerës që gjendet në *PC*. Pasi të përfundojë kjo fazë CU udhëzon BIL që të kthehet në fazën e parë duke e detyruar që të bëj fetch nga adresa aktuale në *PC*.

*CU i duhet një sinjalë Reset për të nisur ciklin për herë të parë, me pranimin e këtij sinjali CPU lexon nga ROM për ta bërë boot BIOS*.

**ALU**: Është pjesë përbërëse e CPU e cila bën kryerjen e veprimeve aritmetike si mbledhje, zbritje, shumëzim dhe pjestim, operacionet logjike që përfshihen në një ALU janë: OR, AND, XOR, NOT, SHIFT, ROTATE.

**BIL**: Është struktura e CPU që koordinon ndërveprimet në mes magjistraleve të CPU dhe atyre të sistemit, BIL definon se si busat e jashtëm (*control bus, address bus, data bus)* operojnë.

**Registers**: Janë memorie e pa qëndrueshme me shpejtësi të lartë dhe kapacitet të vogël të cila gjenden shumë afër CPU, pra janë të integruara brenda mikroprocesorit. Regjistrat përdoren për të ruajtur: *të dhëna, adresa të memories ose informata që i duhen CU*.

Regjistrat i klasifikojmë në: *Regjistra me qëllim të specializuar* dhe *Regjistra me qëllim të përgjithshm*.

**Regjistrat me qëllim të përgjithshëm** (GPR): Nuk janë të lidhur me ndonjë funksion që e kryen CPU dhe mund të përdoren për ruajtien e *të dhënave, variablave, adresave.* Shpesh quhen edhe si regjistra të adresave ose të dhënave. Në vartësi të arkitekturës një CPU mund të ketë dy ose më shumë GPR.

**Regjistrat me qëllim të specializuar**: Janë regjistra që përdoren nga CPU për të kryer funksionet e caktuara, pra në ta ruhen informata rreth funksionit të CPU për të cilin regjistri është i krijuar.

**Struktura themelore e një CPU përbëhet nga katër regjistra të specializuar**:

* **Program Counter (PC) ose Instruction Pointer (IP)**: Ky regjistër mban adresën e instruksionit të radhës të cilin CPU do ta bëj fetch nga memoria pasi të përfundojë ekzekutimin i udhëzimit që gjendet në *IR*. Sa herë që një instruksion dekodohet dhe ekzekutohet vlera e këtij regjistri rritet për një. Me qenë se ky regjistër mban adresa të memories së programit, gjërësia e tij mund ta caktojë madhësinë e memories së programit.
* **Stack Pointer (SP)**: Për të kuptuar pse përdoret ky regjistër së parit duhet ta kupojmë stack, stack është një segment i memories (shpesh i vendosur në RAM) në të cilin të dhënat futen nga lart dhe nxierren nga lartë (LIFO), ofron dy operacione *POP* – nxjerrë infromata nga *TOS* (top of the stack) dhe *PUSH* – vendos informata një më lartë se *TOS* aktual, nëse kjo rrjedhë e operacioneve nuk mbahet atëherë nuk mund të quhet stack, për ta mbajtur këtë rrjedhë të informatave CPU përdorë *Stack Pointer* regjistrin në të cilin ruhet adresa e TOS.
* **Status Register (SR) ose Processor Status Word (PSW) or Flag Register (FR)**: Në këtë regjistër ruhen disa bita të quajtur *flag* (flag është një bit i cili sinjalizon paraqitjen një kushti të caktuar) si dhe bita të tjerë që kontrollojnë statusin e CPU. Shumica e *flag* bitave në SR e reflektojnë sistuatën pasi kryerjes së një operacioni nga *ALU*.
  + *Flag* që gjenden gati në gjdo MCU:
    - *Zero Flag*: Ndryshe quhet biti zero, vendoset në *SR* kur rezultati i një operacioni të *ALU* është zero.
    - *Carry Flag*: Vendoset në *SR* kur rezultati i një operacioni në *ALU* rezulton me një bartje (carry).
    - *Sign Flag*: Ky flag vendoset në *SR* nëse rezultati i një operacioni në *ALU* është negativ, në faktë ky flag reflektonë “most significat bit” të rezultatit.
    - *Overflow Flag*: Ky flag vendose kur rezultati aritemtik i mbledhies ose i zbritjes me numra negativ jep një rezultat jashtë range.
    - *Interrupt Flag ose General Interruption Enbable* (GIE): Ky flag nuk është i lidhur me *ALU*, ky flag tregon nëse programi lejon që të bëhet *interrupted* prej një eventi të jashtëm (interrupt) ose jo.
* **Instrcution Register (IR)**: Ky regjistër mbanë udhëzimin (instruction) aktual i cili është duke i dekoduar dhe ekzekutuar nga CPU, procesi i vendosies së udhëzimit nga memoria në këtë regjistër quhet *fetch*, në shumicën e mikrokontrollerëve gjejmë vetëm nga një *IR*, mirpo në sistemet performative mund të hasim në shumë *IR* të vendosur në një queue, të cilat mundësojnë ofrimin e shumë udhëzimeve njëkohësishtë, në raste të tilla *IR* quhet *Instruction Queue* (rradhë udhëzimi).

**MSP430 CPU**: Është një CPU 16-bit në versionin e vjetër dhe 20-bit versioni i ri (CPUX), përbëhet nga 16 regjistra 16 bitësh. Këta regjistra fillojnë nga *R0, R1,..R15*. Regjistrat nga R4 deri në R15 janë GPR ose regjistra me qëllim të përgjithshëm, ndërsa 4 regjistrat e parë janë me qëllim specifik:

* R0ose Program Counter (PC).
* R1 ose Stack Pointer (SP).
* R2 ose Status Register (SR) i cili ka funksion të dyfishtë edhe si *Constant Generator* CG1.
* R3 ose Constant Generator 2 (CG2) i cili është i përdorur për të vendosur instruksionet pra si një IR (Insctruction Register).

*E veqantë e këtij procesori është se PC dhe SP mbajnë gjithmonë adressa të njëjta dhe kanë last significat bit e barabart me 0, duke lejuar që kta dy regjistra të përdoren si operand dhe të mundësojnë krijimin e aplikacioneve duke përdorur algoritme më të thjeshta*.

Ky CPU në *SR* përmban *flags* si: Zero, Overflow, Carry, Interrupt (GIE), Sign, përveq këtyre mban edhe bita si *CPUOFF, OSCOFF, SGC1* dhe *SGC0*.

A picture containing graphical user interface

Description automatically generated

*ALU* e këtij CPU ka kapacitet 16-bit ndërsa i riu *MSP430 (CPUX)* ka kapacitet nga 16-20 bit. *ALU* kryen veprimet aritmetike si: mbledhje me dhe pa bartje, zbritje me dhe pa bartje, mbledhje dhjetore me dhe pa bartje. Këto operacione që i kryen *ALU* rezultojnë në: *overflow, carry, sign* e për të cilat përdoren *flags*.

*Operactionet logjike që ndikojnë në flags*: AND, XOR.

*Operacionet logjike që nuk ndikojnë në flags*: OR, Bit Clear.

*Bus-at (magjistralet) e sistemit*: Janë linja transemtuese që i mundësojnë CPU komunikimin me Memorien dhe paisjet I/O.

**Bus-at e sistemit i ndajmë në**:

* *Data Bus*: Përdoret për bartien e të dhënave dhe instruksioneve nga memoria ose I/O në CPU dhe anasjellas, pra këto linja kanë drejtim dy kahësh, kur bëhet kalim i të dhënave nga CPU në memorie ose I/O e quajmë *shkrim* (write) ndërsa kur ndodhë kalim i të dhënave nga memoria ose I/O në CPU e quajmë *lexim* (fetch). Një transferim i të dhënave nëpër *bus* e quajmë “*data bus transcation*”, numri i linjave të *data bus* përcaktonë sasinë e të dhënave që mund të transferohen me një transaksion të vetëm. Nëse kemi *data bus* 8-bit, atëherë me një transaksion mund të transferohet vetëm një bajt, transferi i më shumë bajt-ëve kërkonë transaksione të shumëfishta.
* *Address Bus*: CPU mund të mar të dhëna nga vetëm një lokacion i memories ose I/O paisjeve njëkohësishtë, për tu qasur në këto lokacione CPU i duhet të ashtu quajturat *Adresa* të cilat janë numra integer sekuencial dhe reprezentojnë një lokacion në memorie ose në I/O paisje, grupi i linjave që bëjnë bartien e këtyre adresave i quajmë *Address Bus*, këto linja janë një direktionale pra dalin vetëm nga CPU në drejtim të memories ose I/O. Gjërësia e këtij busi determinon adresën më të madhe të memories që CPU mund ta adresojë, një bus me *m* bitë mund të adresojë më së shumti 2*m* adresa. Në rastin e busit me 16-bit do të kemi 216 = 26 \* 210 = 64 KB që fillojnë nga 0x0000 deri në 0xFFFF. Secili bit i adresës vepron si grup dhe quhen *word address* nuk kanë kuptim individualisht

A picture containing text, document

Description automatically generated

Text

Description automatically generated

* *Control Bus*: Përbëhet nga linja të cilat bëjnë bartien e sinjaleve të kontrollit të cilat rregullojnë aktivitetin e sistemit, për dallim nga *data bus* dhe *address bus* të cilat linja i trajtonin si grumbull (dhënash ose adresash), linjat e *Control Bus* trajtohen dhe veprojnë si të ndara. Shumica e linjave të kontrollit janë dy derjtimëshe të cila hyjnë ose dalin nga CPU. Sinjale të kontrollit janë sinjale që tregojnë se kur CPU është duke lexuar, shkruajtur, sinjale që kërkojnë shërbime nga CPU, sinjale që tregojnë kur një transaksion fillone dhe kur mbaronë.