**Nama : Dian Puspitasari**

**NIM : L200130119**

**Kelas : A**

**TUGAS 02**

**Soal**

Daftar instruksi set untuk processor INTEL dan ARM (Android)

**Jawab**

Set Instruksi (Instruction Set Architecture (ISA)) didefinisikan sebagai suatu aspek dalam [arsitektur komputer](https://id.wikipedia.org/wiki/Arsitektur_komputer) yang dapat dilihat oleh para pemrogram. Secara umum, ISA ini mencakup jenis [data](https://id.wikipedia.org/wiki/Data) yang didukung, jenis [instruksi](https://id.wikipedia.org/w/index.php?title=Instruksi&action=edit&redlink=1) yang dipakai, jenis [register](https://id.wikipedia.org/wiki/Register), [mode pengalamatan](https://id.wikipedia.org/w/index.php?title=Mode_pengalamatan&action=edit&redlink=1), [arsitektur memori](https://id.wikipedia.org/w/index.php?title=Arsitektur_memori&action=edit&redlink=1), penanganan [interupsi](https://id.wikipedia.org/wiki/Interupsi), [eksepsi](https://id.wikipedia.org/w/index.php?title=Exception&action=edit&redlink=1), dan operasi I/O eksternalnya (jika ada).

ISA merupakan sebuah spesifikasi dari Pullman semua kode-kode [biner](https://id.wikipedia.org/wiki/Biner) (*opcode*) yang diimplementasikan dalam bentuk aslinya (*native form*) dalam sebuah desain [prosesor](https://id.wikipedia.org/wiki/Mikroprosesor) tertentu. Kumpulan *opcode* tersebut disebut sebagai bahasa mesin (*machine language*) untuk ISA yang bersangkutan. ISA yang populer digunakan adalah set instruksi untuk chip [Intel x86](https://id.wikipedia.org/wiki/X86), [IA-64](https://id.wikipedia.org/w/index.php?title=IA-64&action=edit&redlink=1), [IBM PowerPC](https://id.wikipedia.org/w/index.php?title=IBM_PowerPC&action=edit&redlink=1), [Motorola 68000](https://id.wikipedia.org/w/index.php?title=Motorola_68000&action=edit&redlink=1), [Sun SPARC](https://id.wikipedia.org/w/index.php?title=Sun_SPARC&action=edit&redlink=1), [DEC Alpha](https://id.wikipedia.org/wiki/Alpha_AXP), dan lain-lain.

ISA yang diimplementasikan dalam bentuk perangkat keras salah satunya adalah ARM dan INTEL

**1.** **ARM** adalah 32-bit Reduced Instruction Set Computer (RISC) set instruksi arsitektur (ISA) yang dikembangkan oleh ARM Holdings.

* Set instruksi ARM memiliki sejumlah instruksi untuk operand aritmatika dan logika pada operand yang berada dalam general-purpose register atau dinyatakan sebagai operand immediate dalam instruksi itu sendiri. Terdapat instruksi untuk operand logika AND,OR,NOT,XOR, dan bit-clear. Instruksi seperti compare disediakan untuk men-set condition code flag berdasarkan hasil dari operasi aritmatika dan logika pada dua operand

1. Instruksi Aritmatika

Ekspresi bahasa assembly dasar untuk instruksi aritmatika adalah

Opcode Rd, Rn, Rm

Dimana operasi yang ditetapkan oleh OP code dilakukan menggunakan operand dalam general-purpose register Rn dan Rm. Hasilnya diletakkan dalam register Rd. Misalnya, instruksi

ADD R0, R2, R4

Menjalankan operasi

R0 🡨 [R2] + [R4]

Dan instruksi

SUB R0, R6, R5

Menjalankan operasi

R0 🡨 [R6] – [R5]

1. Instruksi Logika

Operasi logika AND, OR, XOR, dan Bit-clear diimplementasikan oleh instruksi OP code AND, ORR, EOR, dan BIC. Kode tersebut memiliki format yang sama dengan instruksi aritmatika. Instruksi

AND Rd, Rn, Rm

Menjalankan operasi

Rd 🡨 [Rn] ^ [Rm]

Yang merupakan bitwise logical AND antara operand dalam register Rn dan Rm. Misalnya, jika register R0 berisi pola hexadesimal 02FA62CA dan R1 berisi pola 0000FFFF, maka instruksi

AND R0, RO, R1

Akan menyebabkan pola 000062CA diletakkan dalam register R0.

1. Instruksi Branch

Instruksi branch CONDITIONAL berisi offset 24-bit, 2’-complement, bertanda yang ditambahkan ke isi ter-update Program Counter untuk menghasilkan alamat target branch.

Instruksi Branch dieksekusi dengan cara yang sama seperti instruksi ARM yang lain, yaitu dieksekusi hanya jika keadaan terbaru condition code flag berhubungan dengan kondisi ditetapkan dalam field codition instruksi tersebut.

* Inti ARM

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Keluarga** | **Versi Arsitektur** | **Inti** | **Fitur** | **Cache (I/D)/Memory management unit** | **MIPS@MHz** | **Aplikasi** |
| **ARM1** | ARMv1 (kuno) | ARM1 |  | Tidak ada |  | ARM Evaluation System second processor for BBC Micro |
| **ARM2** | ARMv2 (kuno) | ARM2 | Penambahan instruksi multiply/perkalian (MUL) | Tidak ada | 4 MIPS @ 8 MHz 0.33DMIPS/MHz | Acorn Archimedes, Chessmachine |
| ARMv2a (kuno) | ARM250 | MEMC (MMU), prosesor grafis dan IO terintegrasi. Menambahkan instruksi SWP and SWPB (swap). | Tidak ada, MEMC1a | 7 MIPS @ 12 MHz | Acorn Archimedes |
| **ARM3** | ARMv2a (kuno) | ARM2a | Pertama kali penggunaan cache. | 4 kB unified | 12 MIPS @ 25 MHz 0.50 DMIPS/MHz | Acorn Archimedes |
| **ARM6** | ARMv3 (obsolete) | ARM60 | Pertama kali mendukung pengalamatan 32-bit. | Tidak ada | 10 MIPS @ 12 MHz | 3DO Interactive Multiplayer, Zarlink GPS Receiver |
| ARM600 | Penambahan cache dan coprocessor bus (FPA10 floating-point unit). | 4 kB unified | 28 MIPS @ 33 MHz |  |
| ARM610 | Tidak ada coprocessor bus. | 4 kB unified | 17 MIPS @ 20 MHz 0.65 DMIPS/MHz | Acorn Risc PC 600, Apple Newton 100 |
| [**ARM7**](https://id.wikipedia.org/w/index.php?title=ARM7&action=edit&redlink=1) | ARMv3 (kuno) | ARM700 |  | 8 kB unified | 40 MHz | Acorn Risc PC prototype CPU card |
| ARM710 | ARM700 | 8 kB unified | 40 MHz | Acorn Risc PC 700 |
| ARM710a | ARM700 | 8 kB unified | 40 MHz 0.68 DMIPS/MHz | Acorn Risc PC 700, Apple eMate 300 |
| ARM7100 | ARM710a, System-on-a-chip. | 8 kB unified | 18 MHz | Psion Series 5 |
| ARM7500 | ARM710a, SoC. | 4 kB unified | 40 MHz | Acorn A7000 |
| ARM7500FE | ARM7500, penambahan FPA dan EDO memory controller. | 4 kB unified | 56 MHz 0.73 DMIPS/MHz | Acorn A7000+ |
| [**ARM7TDMI**](https://id.wikipedia.org/w/index.php?title=ARM7TDMI&action=edit&redlink=1) | ARMv4T | ARM7TDMI(-S) | 3-stage pipeline, Thumb | Tidak ada | 15 MIPS @ 16.8 MHz 63 DMIPS @ 70 MHz | Game Boy Advance, Nintendo DS, iPod, Lego NXT, Atmel AT91SAM7, Juice Box, NXP Semiconductors LPC2000 dan LH754xx |
| ARM710T | ARM7TDMI, cache | 8 kB unified, MMU | 36 MIPS @ 40 MHz | Psion Series 5mx, Psion Revo/Revo Plus/Diamond Mako |
| ARM720T | ARM7TDMI, cache | 8 kB unified, MMU dengan Fast Context Switch Extension | 60 MIPS @ 59.8 MHz | Zipit Wireless Messenger, NXP Semiconductors LH7952x |
| ARM740T | ARM7TDMI, cache | MPU |  |  |
| ARMv5TEJ | ARM7EJ-S | 5-stage pipeline, Thumb, Jazelle DBX, Enhanced DSP | Tidak ada |  |  |
| [**StrongARM**](https://id.wikipedia.org/w/index.php?title=StrongARM&action=edit&redlink=1) | ARMv4 | SA-110 |  | 16 kB/16 kB, MMU | 203 MHz 1.0 DMIPS/MHz | Acorn Risc PC, Rebel/Corel Netwinder, Chalice CATS, Psion Netbook |
| SA-1110 | SA-110, SoC | 16 kB/16 kB, MMU | 233 MHz | LART (computer), Intel Assabet, Ipaq H36x0, Balloon Board Balloon2, Zaurus SL-5x00, HP Jornada 7xx, Jornada 560 series, Palm Zire 31 |
| **ARM8** | ARMv4 | ARM810 | 5-stage pipeline, static branch prediction, double-bandwidth memory | 8 kB unified, MMU | 84 MIPS @ 72 MHz 1.16 DMIPS/MHz | Acorn Risc PC prototype CPU card |
| **ARM9TDMI** | ARMv4T | ARM9TDMI | 5-stage pipeline, Thumb | Tidak ada |  |  |
| ARM920T | ARM9TDMI, cache | 16 kB/16 kB, MMU | 200 MIPS @ 180 MHz | Armadillo, Atmel AT91SAM9, GP32, GP2X, Tapwave Zodiac (Motorola i. MX1), HHP-49/50 Calculators, [Sun SPOT](https://id.wikipedia.org/w/index.php?title=Sun_SPOT&action=edit&redlink=1), Cirrus Logic EP9302, EP9307, EP9312, EP9315, Samsung S3C2442 (HTC TyTN, First International Computer Neo FreeRunner) |
| ARM922T | ARM9TDMI, caches | 8 kB/8 kB, MMU |  | NXP Semiconductors LH7A40x |
| ARM940T | As ARM9TDMI, caches | 4 kB/4 kB, MPU |  | GP2X (second core), Meizu M6 Mini Player |
| [**ARM9E**](https://id.wikipedia.org/w/index.php?title=ARM9E&action=edit&redlink=1) | ARMv5TE | ARM946E-S | Thumb, Enhanced DSP, cache | variable, tightly coupled memories, MPU |  | Nintendo DS, Nokia N-Gage, Canon PowerShot A470, Conexant 802.11 chips, Samsung S5L2010 |
| ARM966E-S | Thumb, Enhanced DSP instructions | no cache, TCMs |  | [ST Micro](https://id.wikipedia.org/w/index.php?title=ST_Microelectronics&action=edit&redlink=1) STR91xF, includes Ethernet |
| ARM968E-S | As ARM966E-S | no cache, TCMs |  | [NXP Semiconductors](https://id.wikipedia.org/wiki/NXP_Semiconductors) [LPC2900](http://www.standardics.nxp.com/products/lpc2000/lpc29xx/) |
| ARMv5TEJ | ARM926EJ-S | Thumb, Jazelle DBX, Enhanced DSP instructions | variable, TCMs, MMU | 220 MIPS @ 200 MHz, | Mobile phones: [Sony Ericsson](https://id.wikipedia.org/wiki/Sony_Ericsson) (K, W series); [Siemens](https://id.wikipedia.org/wiki/Siemens_AG) and [Benq](https://id.wikipedia.org/w/index.php?title=Benq&action=edit&redlink=1) (x65 series and newer); Texas Instruments [OMAP1710](https://id.wikipedia.org/w/index.php?title=OMAP&action=edit&redlink=1), [OMAP1610](https://id.wikipedia.org/w/index.php?title=OMAP&action=edit&redlink=1), [OMAP1611](https://id.wikipedia.org/w/index.php?title=OMAP&action=edit&redlink=1), [OMAP1612](https://id.wikipedia.org/w/index.php?title=OMAP&action=edit&redlink=1), [OMAP-L137](https://id.wikipedia.org/w/index.php?title=OMAP&action=edit&redlink=1), [OMAP-L138](https://id.wikipedia.org/w/index.php?title=OMAP&action=edit&redlink=1); [Qualcomm](https://id.wikipedia.org/wiki/Qualcomm) MSM6100, MSM6125, MSM6225, MSM6245, MSM6250, MSM6255A, MSM6260, MSM6275, MSM6280, MSM6300, MSM6500, MSM6800; [Freescale](https://id.wikipedia.org/wiki/Freescale) [i.MX21](https://id.wikipedia.org/w/index.php?title=I.MX21&action=edit&redlink=1), i.MX27, [Atmel](https://id.wikipedia.org/wiki/Atmel) [AT91SAM](https://id.wikipedia.org/w/index.php?title=AT91SAM&action=edit&redlink=1)9, [NXP Semiconductors](https://id.wikipedia.org/wiki/NXP_Semiconductors) [LPC3000](http://www.standardics.nxp.com/products/lpc3000/), GPH Wiz, [Marvell](https://id.wikipedia.org/wiki/Marvell_Technology_Group) [Feroceon](https://id.wikipedia.org/w/index.php?title=Feroceon&action=edit&redlink=1) (ex.: [SheevaPlug](https://id.wikipedia.org/w/index.php?title=SheevaPlug&action=edit&redlink=1)), NEC C10046F5-211-PN2-A SoC - undocumented core in the [ATi Hollywood](https://id.wikipedia.org/w/index.php?title=Hollywood_%28graphics_chip%29&action=edit&redlink=1) graphics chip used in the Wii, [Samsung](https://id.wikipedia.org/wiki/Samsung) S3C2412 used in [Squeezebox Duet](https://id.wikipedia.org/w/index.php?title=Squeezebox_%28network_music_player%29&action=edit&redlink=1)'s Controller. [NeoMagic](https://id.wikipedia.org/w/index.php?title=NeoMagic&action=edit&redlink=1) MiMagic Family MM6, MM6+, MM8, MTV; [Buffalo](https://id.wikipedia.org/w/index.php?title=Buffalo_network-attached_storage_series&action=edit&redlink=1) TeraStation Live ([NAS](https://id.wikipedia.org/w/index.php?title=Network-attached_storage&action=edit&redlink=1)); Telechips TCC7801, TCC7901; |
| ARMv5TE | ARM996HS | Clockless processor, as ARM966E-S | no caches, TCMs, MPU |  |  |
| **ARM10E** | ARMv5TE | ARM1020E | 6-stage pipeline, Thumb, Enhanced DSP instructions, (VFP) | 32 KB/32 KB, MMU |  |  |
| ARM1022E | As ARM1020E | 16 KB/16 KB, MMU |  |  |
| ARMv5TEJ | ARM1026EJ-S | Thumb, Jazelle DBX, Enhanced DSP instructions, (VFP) | variable, MMU or MPU |  | Western Digital [MyBook](https://id.wikipedia.org/w/index.php?title=MyBook&action=edit&redlink=1) II World Edition |
| [**XScale**](https://id.wikipedia.org/w/index.php?title=XScale&action=edit&redlink=1) | ARMv5TE | 80200/IOP310/IOP315 | I/O Processor, Thumb, Enhanced DSP instructions |  |  |  |
| 80219 |  |  | 400/600 MHz | [Thecus](https://id.wikipedia.org/w/index.php?title=Thecus&action=edit&redlink=1) N2100 |
| IOP321 |  |  | 600 [BogoMips](https://id.wikipedia.org/w/index.php?title=BogoMips&action=edit&redlink=1) @ 600 MHz | [Iyonix](https://id.wikipedia.org/w/index.php?title=Iyonix&action=edit&redlink=1) |
| IOP33x |  |  |  |  |
| IOP34x | 1-2 core, RAID Acceleration | 32K/32K L1, 512K L2, MMU |  |  |
| PXA210/PXA250 | Applications processor, 7-stage pipeline |  | PXA210: 133 and 200 MHz, PXA250: 200, 300, and 400 MHz | [Zaurus](https://id.wikipedia.org/w/index.php?title=Zaurus&action=edit&redlink=1) SL-5600, [iPAQ](https://id.wikipedia.org/w/index.php?title=IPAQ&action=edit&redlink=1) H3900, [Sony](https://id.wikipedia.org/wiki/Sony) [CLIÉ](https://id.wikipedia.org/w/index.php?title=CLI%C3%89&action=edit&redlink=1) NX60, NX70V, NZ90 |
| PXA255 |  | 32KB/32KB, MMU | 400 BogoMips @ 400 MHz; 371-533 MIPS @ 400 MHz | [Gumstix basix & connex](https://id.wikipedia.org/w/index.php?title=Gumstix&action=edit&redlink=1), [Palm Tungsten](https://id.wikipedia.org/w/index.php?title=Palm_Tungsten&action=edit&redlink=1) E2, [Zaurus](https://id.wikipedia.org/w/index.php?title=Zaurus&action=edit&redlink=1) SL-C860, [Mentor Ranger & Stryder](https://id.wikipedia.org/w/index.php?title=Mentor_Ranger_%26_Stryder&action=edit&redlink=1), iRex [ILiad](https://id.wikipedia.org/w/index.php?title=ILiad&action=edit&redlink=1) |
| PXA263 |  |  | 200, 300 and 400 MHz | [Sony](https://id.wikipedia.org/wiki/Sony) [CLIÉ](https://id.wikipedia.org/w/index.php?title=CLI%C3%89&action=edit&redlink=1) NX73V, NX80V |
| PXA26x |  |  | default 400 MHz, up to 624 MHz | [Palm Tungsten T3](https://id.wikipedia.org/w/index.php?title=Palm_Tungsten&action=edit&redlink=1) |
| PXA27x | Applications processor | 32 KB/32 KB, MMU | 800 MIPS @ 624 MHz | [Gumstix verdex](https://id.wikipedia.org/w/index.php?title=Gumstix&action=edit&redlink=1), PXA270 COM, [HTC](https://id.wikipedia.org/wiki/High_Tech_Computer_Corporation) Universal, [HP](https://id.wikipedia.org/wiki/Hewlett-Packard) hx4700, [Zaurus](https://id.wikipedia.org/w/index.php?title=Zaurus&action=edit&redlink=1) SL-C1000, 3000, 3100, 3200, [Dell Axim](https://id.wikipedia.org/w/index.php?title=Dell_Axim&action=edit&redlink=1) x30, x50, and x51 series, Motorola Q, [Balloon3](https://id.wikipedia.org/w/index.php?title=Balloon_Board&action=edit&redlink=1), [Trolltech Greenphone](https://id.wikipedia.org/w/index.php?title=Greenphone&action=edit&redlink=1), [Palm TX](https://id.wikipedia.org/w/index.php?title=Palm_TX&action=edit&redlink=1), Motorola Ezx Platform A728, A780, A910, A1200, E680, E680i, E680g, E690, E895, Rokr E2, Rokr E6, Fujitsu Siemens LOOX N560, Toshiba Portégé G500, Trēo 650-755p, [Zipit Z2](https://id.wikipedia.org/w/index.php?title=Zipit_Wireless_Messenger&action=edit&redlink=1) |
| PXA800(E)F |  |  |  |  |
| Monahans |  | 32KB/32KB L1, TCM, MMU | 1000 MIPS @ 1.25 GHz | Samsung Omnia |
| PXA900 |  |  |  | Blackberry 8700, Blackberry Pearl (8100) |
| IXC1100 | Control Plane Processor |  |  |  |
| IXP2400/IXP2800 |  |  |  |  |
| IXP2850 |  |  |  |  |
| IXP2325/IXP2350 |  |  |  |  |
| IXP42x |  |  |  | [NSLU2](https://id.wikipedia.org/w/index.php?title=NSLU2&action=edit&redlink=1) IXP460/IXP465 |
| **ARM11** | ARMv6 | ARM1136J(F)-S | 8-stage pipeline, [SIMD](https://id.wikipedia.org/wiki/SIMD), Thumb, Jazelle DBX, (VFP), Enhanced DSP instructions | variable, MMU | 740 @ 532-665 MHz (i.MX31 SoC), 400-528 MHz | Texas Instruments [OMAP2420](https://id.wikipedia.org/w/index.php?title=OMAP&action=edit&redlink=1) ([Nokia E90](https://id.wikipedia.org/wiki/Nokia_E90), [Nokia N93](https://id.wikipedia.org/w/index.php?title=Nokia_N93&action=edit&redlink=1), [Nokia N95](https://id.wikipedia.org/wiki/Nokia_N95), [Nokia N82](https://id.wikipedia.org/wiki/Nokia_N82)), [Zune](https://id.wikipedia.org/wiki/Zune), [BUGbase](https://id.wikipedia.org/w/index.php?title=BUGbase&action=edit&redlink=1)[[1]](http://www.buglabs.net/), [Nokia N800](https://id.wikipedia.org/w/index.php?title=Nokia_N800&action=edit&redlink=1), [Nokia N810](https://id.wikipedia.org/w/index.php?title=Nokia_N810&action=edit&redlink=1), [Qualcomm](https://id.wikipedia.org/wiki/Qualcomm) MSM7200 (with integrated ARM926EJ-S Coprocessor@274 MHz, used in [Eten Glofiish](https://id.wikipedia.org/w/index.php?title=Eten_Glofiish&action=edit&redlink=1), [HTC TyTN II](https://id.wikipedia.org/w/index.php?title=HTC_TyTN_II&action=edit&redlink=1), [HTC Nike](https://id.wikipedia.org/w/index.php?title=HTC_Nike&action=edit&redlink=1)), Freescale [i.MX31](https://id.wikipedia.org/w/index.php?title=I.MX31&action=edit&redlink=1) (used in the original Zune 30gb and Toshiba Gigabeat S), Freescale MXC300-30 ([Nokia E63](https://id.wikipedia.org/wiki/Nokia_E63), [Nokia E71](https://id.wikipedia.org/wiki/Nokia_E71), [Nokia 5800](https://id.wikipedia.org/wiki/Nokia_5800), [Nokia E51](https://id.wikipedia.org/wiki/Nokia_E51), [Nokia E75](https://id.wikipedia.org/wiki/Nokia_E75), [Nokia N97](https://id.wikipedia.org/wiki/Nokia_N97), [Nokia N81](https://id.wikipedia.org/wiki/Nokia_N81)), Qualcomm MSM7201A as seen in the [HTC Dream](https://id.wikipedia.org/wiki/HTC_Dream), [HTC Magic](https://id.wikipedia.org/w/index.php?title=HTC_Magic&action=edit&redlink=1), [Motorola Z6](https://id.wikipedia.org/w/index.php?title=Motorola_Z6&action=edit&redlink=1), [HTC Hero](https://id.wikipedia.org/w/index.php?title=HTC_Hero&action=edit&redlink=1) |
| ARMv6T2 | ARM1156T2(F)-S | 9-stage pipeline, [SIMD](https://id.wikipedia.org/wiki/SIMD), Thumb-2, (VFP), Enhanced DSP instructions | variable, MPU |  |  |
| ARMv6KZ | ARM1176JZ(F)-S | As ARM1136EJ(F)-S | variable, MMU+TrustZone |  | [Apple iPhone](https://id.wikipedia.org/wiki/IPhone), [Apple iPod touch](https://id.wikipedia.org/wiki/IPod_touch), [Conexant CX2427X](https://id.wikipedia.org/w/index.php?title=Conexant&action=edit&redlink=1), [Motorola RIZR Z8](https://id.wikipedia.org/w/index.php?title=Motorola_RIZR_Z8&action=edit&redlink=1), [Motorola RIZR Z10](https://id.wikipedia.org/w/index.php?title=Motorola_Motorizr_Z10&action=edit&redlink=1), [NVIDIA GoForce 6100](https://id.wikipedia.org/w/index.php?title=NVIDIA_GoForce_6100&action=edit&redlink=1); Telechips TCC9101, TCC9201, TCC8900, [Fujitsu MB86H60](https://id.wikipedia.org/w/index.php?title=Fujitsu_MB86H60&action=edit&redlink=1), Samsung S3C6410, S3C6430 |
| ARMv6K | ARM11 MPCore | As ARM1136EJ(F)-S, 1-4 core SMP | variable, MMU |  | [Nvidia APX 2500](https://id.wikipedia.org/w/index.php?title=Nvidia_APX_2500&action=edit&redlink=1) |
| **Cortex** | ARMv7-A | Cortex-A8 | Application profile, VFP, NEON, Jazelle RCT, Thumb-2, 13-stage superscalar pipeline | variable (L1+L2), MMU+TrustZone | up to 2000 (2.0 DMIPS/MHz in speed from 600 MHz to greater than 1 GHz) | Texas Instruments [OMAP3xxx series](https://id.wikipedia.org/w/index.php?title=OMAP&action=edit&redlink=1), [SBM7000](https://id.wikipedia.org/w/index.php?title=SBM&action=edit&redlink=1), Oregon State University [OSWALD](https://id.wikipedia.org/w/index.php?title=OSWALD&action=edit&redlink=1), [Gumstix Overo Earth](https://id.wikipedia.org/w/index.php?title=Gumstix&action=edit&redlink=1), [Pandora](https://id.wikipedia.org/w/index.php?title=Pandora_%28console%29&action=edit&redlink=1), [Apple iPod touch (3rd Generation)](https://id.wikipedia.org/wiki/IPod_touch), [Archos 5](https://id.wikipedia.org/w/index.php?title=Archos_5&action=edit&redlink=1), [FreeScale](https://id.wikipedia.org/wiki/Freescale_Semiconductor) i.MX51-[SOC](https://id.wikipedia.org/w/index.php?title=System-on-a-chip&action=edit&redlink=1), [BeagleBoard](https://id.wikipedia.org/w/index.php?title=BeagleBoard&action=edit&redlink=1), [Apple iPhone 3GS](https://id.wikipedia.org/w/index.php?title=Apple_iPhone_3GS&action=edit&redlink=1), [Palm Pre](https://id.wikipedia.org/w/index.php?title=Palm_Pre&action=edit&redlink=1), [Samsung i8910](https://id.wikipedia.org/w/index.php?title=Samsung_i8910&action=edit&redlink=1), [Sony Ericsson Satio](https://id.wikipedia.org/w/index.php?title=Sony_Ericsson_Satio&action=edit&redlink=1), [Touch Book](https://id.wikipedia.org/w/index.php?title=Touch_Book&action=edit&redlink=1), [Nokia N900](https://id.wikipedia.org/w/index.php?title=Nokia_N900&action=edit&redlink=1). |
| Cortex-A9 | Application profile, (VFP), (NEON), Jazelle RCT and DBX, Thumb-2, Out-of-order speculative issue superscalar | MMU+TrustZone | 2.0 DMIPS/MHz |  |
| [Cortex-A9 MPCore](https://id.wikipedia.org/w/index.php?title=ARM_Cortex-A9_MPCore&action=edit&redlink=1) | As Cortex-A9, 1-4 core SMP | MMU+TrustZone | 2.0 DMIPS/MHz (per core) | Texas Instruments [OMAP4430/4440](https://id.wikipedia.org/w/index.php?title=OMAP&action=edit&redlink=1) ; "Sparrow" (planned) |
| ARMv7-R | Cortex-R4(F) | Embedded profile, Thumb-2, (FPU) | variable cache, MPU optional | 600 [DMIPS](https://id.wikipedia.org/w/index.php?title=DMIPS&action=edit&redlink=1) @ 475 MHz | [Broadcom](https://id.wikipedia.org/wiki/Broadcom) is a user, TMS570 from Texas Instruments |
| ARMv7-M | Cortex-M3 | Microcontroller profile, Thumb-2 only. | no cache, MPU optional | 125 DMIPS @ 100 MHz | [Energy Micro](https://id.wikipedia.org/wiki/Energy_Micro)'s [EFM32](https://id.wikipedia.org/w/index.php?title=EFM32&action=edit&redlink=1), [Texas Instruments](https://id.wikipedia.org/wiki/Texas_Instruments) [Stellaris](http://www.luminarymicro.com/) microcontroller family, [ST Microelectronics](https://id.wikipedia.org/w/index.php?title=ST_Microelectronics&action=edit&redlink=1) [STM32](http://mcu.st.com/), [NXP Semiconductors](https://id.wikipedia.org/wiki/NXP_Semiconductors) [LPC1700](http://www.standardics.nxp.com/products/lpc1000/lpc17xx/), [Toshiba](https://id.wikipedia.org/wiki/Toshiba) [TMPM330FDFG](http://www.toshiba.com/taec/news/press_releases/2008/mcus_08_542.jsp), [Ember](https://id.wikipedia.org/w/index.php?title=Ember_%28company%29&action=edit&redlink=1)'s EM300 Series, [Atmel](https://id.wikipedia.org/wiki/Atmel) [AT91SAM](https://id.wikipedia.org/w/index.php?title=AT91SAM&action=edit&redlink=1)3 |
| ARMv6-M | Cortex-M0 (codenamed "Swift") | Microcontroller profile, Thumb-2 (16-bit Thumb instructions & BL, MRS, MSR, ISB, DSB, and DMB). | No cache. | 0.9 DMIPS/MHz | [NXP Semiconductors](https://id.wikipedia.org/wiki/NXP_Semiconductors) [NXP LPC1100](http://www.standardics.nxp.com/products/lpc1000/lpc11xx/) |
| Cortex-M1 | FPGA targeted, Microcontroller profile, Thumb-2 (16-bit Thumb instructions & BL, MRS, MSR, ISB, DSB, and DMB). | None, tightly coupled memory optional. | Up to 136 DMIPS @ 170 MHz |  |

**2. INTEL**

|  |  |
| --- | --- |
| Prosesor [Intel](https://id.wikipedia.org/wiki/Intel_Corporation) | |
|  | |
| Termasuk prosesor x86 | [8086](https://id.wikipedia.org/wiki/Intel_8086) • [8088](https://id.wikipedia.org/wiki/Intel_8088) • [80186](https://id.wikipedia.org/wiki/Intel_80186) • [80188](https://id.wikipedia.org/w/index.php?title=Intel_80188&action=edit&redlink=1) • [80286](https://id.wikipedia.org/wiki/Intel_80286) • [80386](https://id.wikipedia.org/wiki/Intel_80386) • [80486](https://id.wikipedia.org/wiki/Intel_80486) • [Pentium](https://id.wikipedia.org/wiki/Pentium) • [Pentium Pro](https://id.wikipedia.org/wiki/Pentium_Pro) • [Pentium II](https://id.wikipedia.org/wiki/Pentium_II) • [Celeron](https://id.wikipedia.org/wiki/Celeron) [Pentium III](https://id.wikipedia.org/wiki/Pentium_III) • [Pentium 4](https://id.wikipedia.org/wiki/Pentium_4) • [Pentium M](https://id.wikipedia.org/wiki/Pentium_M) • [Pentium D](https://id.wikipedia.org/wiki/Pentium_D) • [Pentium Extreme Edition](https://id.wikipedia.org/wiki/Pentium_Extreme_Edition) • [Xeon](https://id.wikipedia.org/wiki/Xeon) • [Core](https://id.wikipedia.org/w/index.php?title=Intel_Core&action=edit&redlink=1) • [Core 2](https://id.wikipedia.org/wiki/Intel_Core_2) |
|  | |
| Tidak termasuk prosesor x86 | [4004](https://id.wikipedia.org/wiki/Intel_4004) • [4040](https://id.wikipedia.org/wiki/Intel_4040) • [8008](https://id.wikipedia.org/wiki/Intel_8008) • [8080](https://id.wikipedia.org/wiki/Intel_8080) • [8085](https://id.wikipedia.org/wiki/Intel_8085) • [iAPX 432](https://id.wikipedia.org/w/index.php?title=Intel_iAPX_432&action=edit&redlink=1) • [i860](https://id.wikipedia.org/w/index.php?title=Intel_i860&action=edit&redlink=1) • [i960](https://id.wikipedia.org/w/index.php?title=Intel_i960&action=edit&redlink=1) • [XScale](https://id.wikipedia.org/w/index.php?title=Intel_XScale&action=edit&redlink=1) • [Itanium](https://id.wikipedia.org/wiki/Itanium) • [Itanium 2](https://id.wikipedia.org/w/index.php?title=Itanium_2&action=edit&redlink=1) |

* Set Instruksi Pada Microprocessor 8088

1. Transfer Data

Set instruksi 8088 mencakup 14 transfer data yang menggerakkan byte atau kata data antara memori dan register selain juga antara akumulator dan port I/O.

a.GeneralPurpose(Umum)                              :MOV,PUSH,POP,XCHG,SXLAT

b. Input/Output                                              : In, Out

c. Address Transfer (Pemindahan Alamat)      : LEA, KDS, KES

d. Flag Transfer (Pemindahan Flag)                : LAHF, SAHF, PUSHF, POPF

1. Arithmetic Instructions (Instruksi Perhitungan)

8088 mampu melakukan penambahan, pengurangan, pengurangan dan pembagian data selain juga bytes dan kata. Sistem ini menambah dan mengurangi dengan menggunakan bytes atau kata yang bertanda atau tak bertanda dan data BCD atau ASCII.

a.       Addition (Penjumlahan)                 : ADD, ADC, AAA, DAA, INC

b.      Substraction (Pengurangan)           : SUB, SBB, AAS, DAS, DES, NEG, CMP

c.       Multiplication (Perkalian)              : MUL, IMUL, AAM

d.      Division (Pembagian)                    : DIV, IDIV, AAD

e.       Sign Extension                              : CBW, CWD

1. Manipulasi Bit

Instruksi ini mencakup operasi logika, shift dan rotasi.

a.       Logical (Logika)    : AND, OR, XOR, NOT, TEST

b.      Shift (Geser)          : SAL/SHL, SAR, SHR

c.       Rotate (Gulung)     : ROL, ROR, RCL, RCR

1. Instruksi String

Instruksi string digunakan untuk memanipulasi string data dalam memori. Setiap string tersusun baik dari bytes maupun kata dan hingga mencapai 64 K bytes panjang.nya

a.       Direction                : CLD, STD

b.      Repeat Prefixes      : REP, REPE/REPZ, REPNE/REPNZ,

c.       Move                     : MOVSB, MOVSW

d.      Compare                : CMPS, CMPSB, CMPSW

e.       Scan                       : CSAS, SCASB, SCASW

f.       Load and Store      : LODS, LODSB, LODSW, STOS, STOSB, STOSW

1. Transfer Program

Instruksi transfer program mencakup jump,  call, dan return, instruksi yang sudah kita kenal dalam 8085.

a.       Unconditional transfer (Pemindahan)               : CAL, RET, JMP

b.      Conditional transfer (Pemindahan Bersyarat)   : JA/JNBE, JAE/JNB, JB/JNAE/JC,

JBE/JNA, JCXZ, JE/JZ, JG/JNLE, JGF/JNL, JL/JNGE, JLE/JNG, JNC, JNE/JNZ, JNO, JNP/JPO, JNS, JO, JP/JPE, JS

c.       Loop/Jump                                                    : LOOP, LOOPE/LOOPZ, LOOPNE/LOOPNZ

1. Kontrol Prosesor

Instruksi kontrol prosesor memungkinkan dan tidak memungkinkan interupsi, memodifikasi bit flag dan mensinkronisasi kejadian eksternal.

a.       Flag Operations                            : CLC, STC, CMC, CLD, STD, CLI, STI,

b.      External Synchronization Interrupt : INT, INTO, IRET

§  Berikut ini penjelasan tentang instruksi yang digunakan pada bahasa rakitan 8086 dan 8088 atau sejenisnya :

1.        **AAA** (ASCII adjust for edition) Pengaturan ASCII bagi penambahan

2.        **AAD** (ASCII adjust for division) Pengaturan ASCII bagi pembagian

3.        **AAM** (ASCII adjust for multipy) Pengaturan ASCII bagi perkalian

4.        **AAS** (ASCII adjust for substraction) Pengaturan ASCII bagi pengurangan

5.        **ADC** (add with carry) T ambahkan dengan carry

6.        **ADD** (addition) Penambahan

7.        **AND** (logic AND) Logik AND

8.        **CALL** (CALL subroutine) Subrutin panggil

9.      **CBW** (Convert byte to word) Konversikan byte ke kata

10.     **CLC** (Clear Carry) Kosongkan corry flag

11.     **CLD** (Clear directon flag) Kosongkan flag arah

12.     **CLI** (Clear interrupt enable) Kosongkan flag penggerak interupsi

13.     **DAS** (decimal adjust for substraction) Pengaturan decimal bagi pengurangan

14.     **DEC** (decrement) Penurunan operand tujuan dengan 1

15.     **DIV** (divide) Pembagian tak bertanda

16.     **ESC** (escape) Escape digunakan sehubungan dengan suatu co-procesor external

17.     **HLT** (halt) HLT menghentikan prosesor sampai saluran reset diaktifkan

18.     **IDIV** (Integer division) Pembagian kilat

19.     **IMUL** (Integer multiply) Perkalian kilat

20.     **IN** (Input) IN (masukan) mentransfer data dari port yang disperifikasikan ke dalam register AC atau AX

21.     **NC** (Increment) INC menaikkan operand tujuan dalam 1

22.     **INT** (Interupt) INT (Interupsi) mengawali suatu prosedur interupsi dengan jenis yang dispesifikasikan oleh instruksi yang bersangkutan

23.     **INTO** (Interupt on overflow) Instrupsi bila ada overflow INTO digunakan untuk membangkitkan suatu interupsi perangkat lunak yang bergantumg pada status flag OF

24.     **IRET** (Interupt return) IRET digunakan untuk kembali dari suatu interupsi dan mendapatkan kembali IP,CS dan flag-flag dari stack

25.     **JA** (Jump on above) Digunakan jika operand pertama lebih besar daripada operand kedua

26.     **JNBE** (Jump) Digunakan jika pada saat pembandingan operand 1, tidak lebih kecil atausama dengan operand 2

27.     **JAE** (Jump on above or equal) Digunakan pada saat pembandingan operand 1 lebih besar atau sama dengan operand 2

28.     **JNB** (Jump not below) Digunakan pada saat pembandingan operand 1 tidak lebih kecil dari operand 2

29.     **JB** (Jump on below) Digunakan pada saat operand 1 lebih kecil dari operand 2

30.     **JNAE** (Jump on not above or equal) Digunakan pada saat operand 1 tidak lebih besar atau sama dengan operand 2

31.     **JBE** (Jump on below or equal) Digunakan pada saat operand 1 lebih kecil atau sama dengan operand 2

32.     **JNA** (Jump on not above) Digunakan pada saat operand 1 tidak lebih besar dari operand 2

33.     **JC** (Jump on carry) Digunakan pada saat akan diprogram CF=1

34.     **JCXZ** (Jump if CX=0) Digunakan bila isi register CX=0

35.     **JE** (Jump on equal) Digunakan pada saat pembandingan kedua operand sama

36.     **JZ** (Jump on zero) Digunakan pada saat akan diproses ZF=1

37.     **JG** (Jump on greather than) Digunakan pada saat pembandingan operand 1 menunjukkan lebih besar dari operand 2

38.     **JNLE** (Jump on greather or equal) Digunakan pada saat operand 1tidak lebih kecil atau sama dengan operand 2

39.     **JGE** (Jump on greather or equal) Digunakan pada saat operand 1 lebih besar atau sama dengan operand 2

40.     **JNL** (Jump on less) Digunakan pada saat operand 1 tidak lebih kecil dari operand 2

41.     **JL** (Jump on less) Digunakan pada saat operand 1 lebih kecil dari operand 2

42.     **JNGE** (Jump on not greather or equal) Digunakan pada saat operand 1 tidak lebih besar atau sama dengan operand 2

43.     **JLE** (Jump on less or equal) Digunakan pada saat operand 1 lebih kecil atau sama dengan operand 2

44.     **JNG** (Jump on not greather than) Digunakan pada saat operand 1 tidak lebih besar daripada operand 2

45.     **JMP** (Unconditional jump) Lompatan tidak bersyarat

46.     **JNC** (Jump on not carry) Digunakan pada saat diproses CF=O

47.     **JNE** (Jump on not equal) Digunakan pada saat operand 1 tidak sama dari operand 2

48.     **JNO** (Jump on not zero) Digunakan pada saat diproses ZF=0

49.     **JNO** (Jump on not overflow) Digunakan bila tidak ada overflow (OF=O)

50.     **JNS** (Jump on not sign) Digunakan pada saat diproses SF=O

51.     **ZNP** (Jump on not pority) Digunakan bila tidak ada pointer

52.     **JPO** (Jump on poity odd) Digunakan bila pointer ganjil

53.     **O** (Jump on overflow) Digunakan pada saat ada overflow

54.     **JP** (Jump on Pority equal) Digunakan bila pointer genap

55.     **JS** (Jump on sign) Digunakan pada saat diproses ZF=1

56.     **LAHF** (Load AH from flag) LAHF me-load bit 7,6,4,2 & 0 pada register AH masing-masing dengan isi flag SF.ZF,AF & CF

57.     **LDS** (Load pointer using DS) Muatkan peminjak dengan menggunakan DS

58.     **LEA** (Load effective address) LEA mentransfer opsan sumber 16 bit dalam memori ke tujuan 16 bit

59.     **LES** (Load pointer using ES) Muatkan pointer dengan menggunakan ES

60.     **LOCK** (Lock bas) Digunakan dalam penerapan pemakaian sumber bersama, untuk memastikan bahwa memori tidak diakses secara serentak oleh lebih dari satu proses

61.     **LODS** (Load string) Muatkan string byte atau kata

62.     **LOOP** Loop jika EX bukan 0

63.     **LOOPE** dan LOOPZ -Loop bila sama

64.     **-Loop** bila 0

65.     **LOOPNE/LOOPNZ** -Loop bila tidak sama (ZF=0)

66.     **-Loop** bila tidak nol (ZF=O)

67.     **MOVE** Pindahkan byte atau kata memindahkan data 8 bit atau 16 bit

68.     **MOUS** (Move string) Pindahkan data string 8 bit atau 16 bit

69.     **MUL** (Multifly) Digunakan untuk mengalihkan isi bertandapada akumulator dengan suatu operand sumber yang dispesifikasikan

70.     **NEG** (Negate) Mengurangkan suatu operand tujuan dari 0 & menyimpan hasil komplemen keduanya dalam tujuan

71.     **NOP** (Logic NOT) NOP tidak membe rikan dampak tertentu (tidak ada operasi)

72.     **NOT** (Logic OR) Operasi logic OR

73.     **OR** (Logic OR) Operasi logic OR

74.     **OUT** (Output) Digunakan untuk mentransfer data dari AL atau AX ke point (pangkalan)

75.     **POP** (POP from stack) Keluaran data dari stack dari AL atau AX ke point (pangkalan)

76.     **POPF** (POP flag) Keluarkan ke flag dan stack

77.     **POSH** (Push to stack) Dorong sumber ke stack

78.     **PUSHF** (Push flag) Dorong flag ke stack

79.     **RCL** (Rotate lift with carry) Putar ke kirin dengan carry 1

80.     **RCR** (Rotate right with carry) Putar ke kanan dengan carry 1

81.     **REP** Ulangi

82.     **REE** Ulangi jika sama

83.     **REPZ** Ulangi jika nol

84.     **REPNE** Ulangi jika tidak sama

85.     **REPNZ** Ulangi jika tidak nol

86.     **RET** (Return from subroutine) Digunakan untuk kembali ke subrutin

87.     **ROL** (Rotate left) Rotasi (putar) ke kiri 1

88.     **ROR** (Rotate right) Rotasi (putar) ke kanan 1

89.     **SAHF** (store AH in flags register) Simpan AH dalam reegister flag

90.     **SAL/SHL** (Shift left arithmetic/logical) Geser aritmatik/logika ke kiri 1

91.     **SAR/SHR** (Shift right arithmetic/logical) Geser aritmatik/logika ke kanan 1

92.     **SBB** (Substract with borrow) Kurangkan dengan borrow

93.     **SCAS** (Scan string) Digunakan untuk mengurangkan string tujuan yang dialamatkan a/n regiser DI,dari AX atau AL

94.     **STC** (Set carry flag) Menset carry flag (CF)

95.     **STD** (Set direction flag) Menset flag arah (DF)

96.     **STI** (Set interrupt enable flag) Menset flag penggerak interupsi (IF)

97.     **STOS** (Substract) Memindahkan operand sumber yang terkandung dalam AX atau AL ke suatu tujuan yang dialamatkan oleh DI

98.     **SUB** (Substract) Mengurangkan sumber dari operand tujuan dan hasilnya ditepatkan dalam tujuan

99.     **TEST** (Logical comporison) Menguji operand-operand atau logic perbandingan

100.  **WAIT** Menunggu sampai saluran test aktif

101.  **XCHG** (Exchange) Pertukaran dan sumber

102.  **XLAT** (Translate) Digunakan untuk menerjemahkan se-byte dalam register AL ke dalam suatu byte yang diambil dari tabel terjemahan

103.  **XOR** (Logic exclusive OR) Operasi exclusive OR