Microcontroller Based System Design

- This class is focused on the principles and practices of modern embedded systems design.
- In class, we will focus on computer architecture beyond the CPU, fundamentals of the hardware/software interface, techniques for sensing and controlling the physical world, and a few other topics.
- We will focus on the ARM Cortex-M3 (Acorn RISC (Reduced instruction set computing) Machine), FPGAs (Field Programmable Gate Array) and other supporting software, to learn how to build and program embedded systems.

What is driving the
embedded everywhere explosion?

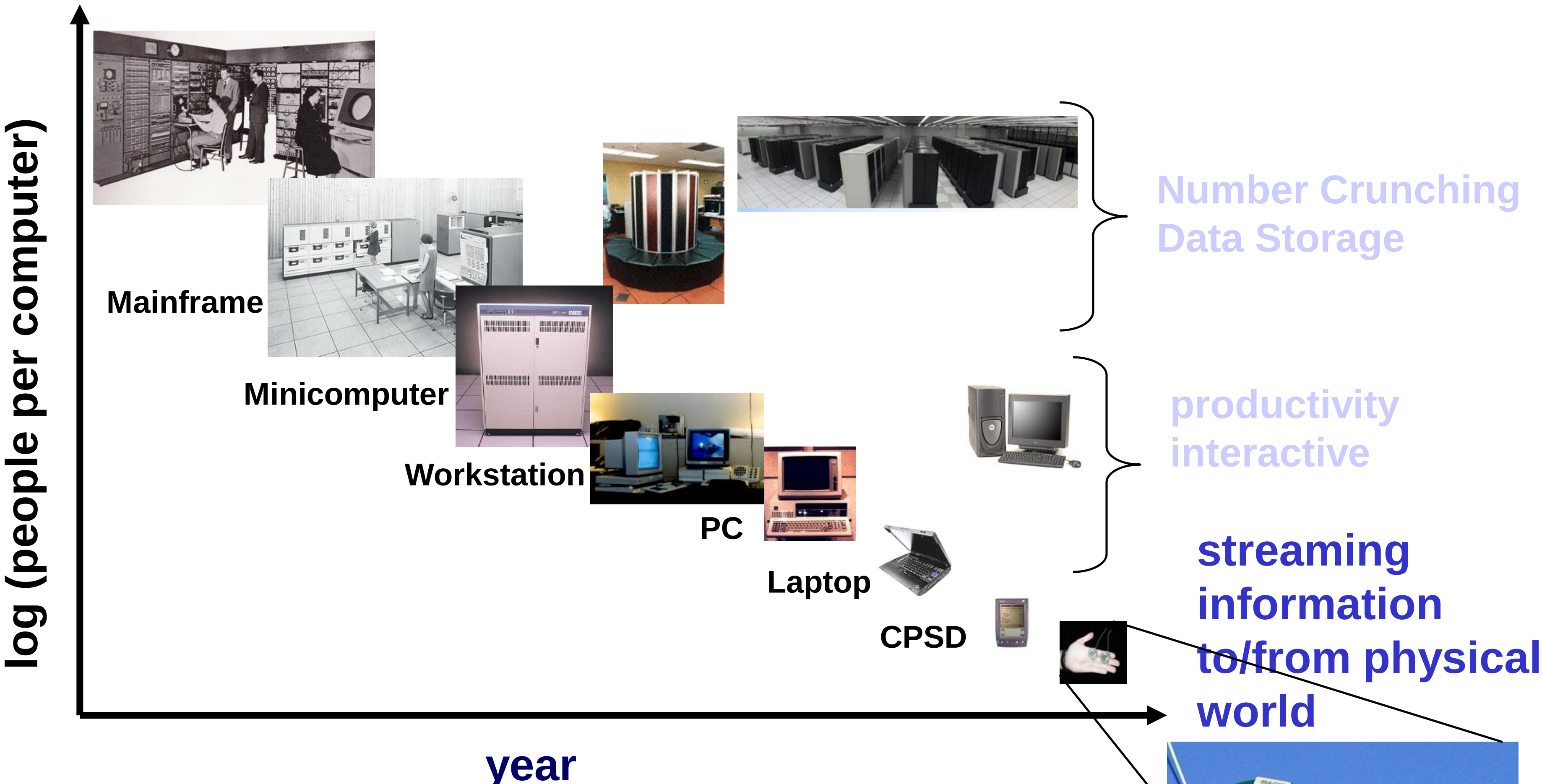
Embedded, everywhere

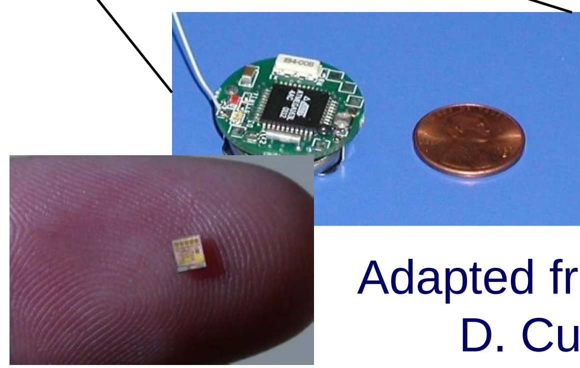
Outline

Technology Trends

Design Questions

Bell's Law of Computer Classes:
A new computing class roughly every decade

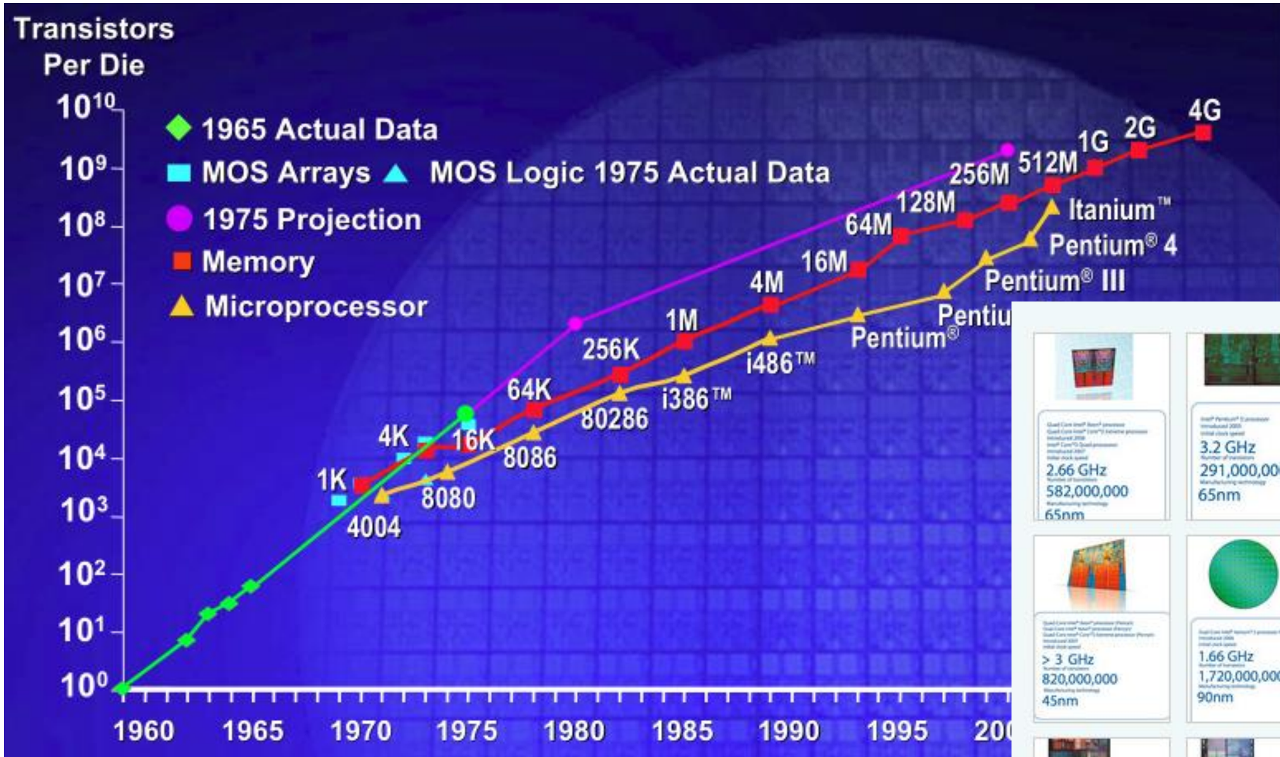




"Roughly every decade a new, lower priced computer class forms based on a new programming platform, network, and interface resulting in new usage and the establishment of a new industry."

Adapted from D. Culler

Moore's Law:
IC transistor count doubles every two years



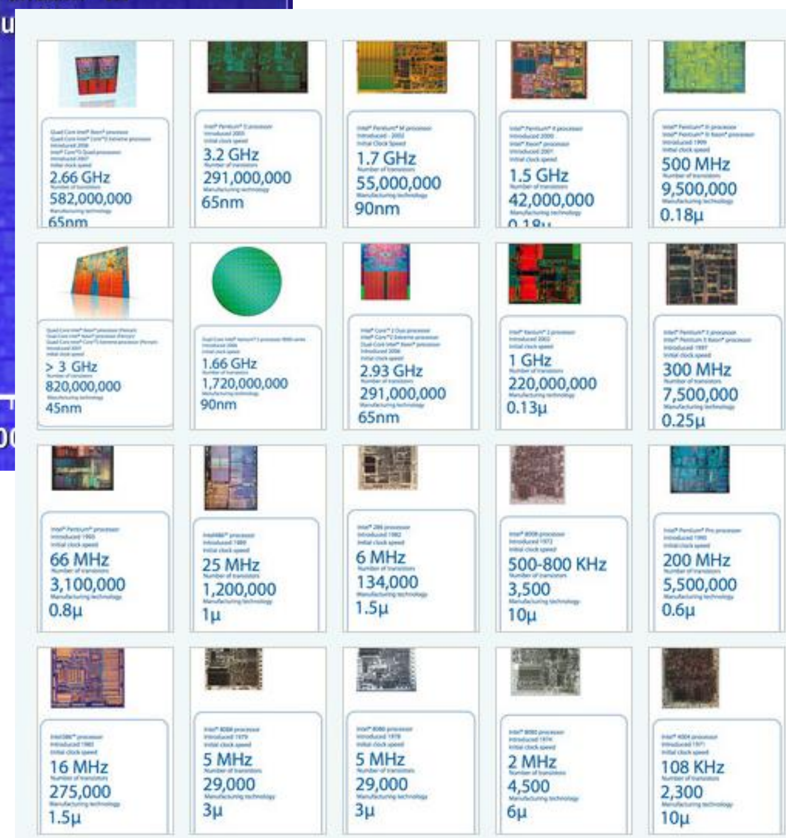


Photo Credit: Intel

Flash memory scaling:
Rise of density & volumes; Fall (and rise) of prices

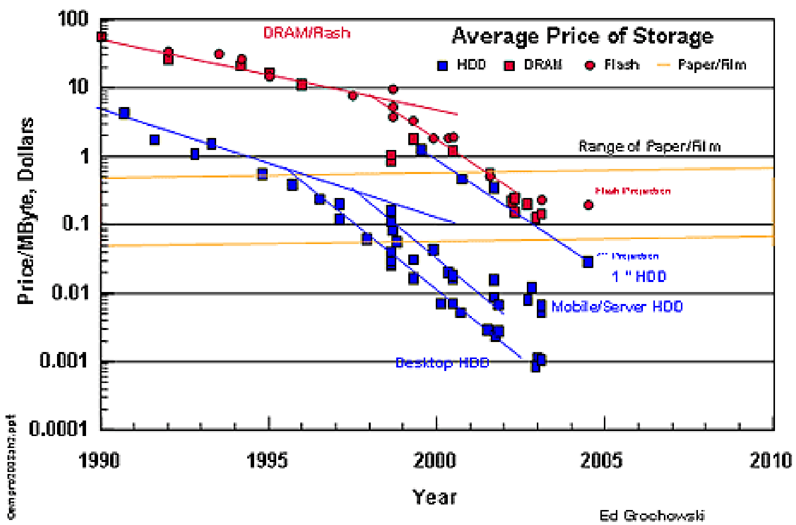
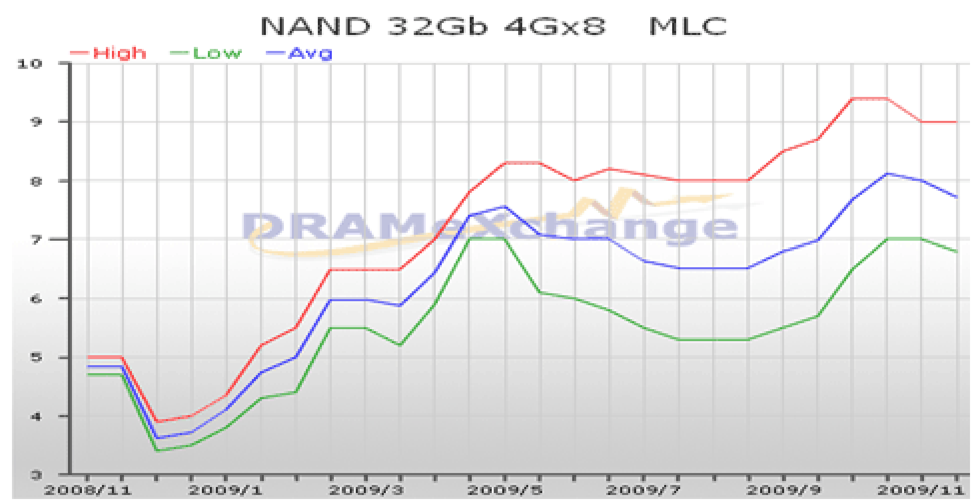
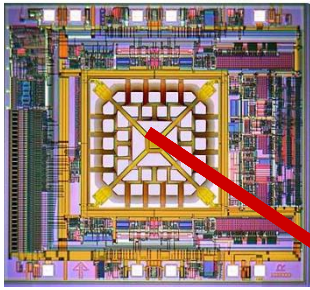


Figure-1 32Gb MLC NAND Flash contract price trend

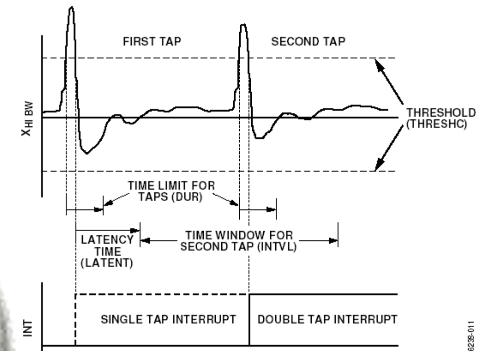


Source: DRAMeXchange, Nov. 2009.

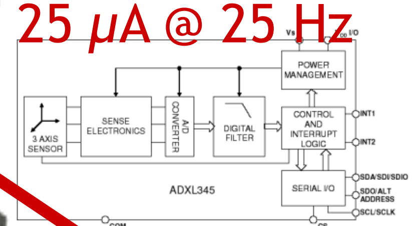
MEMS Accelerometers:
Rapidly falling price and power



O(mA)



25 μA @ 25 Hz

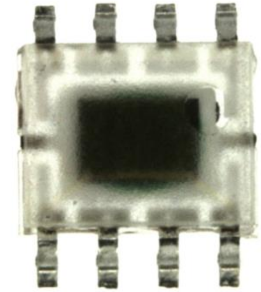


ADXL345
[Analog Devices, 2009]

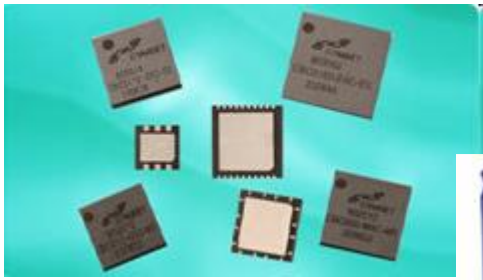
10 μA @ 10 Hz @ 6 bits
[ST Microelectronics, annc. 2009]

Energy harvesting and storage: Small doesn't mean powerless…

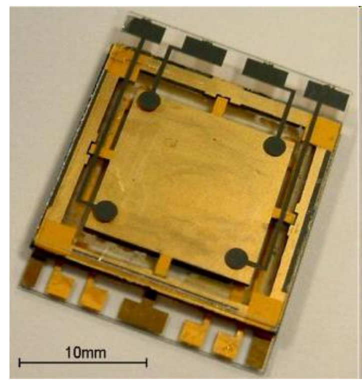
RF [Intel]



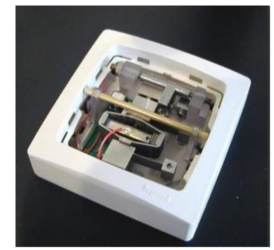
Clare Solar Cell



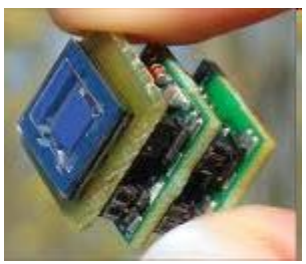
Thin-film batteries



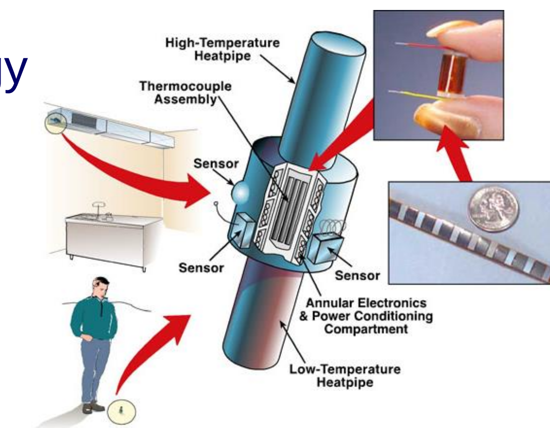
Electrostatic Energy Harvester [ICL]

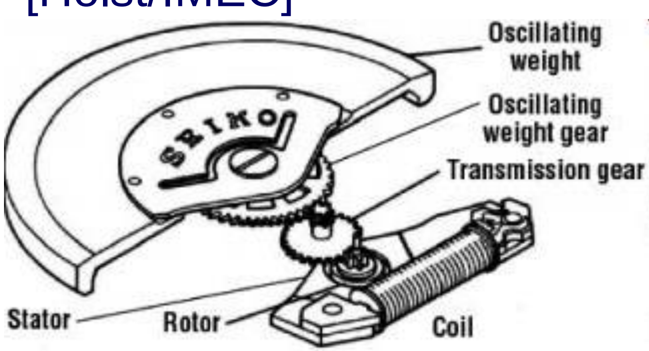


Shock Energy Harvesting CEDRAT Technologies



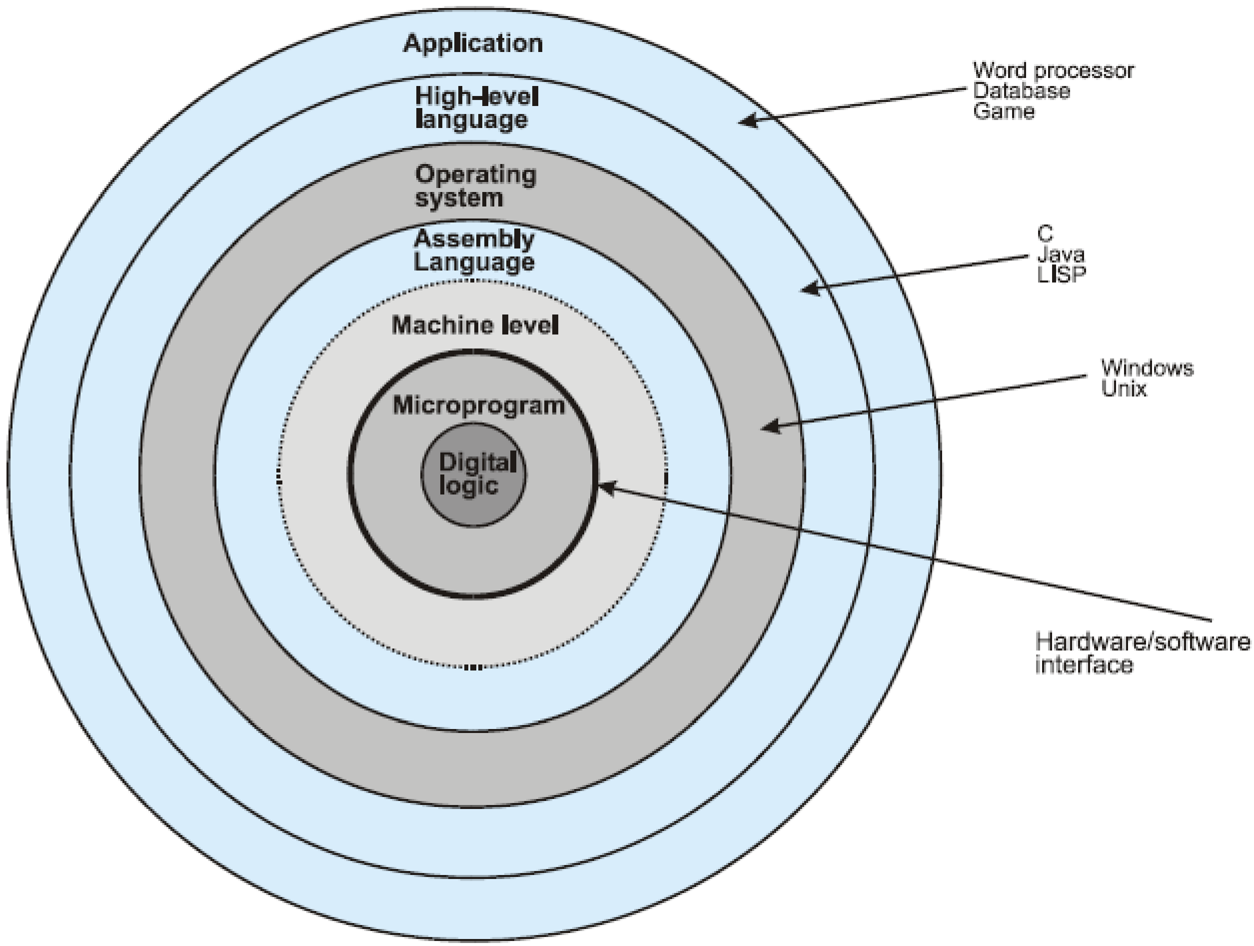
Piezoelectric [Holst/IMEC]



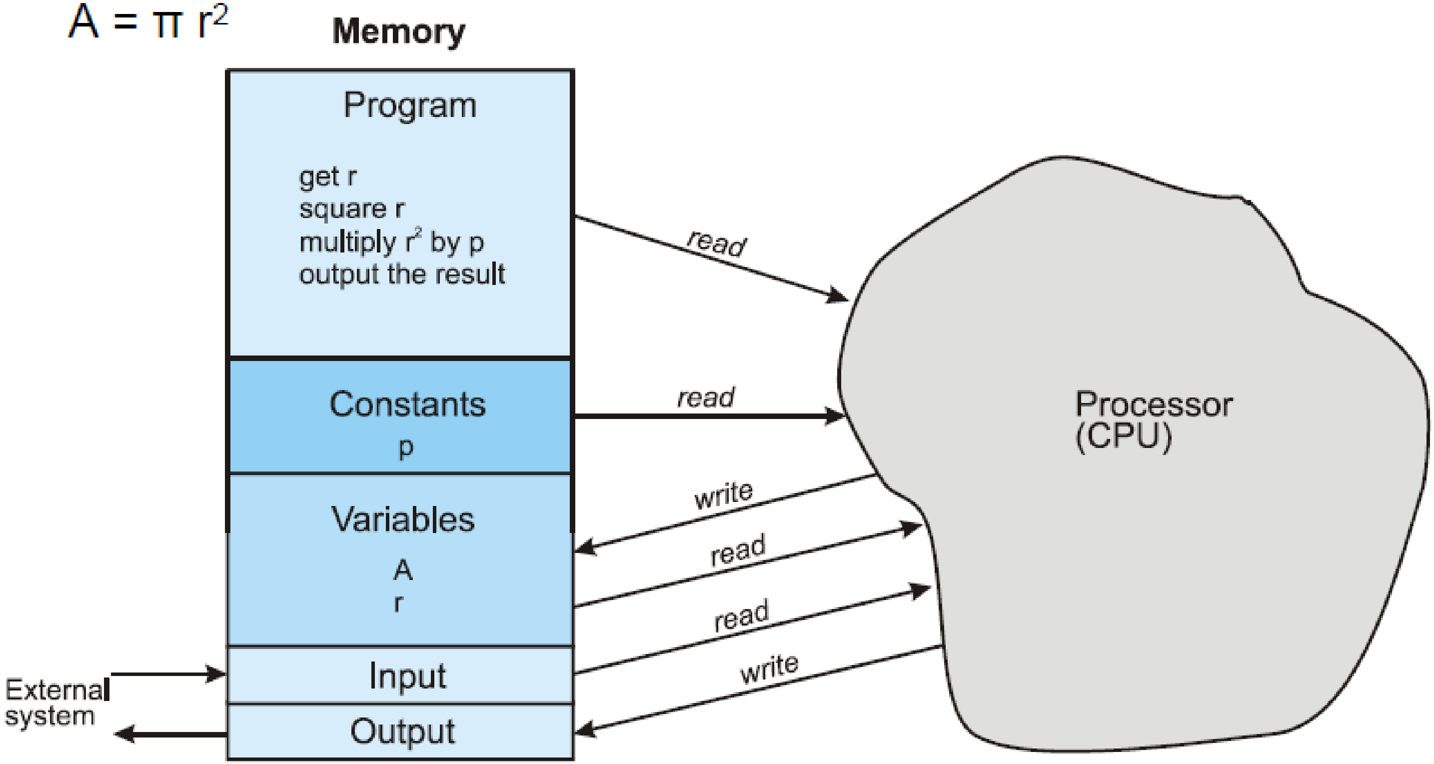


Thermoelectric Ambient Energy Harvester [PNNL]

Machine Levels

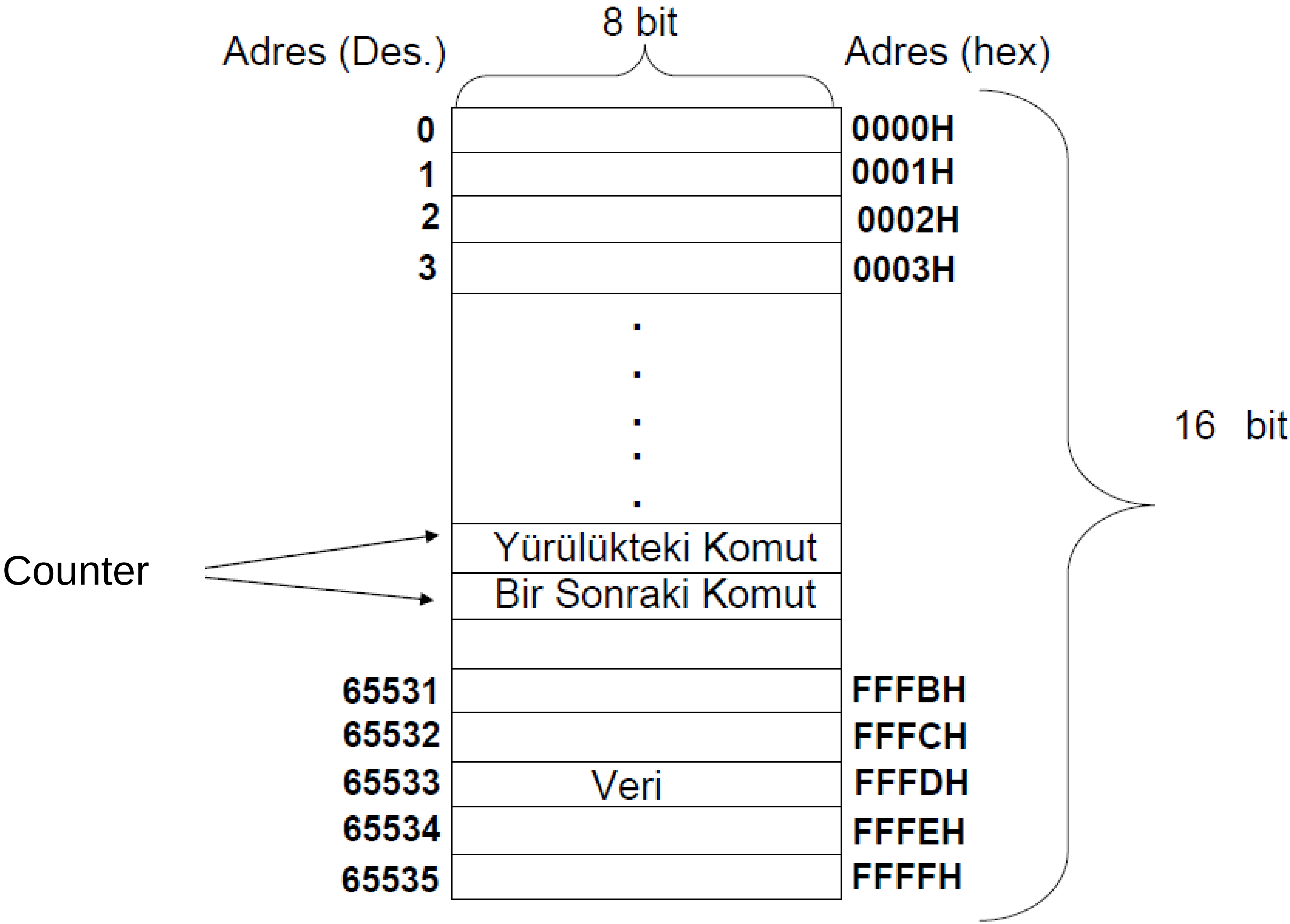


$A = \pi r^2$



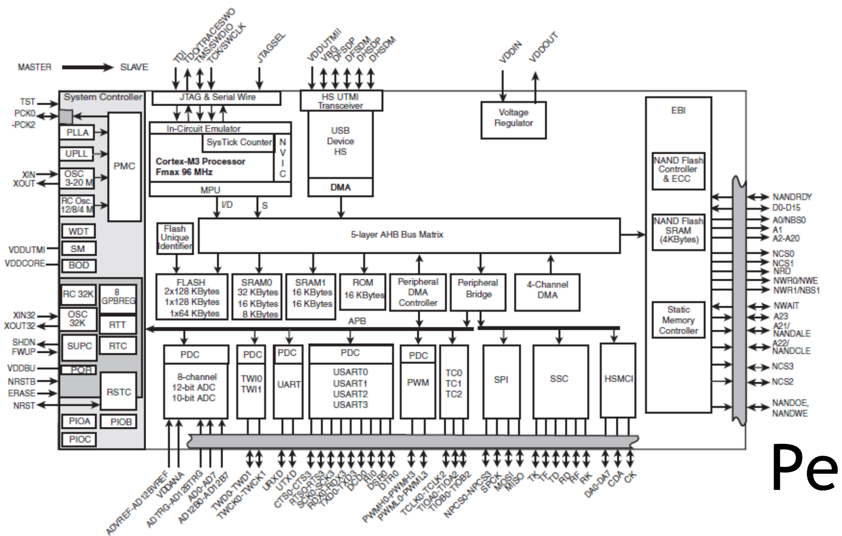
Memory
Program
get r
square r
multiply r² by p
output the result
Constants
p
Variables
A
r
Input
Output
External
system
read
read
write
read
read
write
Processor
(CPU)

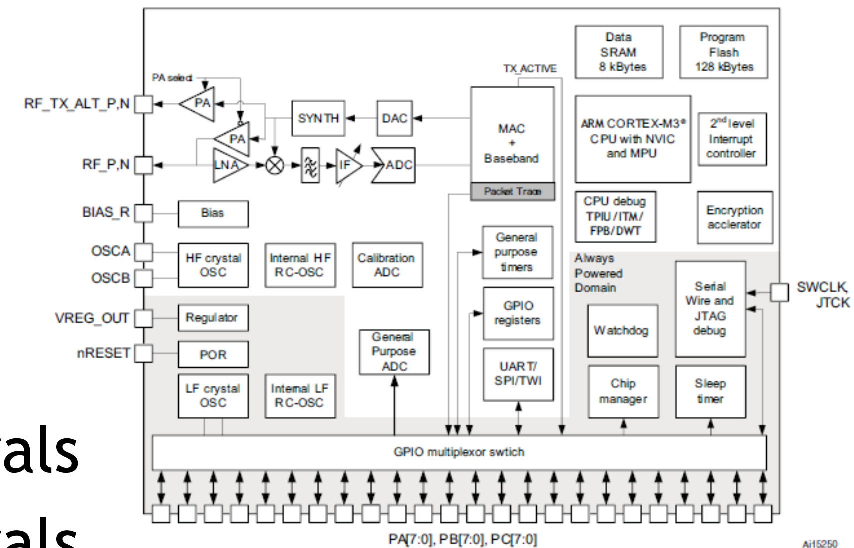
Why study 32-bit MCUs and FPGAs?


8 bit
Adres (Des.)
Adres (hex)
0
1
2
3
0000H
0001H
0002H
0003H
16 bit
Counter
Yürülükteki Komut
Bir Sonraki Komut
65531
65532
65533
65534
65535
Veri
FFFBH
FFFCH
FFFDH
FFFEH
FFFFH

Why study the ARM architecture
(and the Cortex-M3 in particular)?
What differentiates these
products from one another?

The difference is…

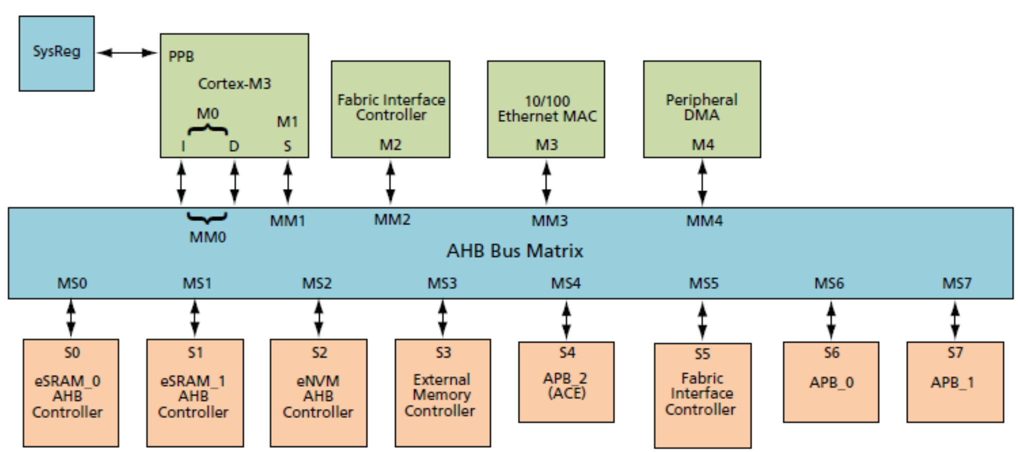




Peripherals

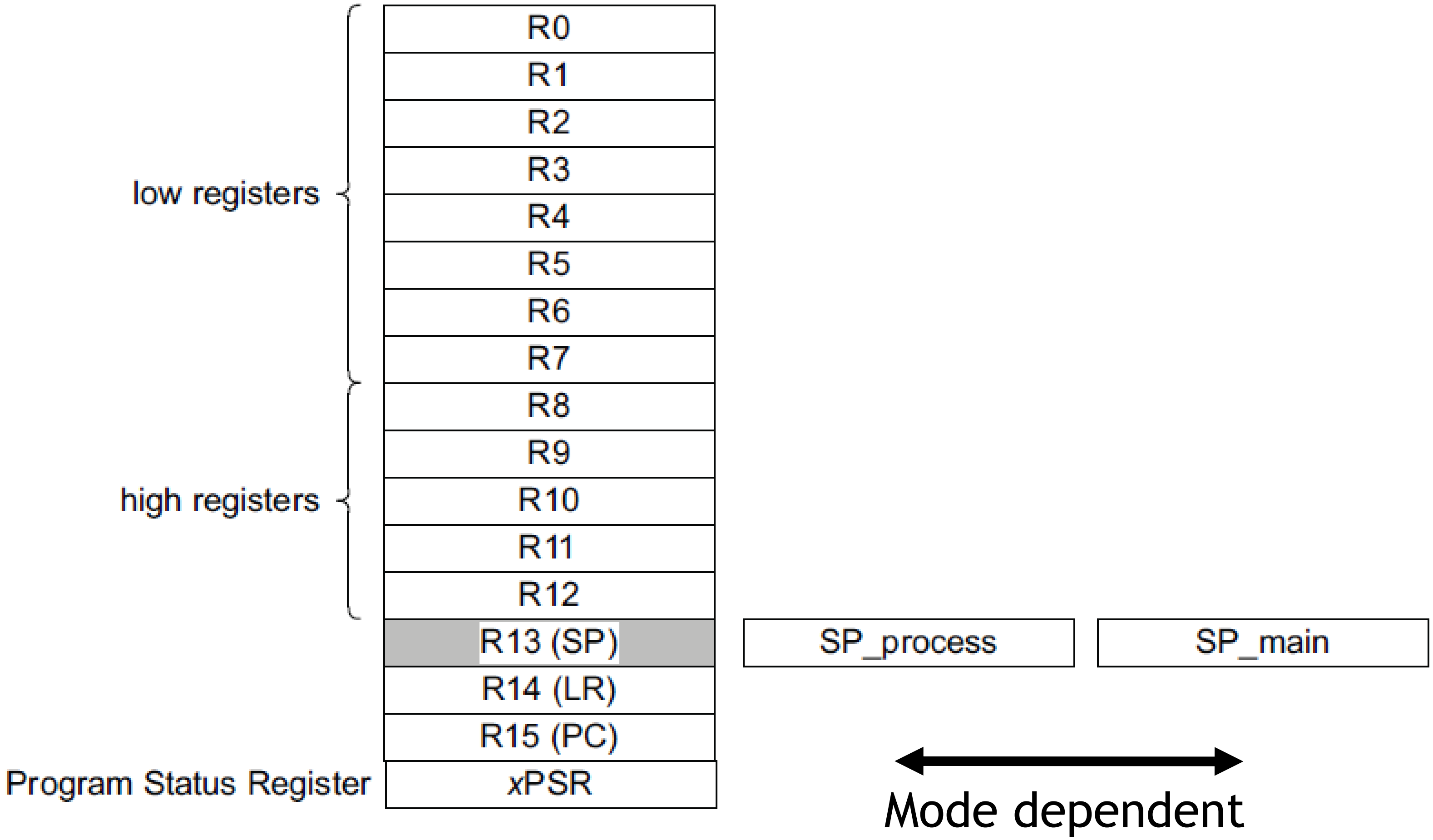
Peripherals

Peripherals

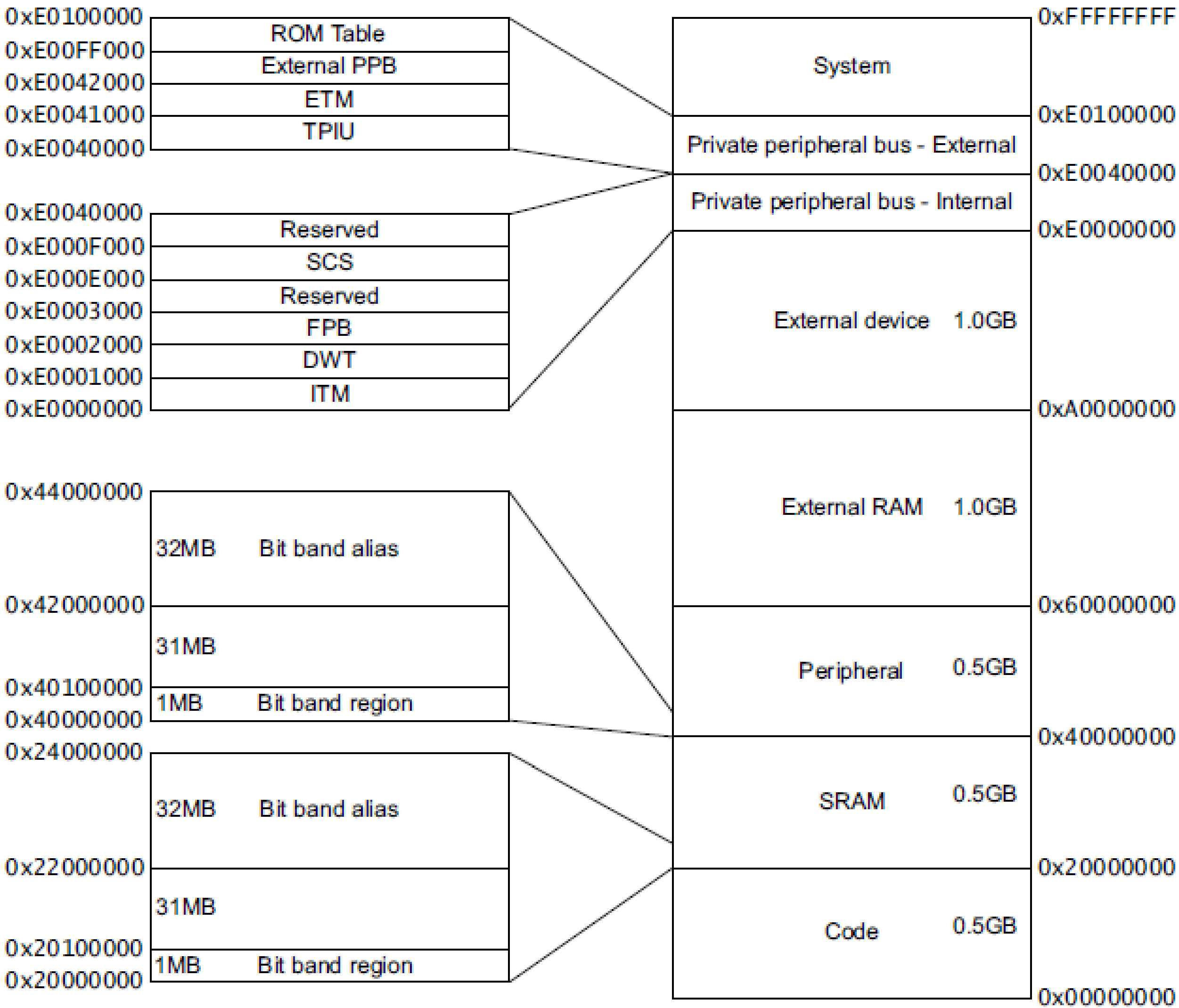


Architecture

Registers


low registers
high registers
Program Status Register
R0
R1
R2
R3
R4
R5
R6
R7
R8
R9
R10
R11
R12
R13 (SP)
R14 (LR)
R15 (PC)
xPSR
SP_process
SP_main
Mode dependent

Address Space



Data processing instructions

Table A4-2 Standard data-processing instructions

Mnemonic	Instruction	Notes
ADC	Add with Carry	-
ADD	Add	Thumb permits use of a modified immediate constant or a zero-extended 12-bit immediate constant.
ADR	Form PC-relative Address	First operand is the PC. Second operand is an immediate constant. Thumb supports a zero-extended 12-bit immediate constant. Operation is an addition or a subtraction.
AND	Bitwise AND	-
BIC	Bitwise Bit Clear	-
CMN	Compare Negative	Sets flags. Like ADD but with no destination register.
CMP	Compare	Sets flags. Like SUB but with no destination register.
EOR	Bitwise Exclusive OR	-
MOV	Copies operand to destination	Has only one operand, with the same options as the second operand in most of these instructions. If the operand is a shifted register, the instruction is an LSL, LSR, ASR, or ROR instruction instead. See *Shift instructions* on page A4-10 for details. Thumb permits use of a modified immediate constant or a zero-extended 16-bit immediate constant.

Application Program Status Register (APSR)

31	30	29	28	27	26 … 0
N	Z	C	V	Q	RESERVED

APSR bit fields are in the following two categories:

- Reserved bits are allocated to system features or are available for future expansion. Further information on currently allocated reserved bits is available in *The special-purpose program status registers (xPSR)* on page B1-8. Application level software must ignore values read from reserved bits, and preserve their value on a write. The bits are defined as UNK/SBZP.

- Flags that can be set by many instructions:

  N, bit [31] Negative condition code flag. Set to bit [31] of the result of the instruction. If the result is regarded as a two's complement signed integer, then N == 1 if the result is negative and N = 0 if it is positive or zero.

  Z, bit [30] Zero condition code flag. Set to 1 if the result of the instruction is zero, and to 0 otherwise. A result of zero often indicates an equal result from a comparison.

  C, bit [29] Carry condition code flag. Set to 1 if the instruction results in a carry condition, for example an unsigned overflow on an addition.

  V, bit [28] Overflow condition code flag. Set to 1 if the instruction results in an overflow condition, for example a signed overflow on an addition.

  Q, bit [27] Set to 1 if an `SSAT` or `USAT` instruction changes (saturates) the input value for the signed or unsigned range of the result.