

### Различити сетови инструкција

Кроз историју развоја микропроцесора било је битно развити технологије изградње микропроцесора које би подржале брже извршавање инструкција.

Исто тако, потребно је било повећати и број извршених операција у јединици времена.

Ова два захтева су често била супротстављена један другоме и понекад је у приступу изградње новог микропроцесора било потребно жртвовати један од ова два захтева.

Америчка компанија IBM је истраживањем закључила да већина микропроцесора користи само 20% од целокупног броја инструкција.

То је довело до идеја развоја микропроцесора код којих се инструкције извршавају са повећаном брзином.

### CISC сет инструкција

Првобитни приступ се називао CISC приступ у изградњи микропроцесора.

CISC (Complex Instruction Set Computer) – рачунари са комплексним сетом машинских инструкција.

У реализацији инструкција овог сета увек постоји макар једна инструкција којој је потребно много више времена да се реализује.

Први Intel микропроцесори су направљени са овим сетом инструкција (Intel x86).

Тада компајлери (програми који преводе програмски језик на машински) нису ни постојали па је било изузетно тешко програмирати на нивоу процесора.

Зато се током времена придодавало све већи број различитих инструкција чије извршавање је тражило све више времена (циклуса).

### RISC сет инструкција

Онда је настало RISC приступ у изградњи микропроцесора, IBM 70-тих година прошлог века.

RISC (Reduced Instruction Set Computer) – рачунари са смањеним сетом машинских инструкција.

Они користе велики број регистара у процесору, а већина инструкција се извршава током једног циклуса.

Смањен је број инструкција које се обрађују меморији што убрзава извршење самог програма.

Саме инструкције користе мали број операција са малим бројем различитих начина адресирања.

Формат инструкција је прост и фиксан (олакшава позивање инструкције из главне меморије).

Све ово је довело до лакшег писања компајлера за рад са процесорима и до наглог развоја програмских језика високог нивоа (PASCAL, FORTRAN, BASIC, C).