Introduction to mobile devices

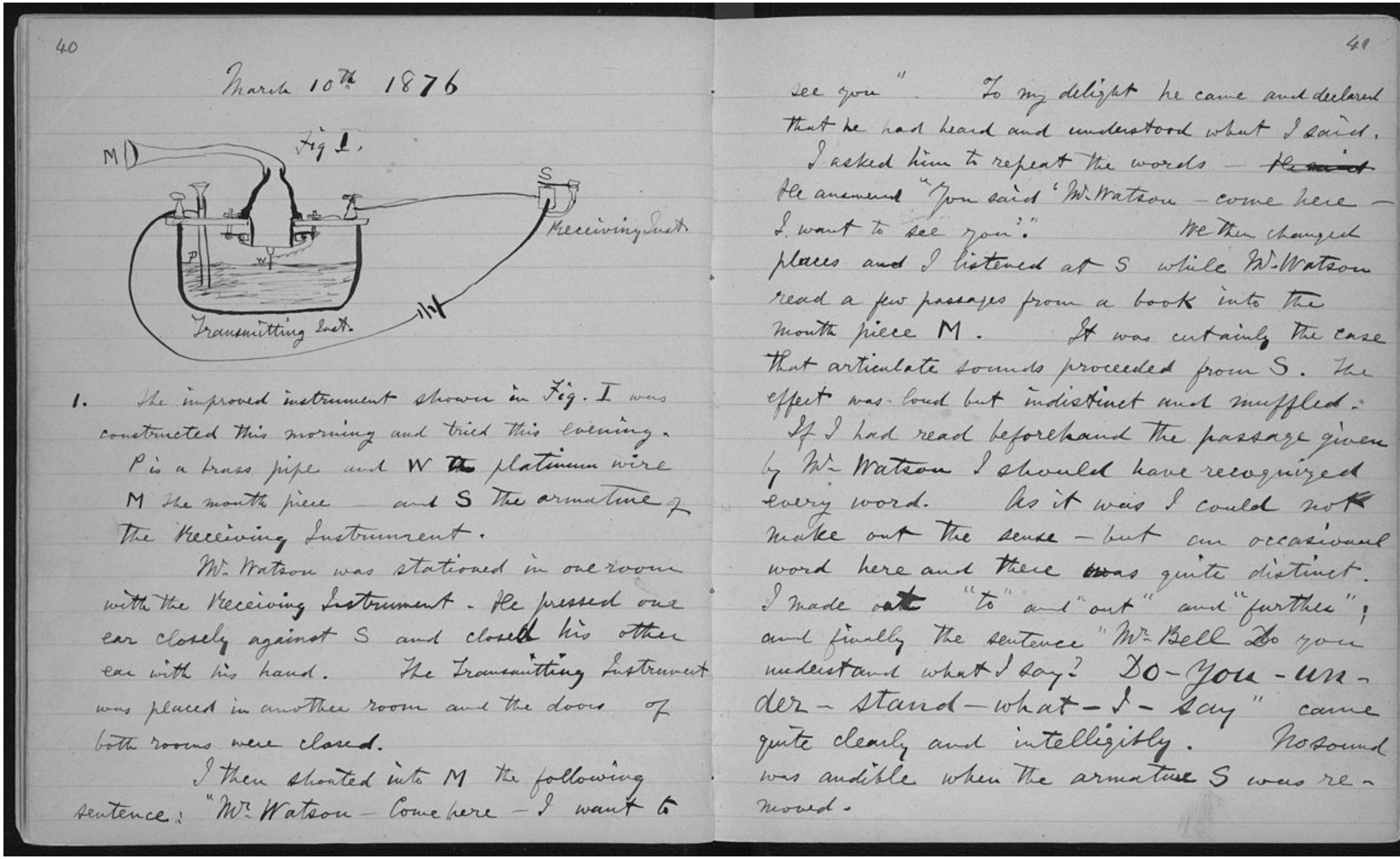
CS 436 Software Development on Mobile

By Dr.Paween Khoenkaw

Welcome to the world of smart devices

http://www.rudebaguette.com/2014/02/13/mobile-smart-devices-set-drive-turnaround-frances-tech-device-market/

The fist telephone


40

March 10th 1876

1\. The improved instrument shown in Fig. I was constructed this morning and tried this evening.

P is a brass pipe and W ~~the~~ the platinum wire M the mouth piece — and S the armature of the Receiving Instrument.

Mr. Watson was stationed in one room with the Receiving Instrument. He pressed one ear closely against S and closed his other ear with his hand. The Transmitting Instrument was placed in another room and the doors of both rooms were closed.

I then shouted into M the following sentence: "Mr. Watson — Come here — I want to

41

see you." To my delight he came and declared that he had heard and understood what I said.

I asked him to repeat the words — ~~He said~~ He answered "You said 'Mr. Watson — come here — I want to see you.'" We then changed places and I listened at S while Mr. Watson read a few passages from a book into the mouth piece M. It was certainly the case that articulate sounds proceeded from S. The effect was loud but indistinct and muffled.

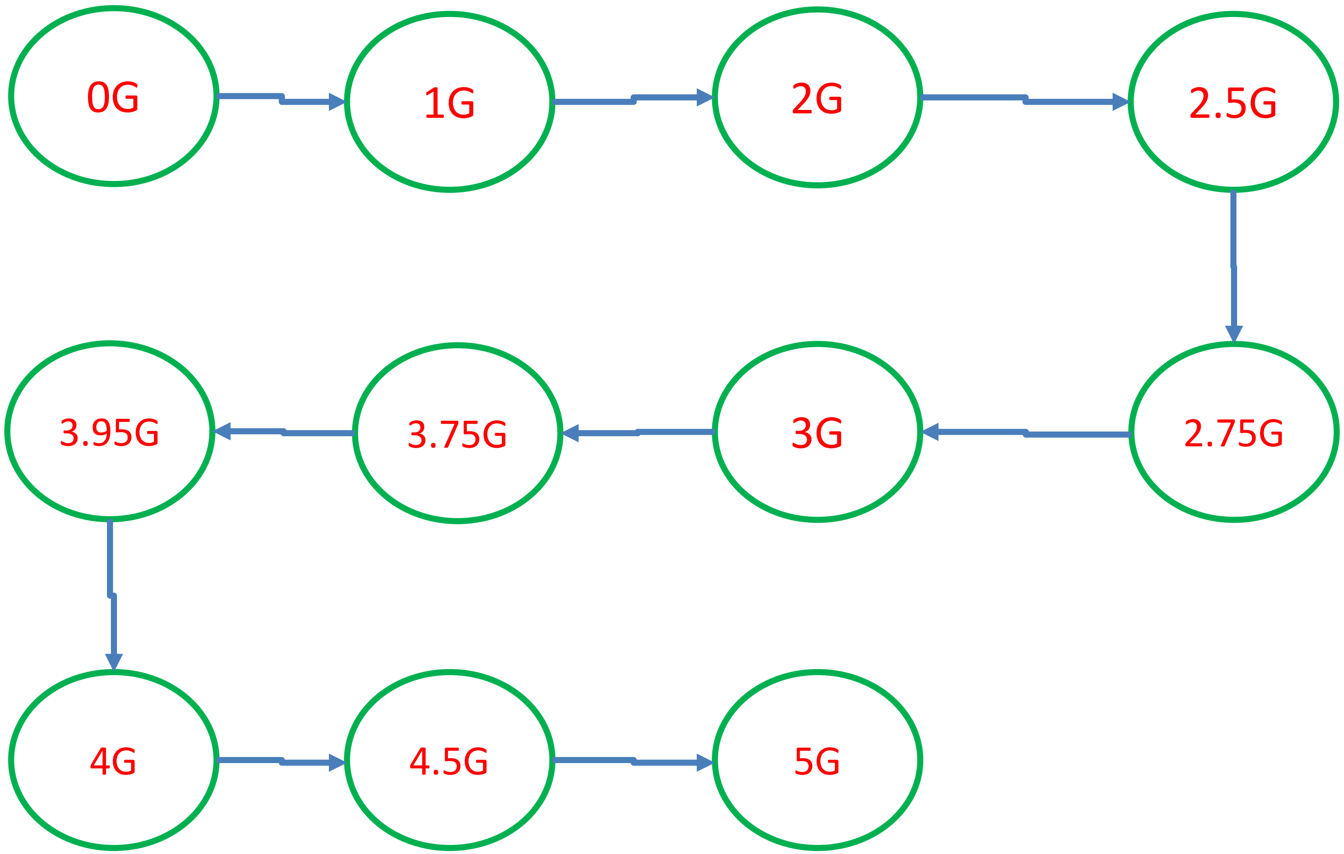
If I had read beforehand the passage given by Mr. Watson I should have recognized every word. As it was I could not make out the sense — but an occasional word here and there ~~was~~ was quite distinct.

I made ~~out~~ out "to" and "out" and "further"; and finally the sentence "Mr. Bell Do you understand what I say? Do-you-un-der-stand-what-I-say" came quite clearly and intelligibly. No sound was audible when the armature S was removed.

Bell's March 10, 1876 laboratory notebook entry describing his first successful experiment with the telephone.

We make a phone call to a place

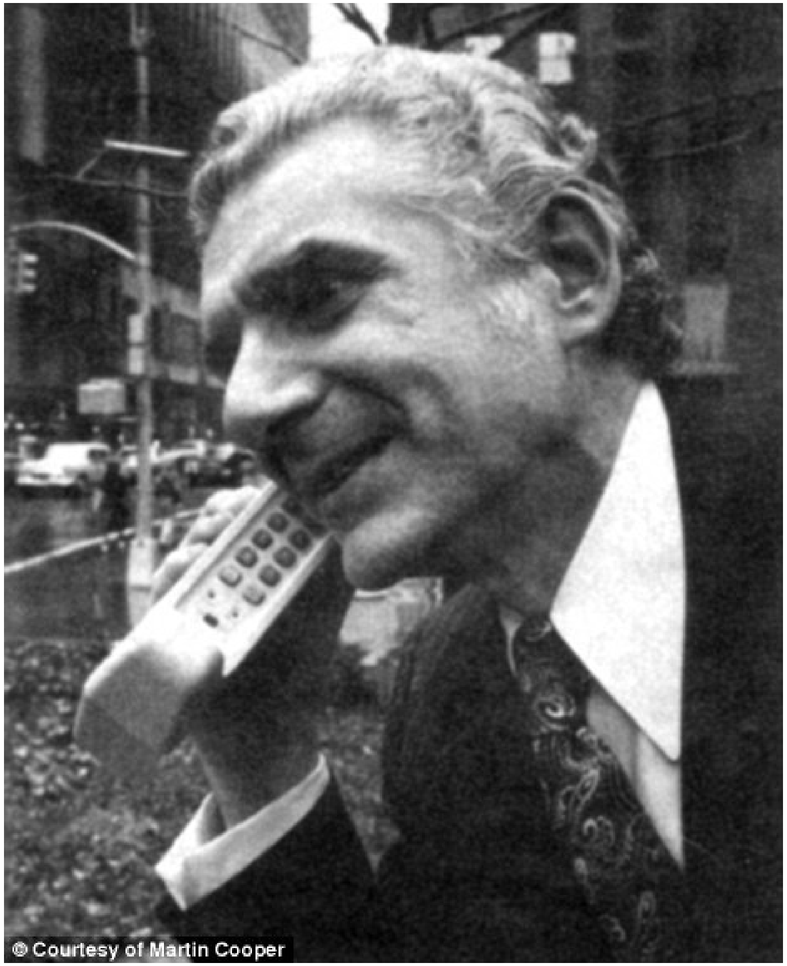
Mobile phone generations



O G The early mobile phones

Car phone service originated with the Bell System, and was first used in St. Louis on June 17, **1946**. The original equipment weighed 80 pounds (**36 kg**), and there were initially only **3** channels for all the users in the metropolitan area

The first hand-held mobile phone



Martin Cooper and DynaTAC

On April 3, **1973** Cooper and Mitchell demonstrated two working phones

Cooper dialed the number of his chief competitor Dr. Joel S. Engel, who was head of Bell Labs. "***Joel, this is Marty. I'm calling you from a cell phone, a real handheld portable cell phone*.**"

1G Cellular Phones

1979

NMT – Nordic Mobile Telephony

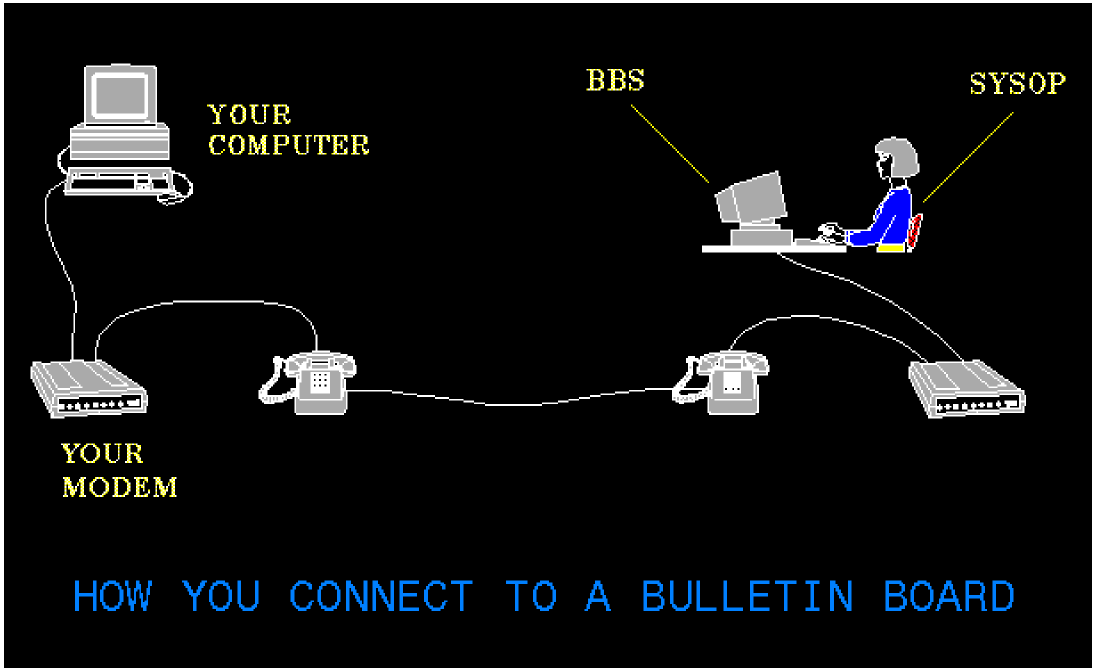
TACS – Total Access Communication System

AMPS - Advanced Mobile Phone System

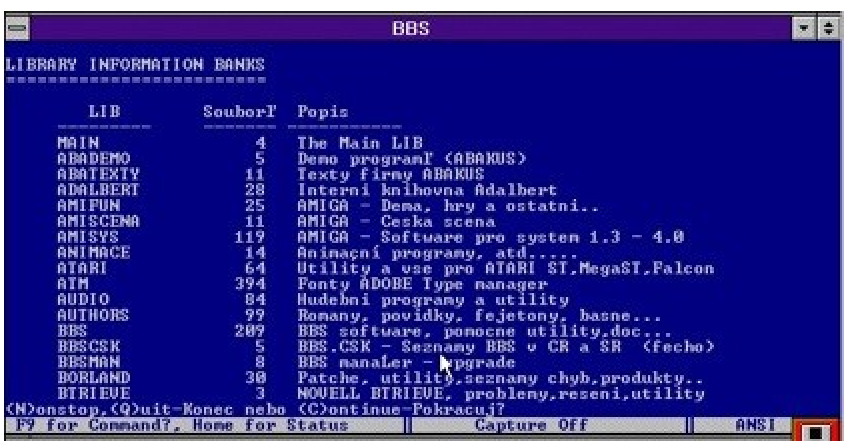
ETACS – Total Access Communication System

- Analog System
- Low capacity
- Do not coverage long distance
- Not Secured

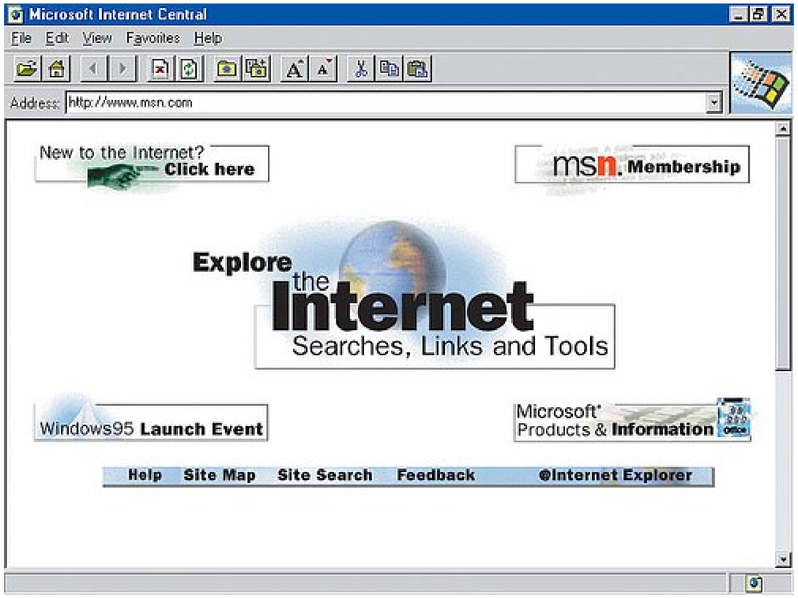
Connected PCs



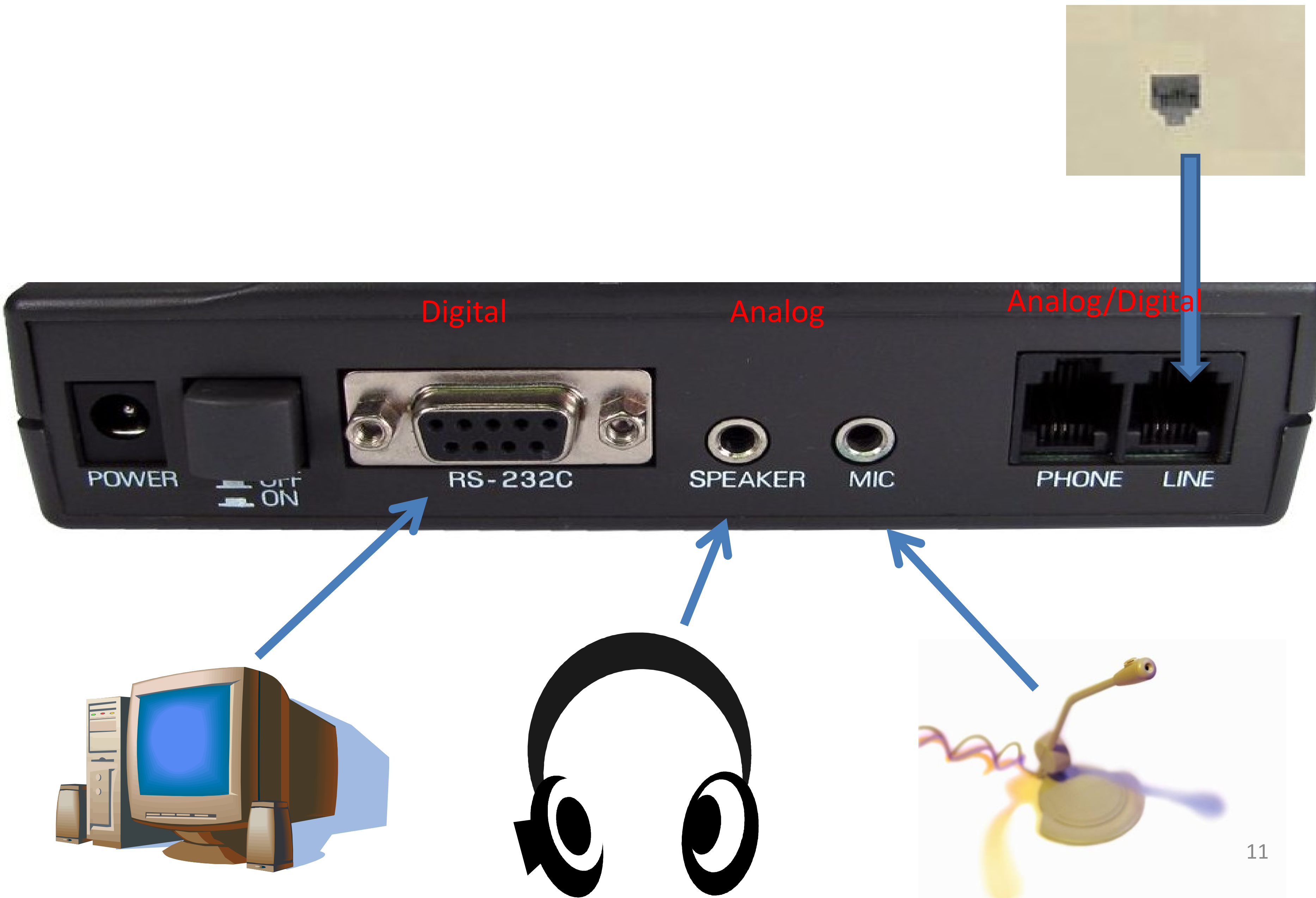
Bulletin board system (early 80’s to late 90’s)



Down of the internet



Digital data as analog audio

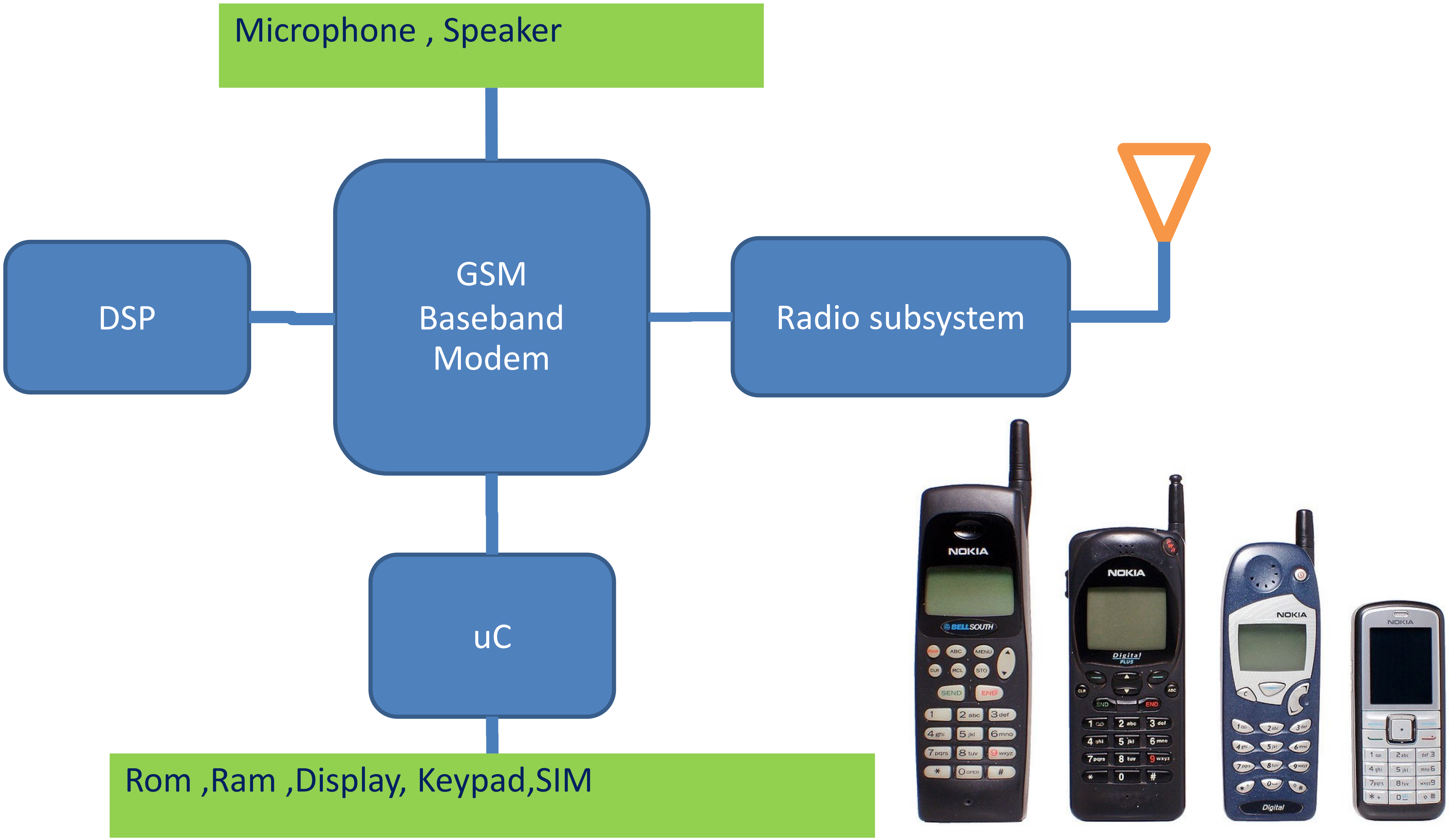


The era of laptop PC

Pager

Pager is a device that received the wireless broadcast digital message

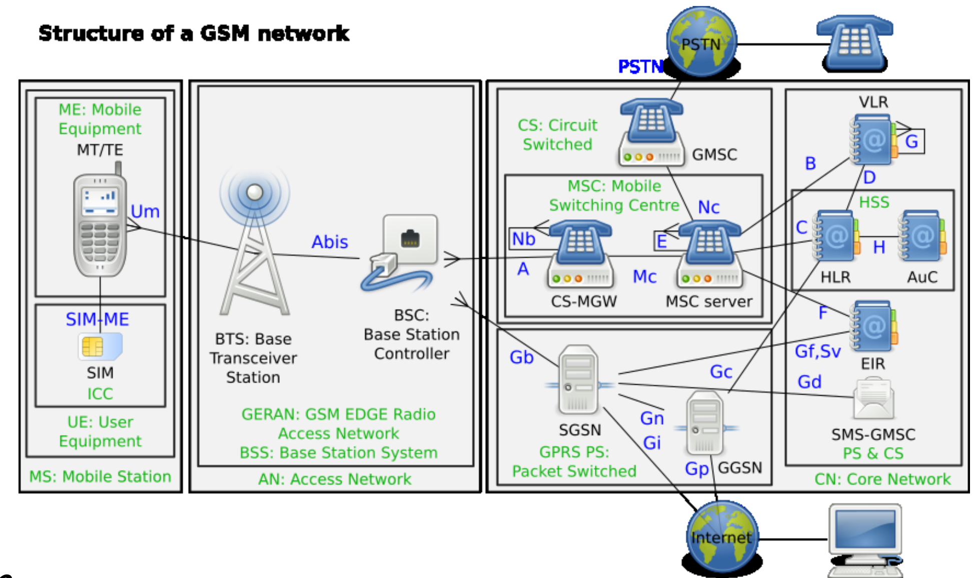
2G GSM Phone



Microphone , Speaker
DSP
GSM
Baseband
Modem
Radio subsystem
uC
Rom ,Ram ,Display, Keypad,SIM
NOKIA
BELLSOUTH
NOKIA
NOKIA
NOKIA
Digital

2G GSM Phone

1990



GSM – *Groupe Spécial Mobile*

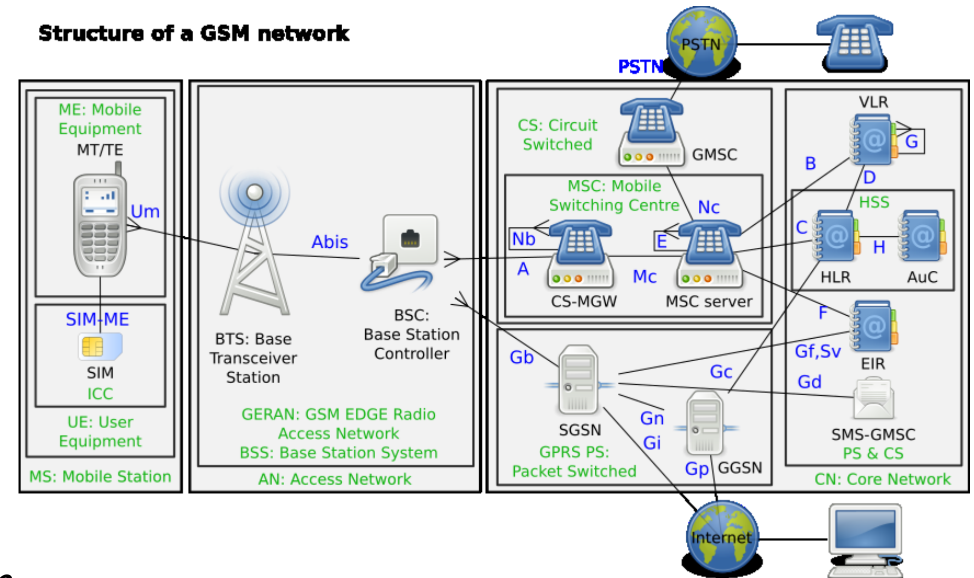
GSM – ***Global System*** *for Mobile Communications*

- Digital voice communication
- Coverage long distance
- Globally Accepted (roaming)
- Short Message
- Secured
- CSD (Circuit Switch Data) @9.6kbps

https://en.wikipedia.org/wiki/GSM

2.5G GSM Phone

1990



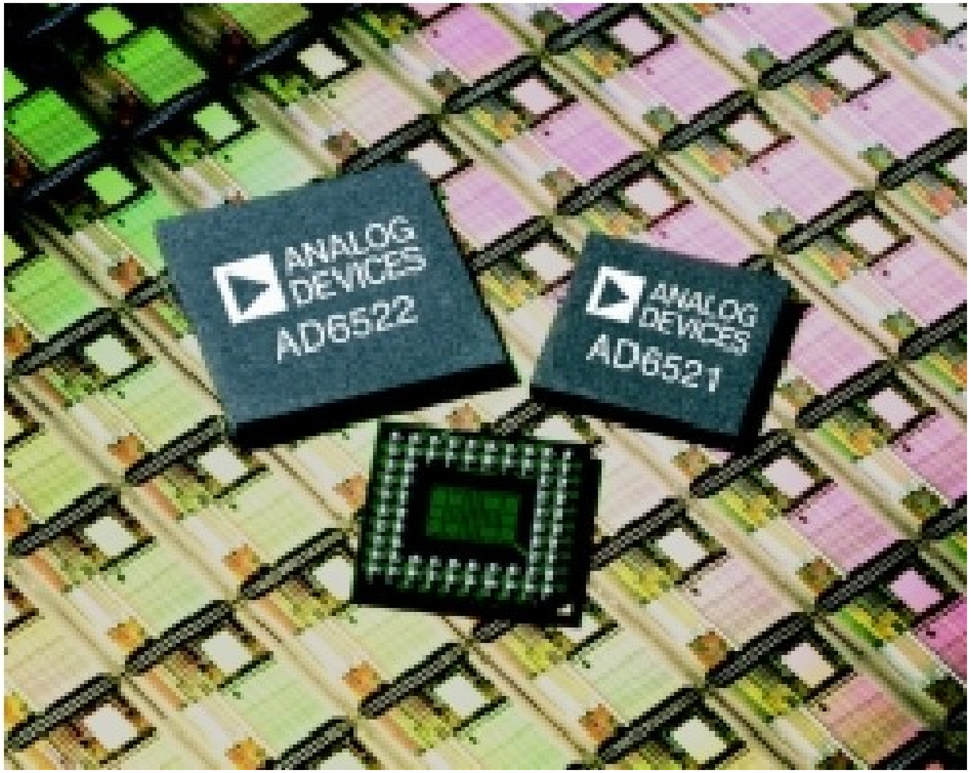
GSM – *Groupe Spécial Mobile*

GSM – ***Global System** for Mobile Communications*

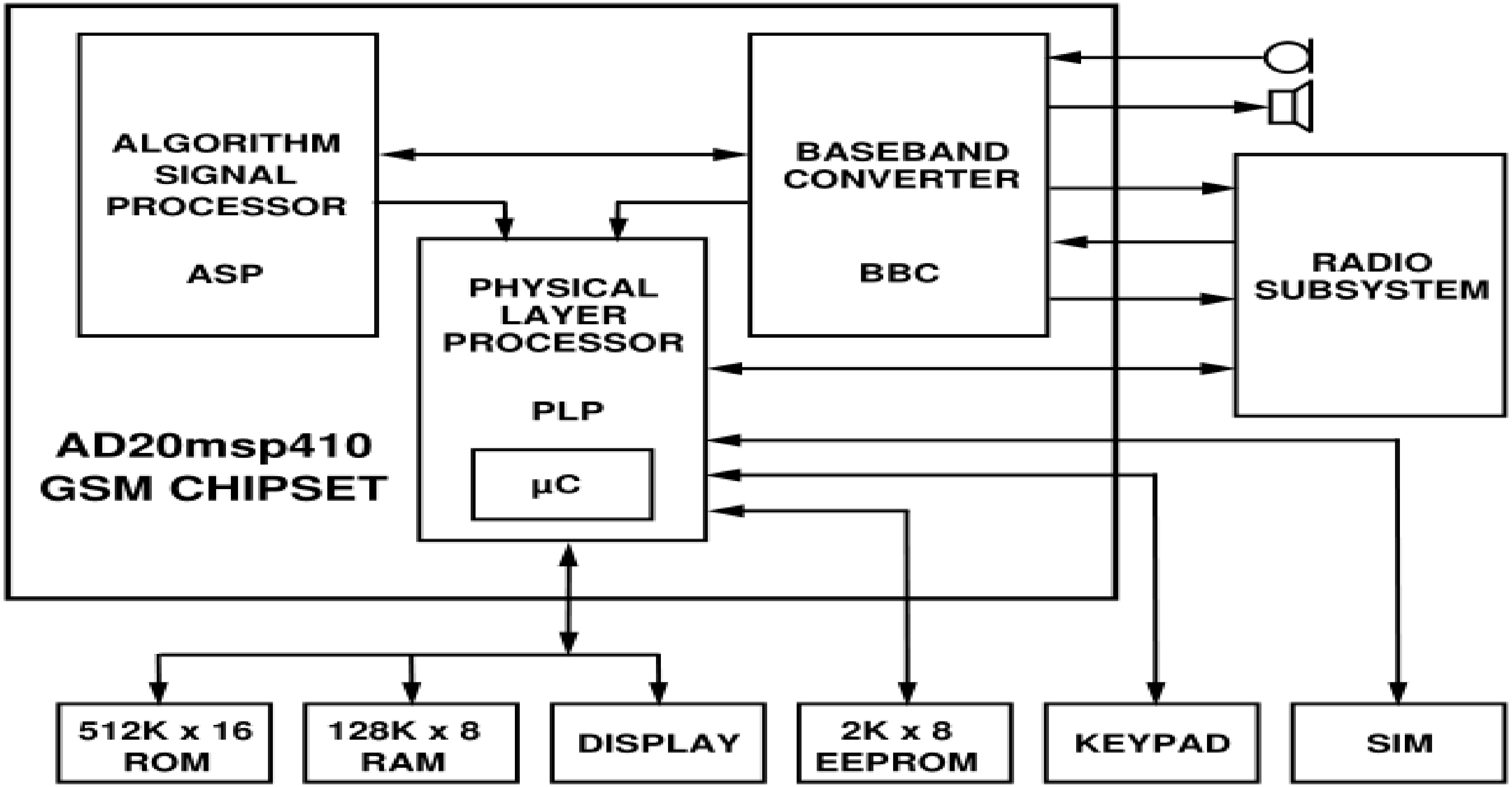
- Digital voice communication
- Coverage long distance
- Globally Accepted (roaming)
- Short Message
- Secured
- Data Channel
- GPRS (General Package Radio Service)@64kbps

GSM Chipset

MediaTek
Broadcom
Icera
Infineon
Qualcomm
ST-Ericsson
....



GSM Chipset



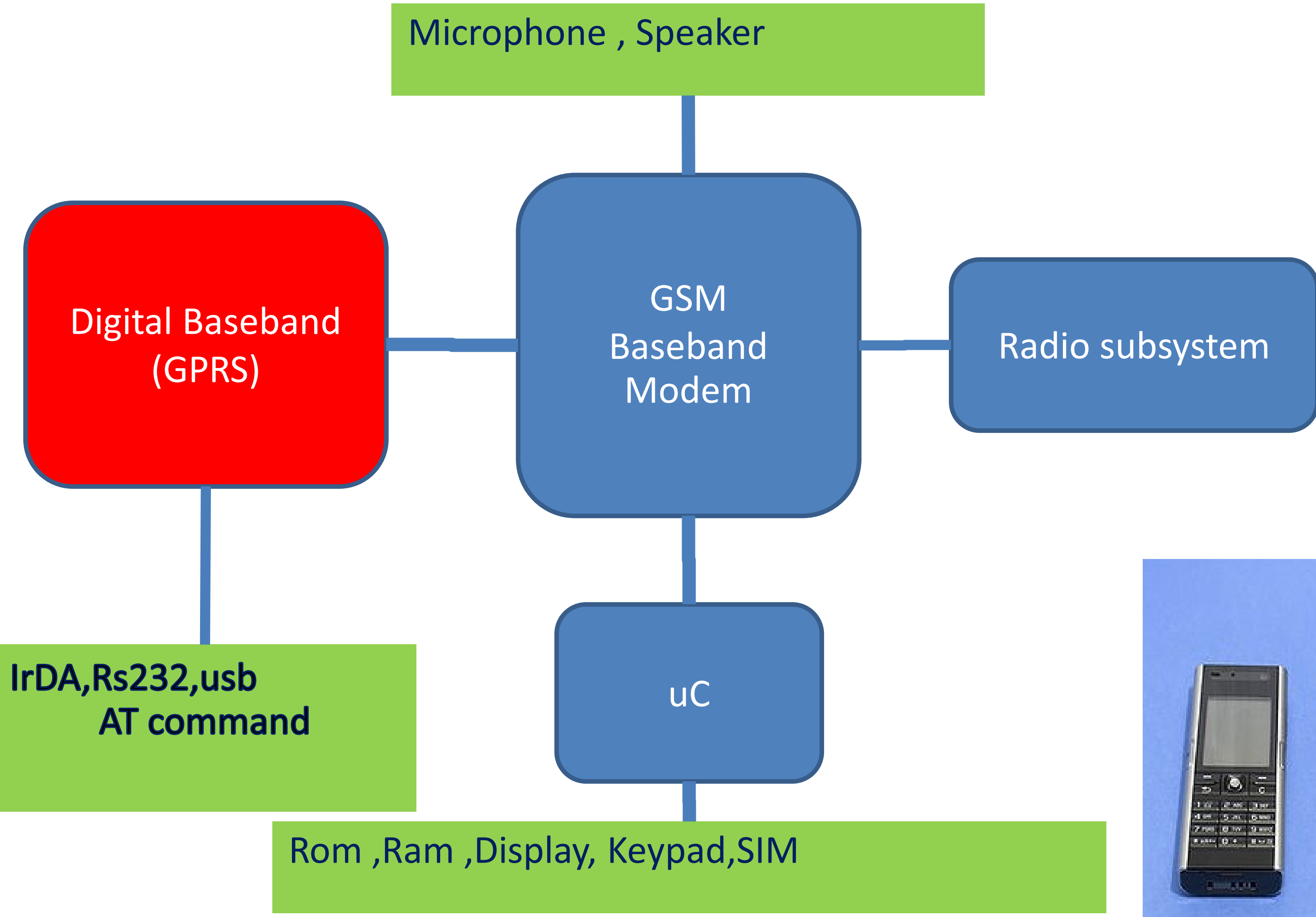
SYSTEM ARCHITECTURE
AD20msp410

The phone now have some features

		- SMS Templates
CAMERA		No
SOUND	Alert types	Vibration; Monophonic ringtones, composer
	Loudspeaker	No
	3.5mm jack	No
COMMS	WLAN	No
	Bluetooth	No
	GPS	No
	Radio	No
	USB	
FEATURES	Messaging	SMS
	Browser	WAP
	Clock	Yes
	Alarm	Yes
	Games	4 - Tetris, Erix, E-maze, Ballpop
	Languages	24
	Java	No
		- Swatch Internet Time - Voice dial - Voice answer - Calculator - Stopwatch - Start-up shut-down shows - Profiles - Active flip
BATTERY		NiMH battery
	Stand-by	200 h
	Talk time	10 h
MISC	Colors	4 - Blue Whirl, Lime Twist, Silver Weave, Beige Harmony

Disclaimer. We can not guarantee that the information on this page is 100% correct. Read more

Digital baseband

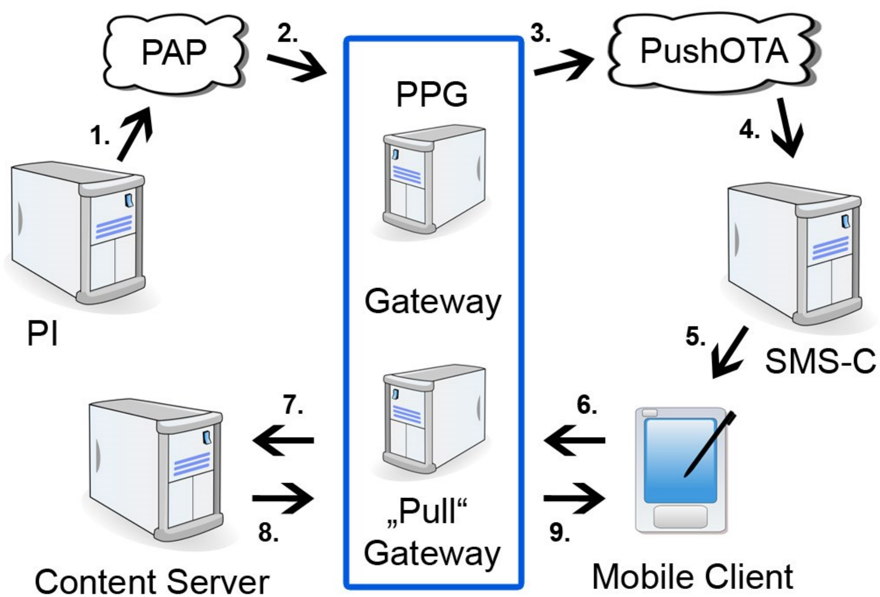


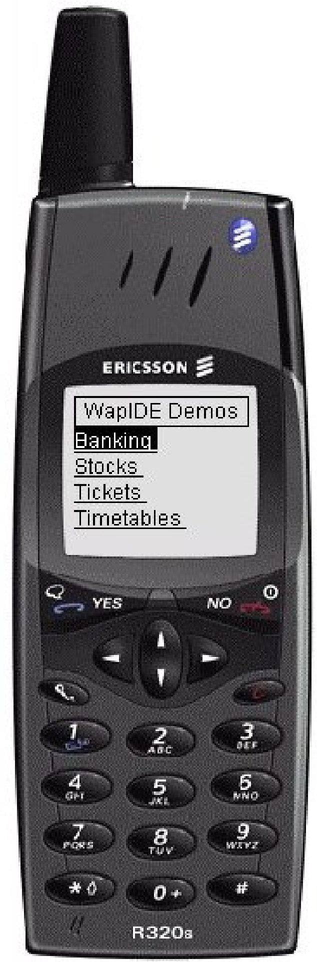
Interactive Pager

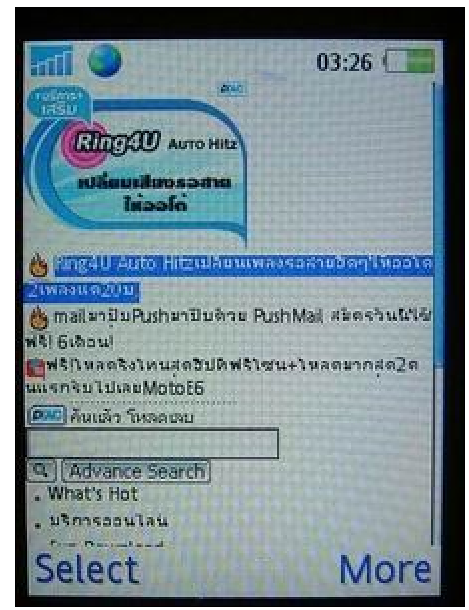
Launched by RIM in 1996. It specialized in two-way messaging and had limited HTML access, though it was e-mail capable.

Internet on the move

Wireless Application Protocol (WAP) + General packet radio service (GPRS)





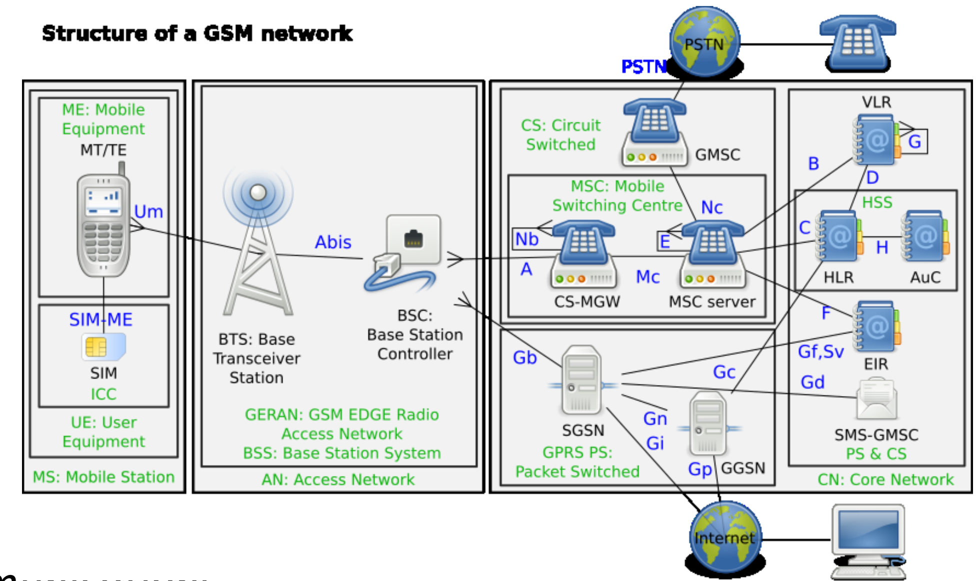


Mobile Programming Language

Qt NOKIA

2.75G GSM Phone

2003



GSM – *Groupe Spécial Mobile*

GSM – ***Global System*** *for Mobile Communications*

- Digital voice communication
- Coverage long distance
- Globally Accepted (roaming)
- Short Message
- Secured
- Data Channel
- GPRS (General Package Radio Service)
- EDGE (Enhance Data rates for GSM Evolution)@170kpbs

2G CDMA Phone

- CDMA (Code-division multiple access)
- SSMA (spread-spectrum multiple access)

3G GSM Phone with Faster Data

2007

- UMTS (Universal Mobile Telecommunication System) @384kbps
- Video Calling

Target

4G Very Fast Data

- 1Gbps for Stationary users
- 100Mbps for High mobility users (Bullet train)

PCs in the pocket

Personal Digital Assistant

Apple Newton MessagePad 100

Manufacturer	Apple Computer
Release date	1993
Discontinued	1998
Operating system	Newton OS
CPU	ARM 610 RISC
Weight	1.4 lb (0.64 kg) W/ Battery

Pda

-Motorola MC68328
-68000 Core
-32-bit CISC microprocessors
-UART
-Touch screen
-Palm OS

PDA have no phone capability

PDA

Application Processor

Baseband Processor

THE SMART PHONE

iPhone

iPhone 1

2007

- Internet
- Application

3G GSM Phone with Faster Data

2007

- UMTS (Universal Mobile Telecommunication System) @384kbps
- Video Calling

4G Very Fast Data

- 1Gbps for Stationary users
- 100Mbps for High mobility users (Bullet train)

Android phone
2008

- Internet
- Application

3.5G GSM Phone with Faster Data

2007

- HSDPA (High Speed Downlink Packet Access)@2Mbps

- HSUPA (High Speed Uplink Packet Access)@2Mbps

3.75G GSM Phone with Faster Data

- HSPA+ (Evolved High Speed Packet Access Plus)
- Downlink@42.2Mbps
- Uplink@22Mbps

3.9G GSM Phone with Faster Data 2009

4G LTE phone: Samsung Galaxy S Aviator
Android 2.3
2012

- LTE (Long Term Evolution)
- 4G LTE
- Downlink @ 300Mbps
- Uplink @ 75Mbps

3G W-CDMA Phone

- W-CDMA Wideband Code Division Multiple Access
- CDMA2000
- EVDO (Evolution-Data Optimized)

4G Fast Data

In 2011, Thailand's Truemove-H launched a pre-4G HSPA+ network with nationwide availability.

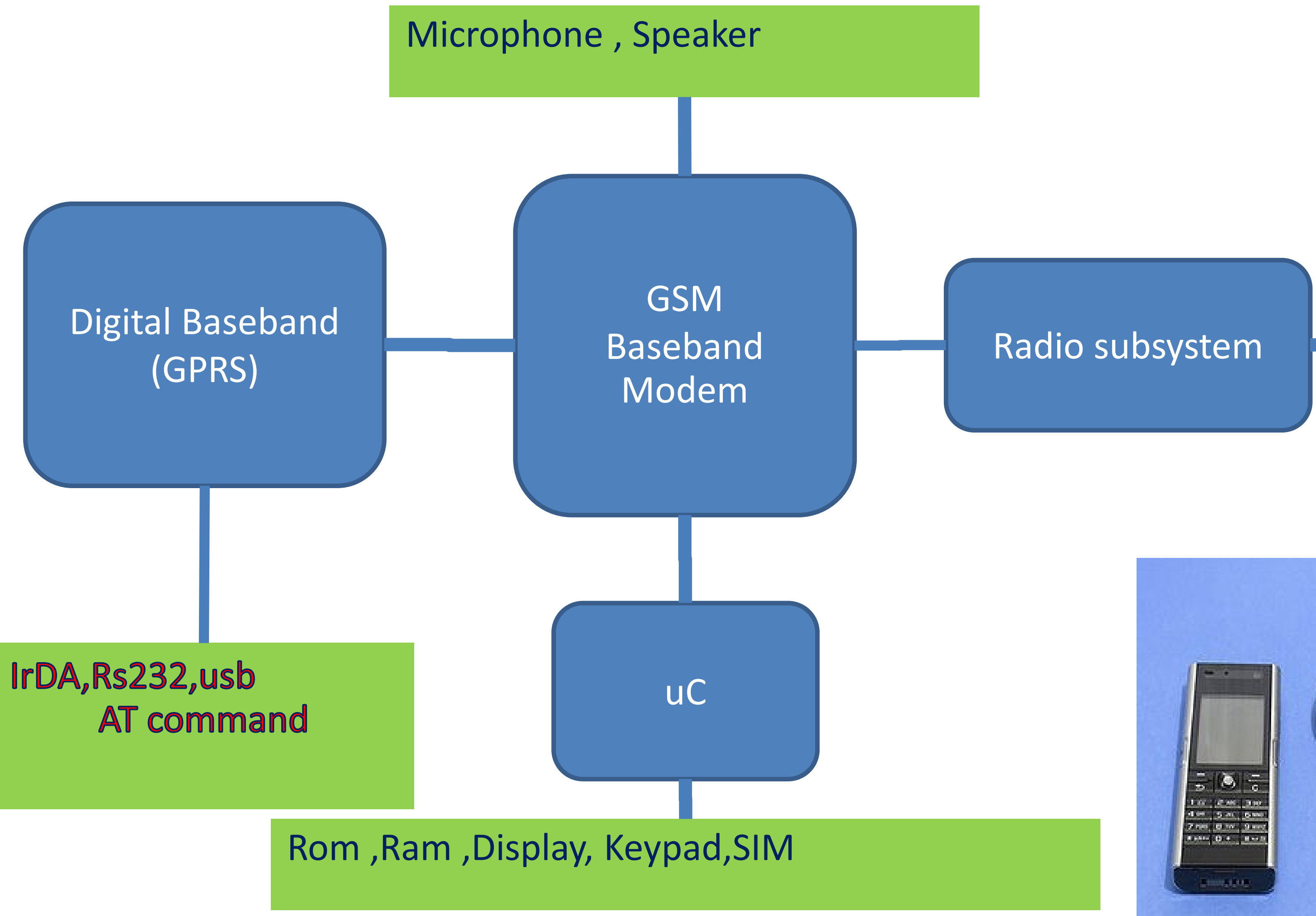
- LTE (Long Term Evolution)
- 4G LTE
- LTE Advanced
- 4G Voice call
- Fall back to 3G for voice calling
- Downlink @1000Mbps
- Uplink @ 500Mbps

5G ??????????

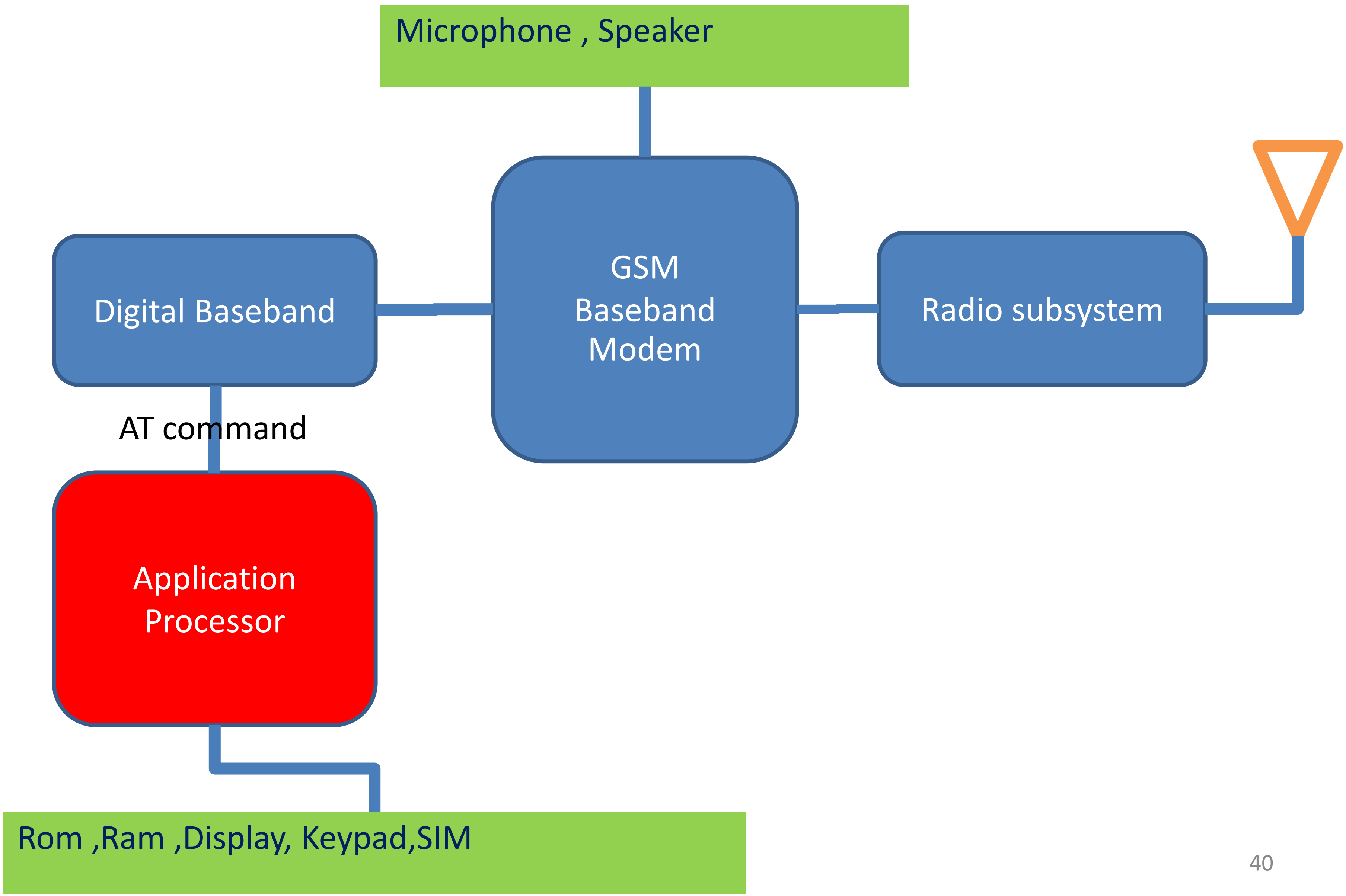
- 5 Gbps data rate

https://en.wikipedia.org/wiki/4G

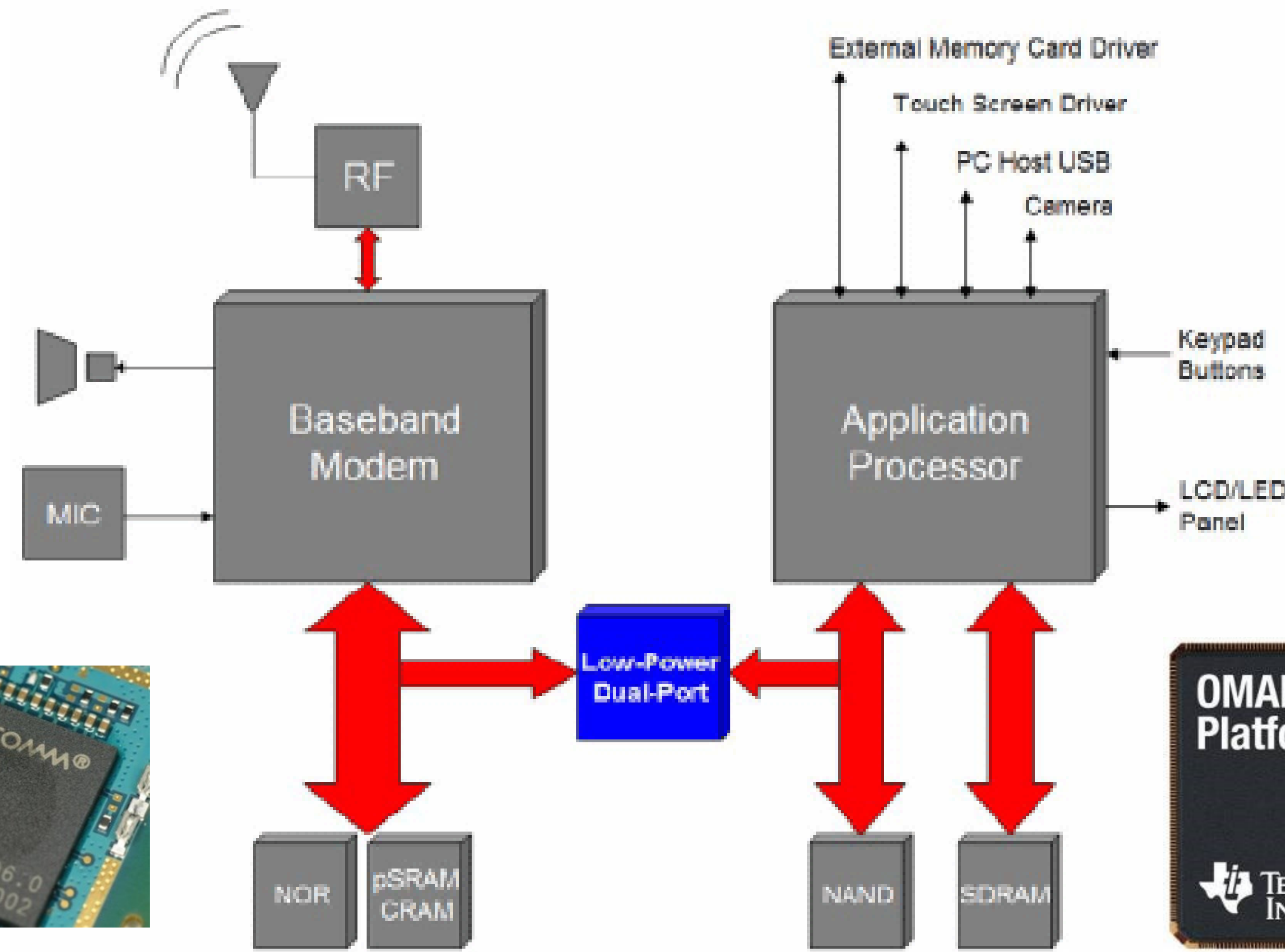
THE FEATURE PHONE



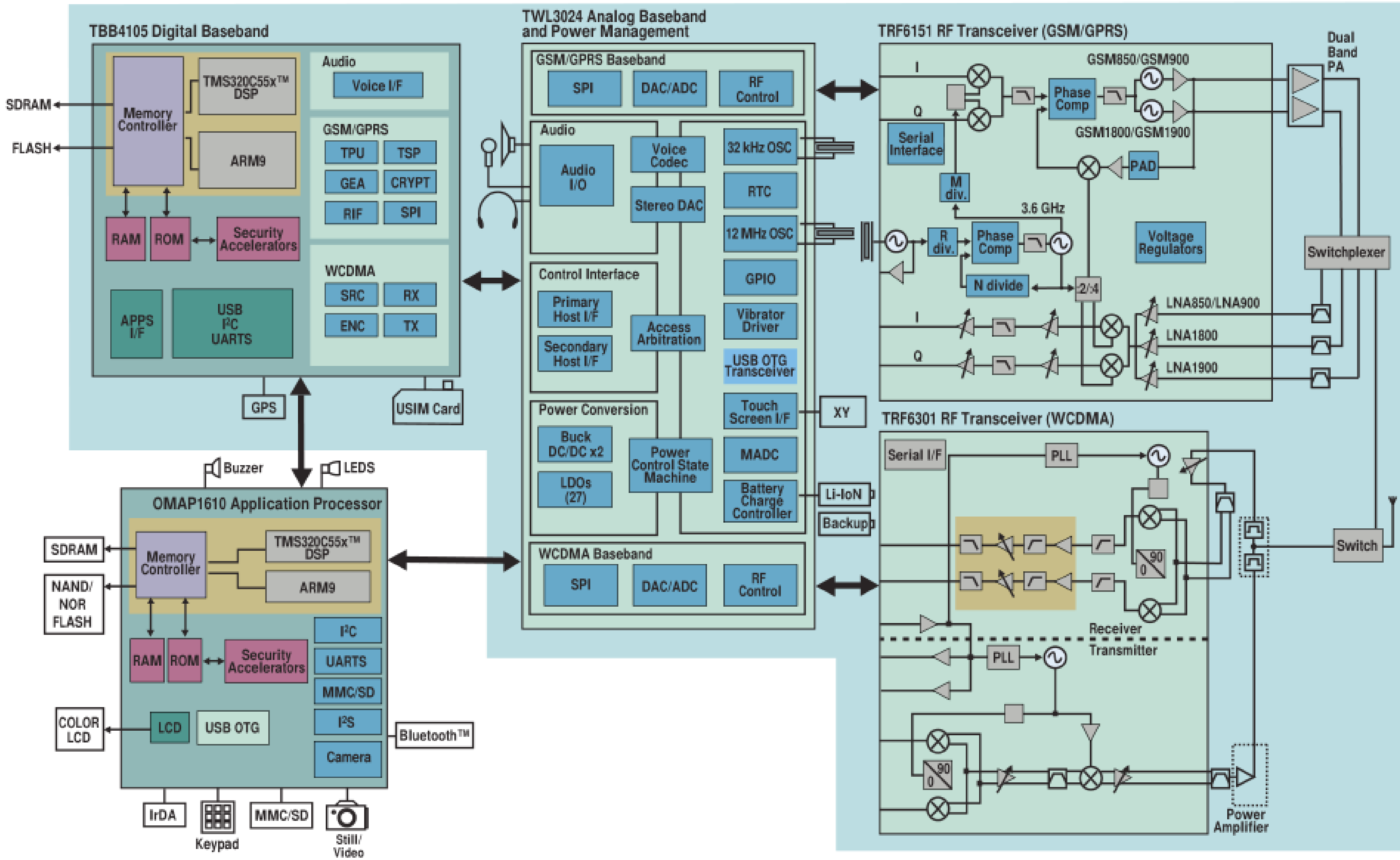
THE SMART PHONE



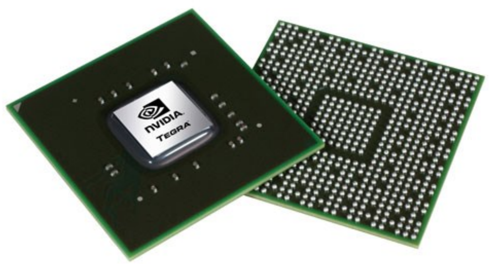
THE SMART PHONE



THE SMART PHONE



Processor wars



MIPS

ARM

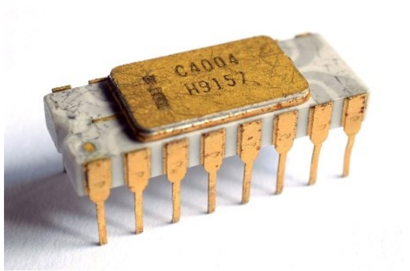
ATOM

Instruction Set Architecture Wars

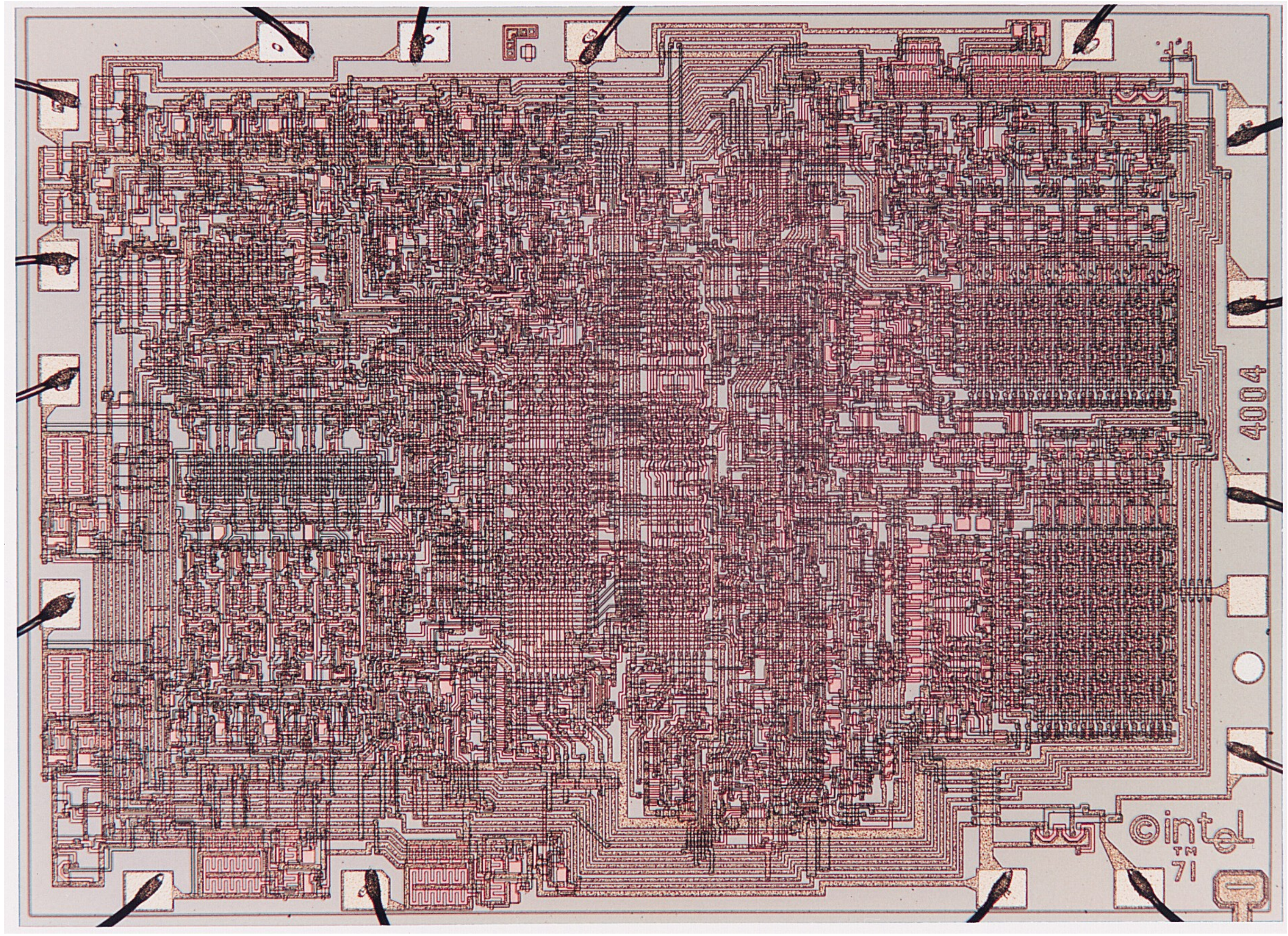
CISC vs RISC

Instruction Set Architecture Wars

The first general-purpose CPU



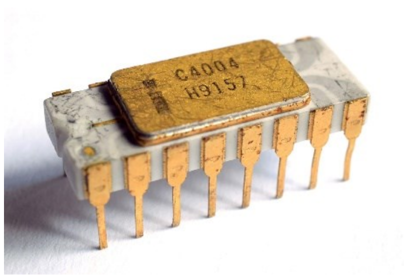
Intel4004
1970
4bits Data width
2250 Transistors
46 Instructions
740kHz



https://en.wikipedia.org/wiki/Intel_4004

Instruction Set Architecture Wars

The first general-purpose CPU



Intel4004
1970
4bits Data width
2250 Transistors
46 Instructions
740kHz

Intel 4004 Instructions Set

INSTRUCTION	MNEMONIC	BINARY EQUIVALENT 1st byte	2nd byte	MODIFIERS
No Operation	NOP	00000000	-	none
Jump Conditional	JCN	0001CCCC	AAAAAAAA	condition, address
Fetch Immediate	FIM	0010RRR0	DDDDDDDD	register pair, data
Send Register Control	SRC	0010RRR1	-	register pair
Fetch Indirect	FIN	0011RRR0	-	register pair
Jump Indirect	JIN	0011RRR1	-	register pair
Jump Uncoditional	JUN	0100AAAA	AAAAAAAA	address
Jump to Subroutine	JMS	0101AAAA	AAAAAAAA	address
Increment	INC	0110RRRR	-	register
Increment and Skip	ISZ	0111RRRR	AAAAAAAA	register, address
Add	ADD	1000RRRR	-	register
Subtract	SUB	1001RRRR	-	register
Load	LD	1010RRRR	-	register
Exchange	XCH	1011RRRR	-	register
Branch Back and Load	BBL	1100DDDD	-	data
Load Immediate	LDM	1101DDDD	-	data
Write Main Memory	WRM	11100000	-	none
Write RAM Port	WMP	11100001	-	none
Write ROM Port	WRR	11100010	-	none
Write Status Char 0	WR0	11100100	-	none
Write Status Char 1	WR1	11100101	-	none
Write Status Char 2	WR2	11100110	-	none
Write Status Char 3	WR3	11100111	-	none
Subtract Main Memory	SBM	11101000	-	none
Read Main Memory	RDM	11101001	-	none
Read ROM Port	RDR	11101010	-	none
Add Main Memory	ADM	11101011	-	none
Read Status Char 0	RD0	11101100	-	none
Read Status Char 1	RD1	11101101	-	none
Read Status Char 2	RD2	11101110	-	none
Read Status Char 3	RD3	11101111	-	none
Clear Both	CLB	11110000	-	none
Clear Carry	CLC	11110001	-	none
Increment Accumulator	IAC	11110010	-	none
Complement Carry	CMC	11110011	-	none
Complement	CMA	11110100	-	none
Rotate Left	RAL	11110101	-	none
Rotate Right	RAR	11110110	-	none
Transfer Carry and Clear	TCC	11110111	-	none
Decrement Accumulator	DAC	11111000	-	none
Transfer Carry Subtract	TCS	11111001	-	none
Set Carry	STC	11111010	-	none
Decimal Adjust Accumulator	DAA	11111011	-	none
Keybord Process	KBP	11111100	-	none
Designate Command Line	DCL	11111101	-	none

http://e4004.szyc.org/iset.html

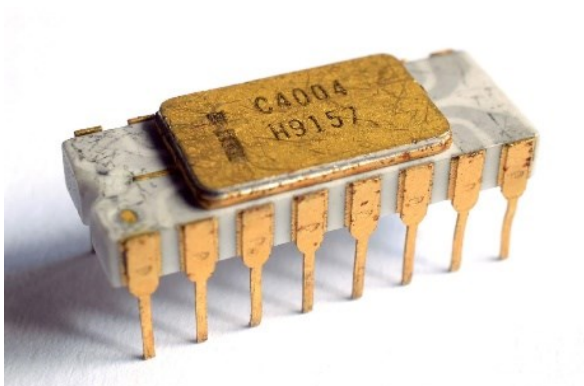
https://en.wikipedia.org/wiki/Intel_4004

Instruction Set Architecture Wars

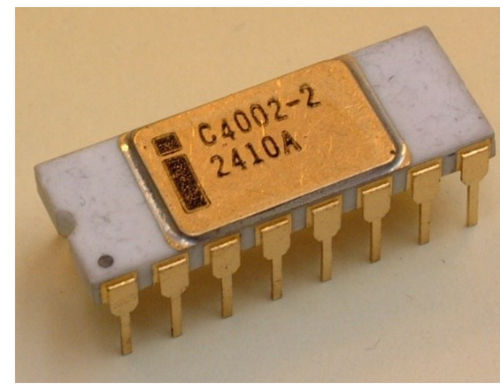
The first general-purpose CPU

How to add data stored in memory

CPU



Memory



Read data

Execute add operation

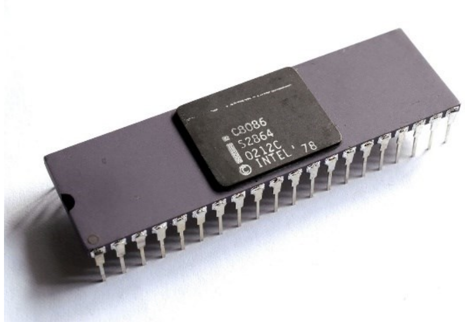
Write data back

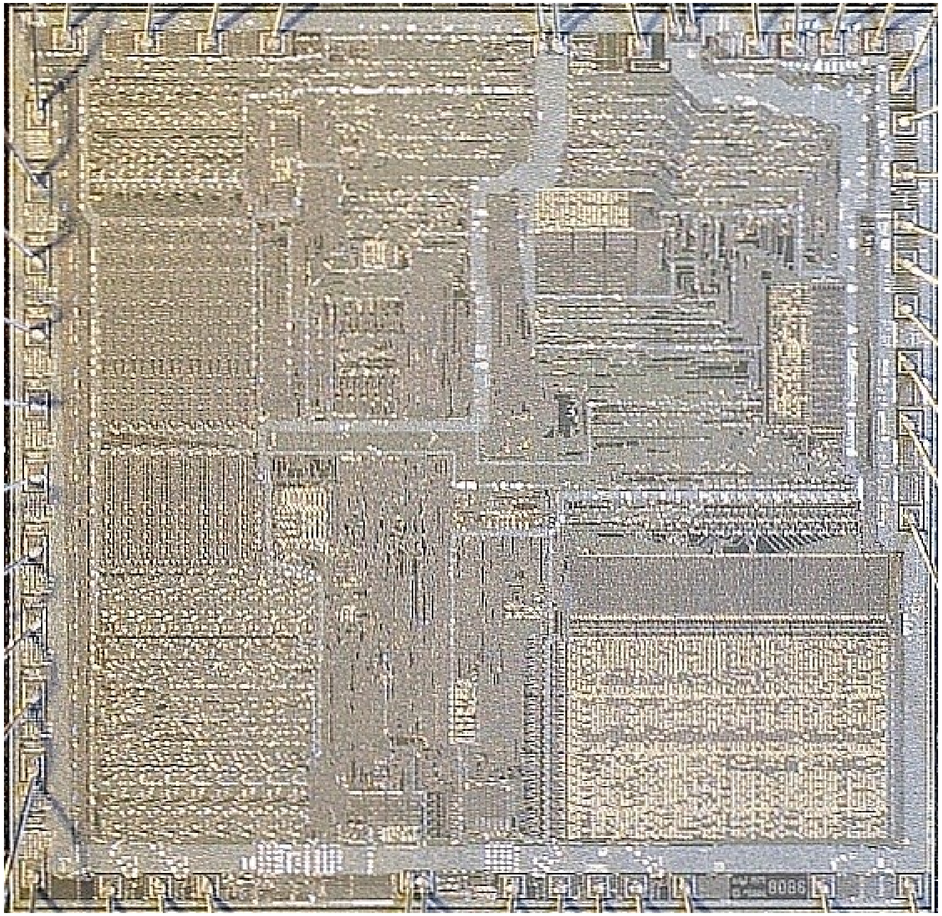
Slow and very difficult to program

https://en.wikipedia.org/wiki/Intel_4004

Instruction Set Architecture Wars

Intel8086



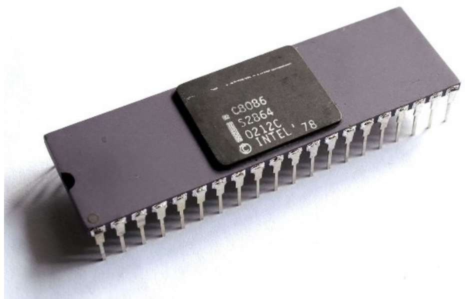


Intel8086
1979
16 bits Data width
29000 Transistors
154 Instructions (mnemonic)
5MHz – 10MHz

https://en.wikipedia.org/wiki/Intel_8086

Instruction Set Architecture Wars

Intel8086



Intel8086
1979
16 bits Data width
29000 Transistors
154 Instructions (mnemonic)
5MHz – 10MHz

The operation of ADD mnemonic

ADD - Arithmetic Addition

mnemonics	op	xx	xx	xx	xx	xx	sw	len	flags
ADD AL,ib	04	i0					B	2	o---szap
ADD AX,iw	05	i0	i1				W	3	o---szap
ADD rb,rmb	02	mr	d0	d1			B	2~4	o---szap
ADD rw,rmw	03	mr	d0	d1			W	2~4	o---szap
ADD rmb,ib	80	/0	d0	d1	i0		NB	3~5	o---szap
ADD rmw,iw	81	/0	d0	d1	i0	i1	NW	4~6	o---szap
ADD rmw,ib	83	/0	d0	d1	i0		EW	3~5	o---szap
ADD rmb,rb	00	mr	d0	d1			B	2~4	o---szap
ADD rmw,rw	01	mr	d0	d1			W	2~4	o---szap

Usage

```
ADD     dest,src
```

Modifies flags

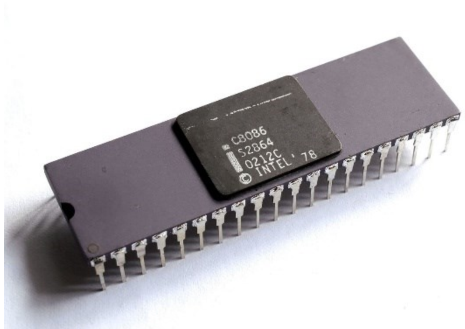
AF CF OF PF SF ZF

Adds "src" to "dest" and replacing the original contents of "dest". Both operands are binary.

https://en.wikipedia.org/wiki/Intel_8086

Instruction Set Architecture Wars

Intel8086



The one instruction is doing many operation

The complex operation was done in a single instruction

Program is simple to write
Fewer instruction = Program run faster

Complex Instruction Set Computer

https://en.wikipedia.org/wiki/Intel_8086

Instruction Set Architecture Wars

x86 integer instructions [edit]

This is the full 8086/8088 instruction set of Intel. Most if not all of these instructions are available in 32-bit mode; they just operate on 32-bit registers (**eax**, **ebx**, etc.) and values instead of their 16-bit (**ax**, **bx**, etc.) counterparts. See also x86 assembly language for a quick tutorial for this processor family. The updated instruction set is also grouped according to architecture (i386, i486, i686) and more generally is referred to as x86 32 and x86 64 (also known as AMD64).

Original 8086/8088 instructions [edit]

Original 8086/8088 instruction set

Instruction	Meaning	Notes	Opcode
AAA	ASCII adjust AL after addition	used with unpacked binary coded decimal	0x37
AAD	ASCII adjust AX before division	8086/8088 datasheet documents only base 10 version of the AAD instruction (opcode 0xD5 0x0A), but any other base will work. Later Intel's documentation has the generic form too. NEC V20 and V30 (and possibly other NEC V-series CPUs) always use base 10, and ignore the argument, causing a number of incompatibilities	0xD5
AAM	ASCII adjust AX after multiplication	Only base 10 version (Operand is 0xA) is documented, see notes for AAD	0xD4
AAS	ASCII adjust AL after subtraction		0x3F
ADC	Add with carry	`destination := destination + source + carry_flag`	0x10…0x15, 0x80/2…0x83/2
ADD	Add	(1) `r/m += r/imm;` (2) `r += m/imm;`	0x00…0x05, 0x80/0…0x83/0
AND	Logical AND	(1) `r/m &= r/imm;` (2) `r &= m/imm;`	0x20…0x25, 0x80/4…0x83/4
CALL	Call procedure	`push eip; eip points to the instruction directly after the call`	0x9A, 0xE8, 0xFF/2, 0xFF/3
CBW	Convert byte to word		0x98

Instruction Set Architecture Wars



- Process is too complex
- Expensive
- More energy requirement
- Not all instruction is used

Instruction Set Architecture Wars

John L. Hennessy

RISC Instruction Set Architecture

Reduced Instruction Set Computer

MIPS (Microprocessor without Interlocked Pipelined Stages) Processors

1985

MIPS Instruction Reference

This is a description of the MIPS instruction set, their meanings, syntax, semantics, and bit encodings. The syntax given for each instruction refers to the assembly language syntax supported by the MIPS assembler. Hyphens in the encoding indicate "don't care" bits which are not considered when an instruction is being decoded.

General purpose registers (GPRs) are indicated with a dollar sign ($). The words SWORD and UWORD refer to 32-bit signed and 32-bit unsigned data types, respectively.

The manner in which the processor executes an instruction and advances its program counters is as follows:

1. execute the instruction at *PC*
2. copy *nPC* to *PC*
3. add 4 or the branch offset to *nPC*

This behavior is indicated in the instruction specifications below. For brevity, the function advance_pc (int) is used in many of the instruction descriptions. This function is defined as follows:

```
void advance_pc (SWORD offset)
{
    PC  =  nPC;
   nPC  += offset;
}
```

Note: ALL arithmetic immediate values are sign-extended. After that, they are handled as signed or unsigned 32 bit numbers, depending upon the instruction. The only difference between signed and unsigned instructions is that signed instructions can generate an overflow exception and unsigned instructions can not.

The instruction descriptions are given below:

ADD – ***Add (with overflow)***

Description:	Adds two registers and stores the result in a register
Operation:	$d = $s + $t; advance_pc (4);
Syntax:	add $d, $s, $t

Instruction Set Architecture Wars

PC in 80's – 90's

Assembly / C

BASIC

Instruction Set Architecture Wars

Pros

CISC	RISC
Easy to program (in assembly) Fast memory access Small code size	Low-cost Low power consumption Single cycle instruction

Cons

CISC	RISC
Expensive High power consumption Instruction can take several cycles	Large code size Ram Bottleneck Hard to program (in assembly)

Who still program in Assembly ?

Smart compiler can overcome this problem!

Instruction Set Architecture Wars

RISC Instruction Set Architecture

Acorn Computers Ltd.

BBC Micro

Archimedes 400/1 series computer

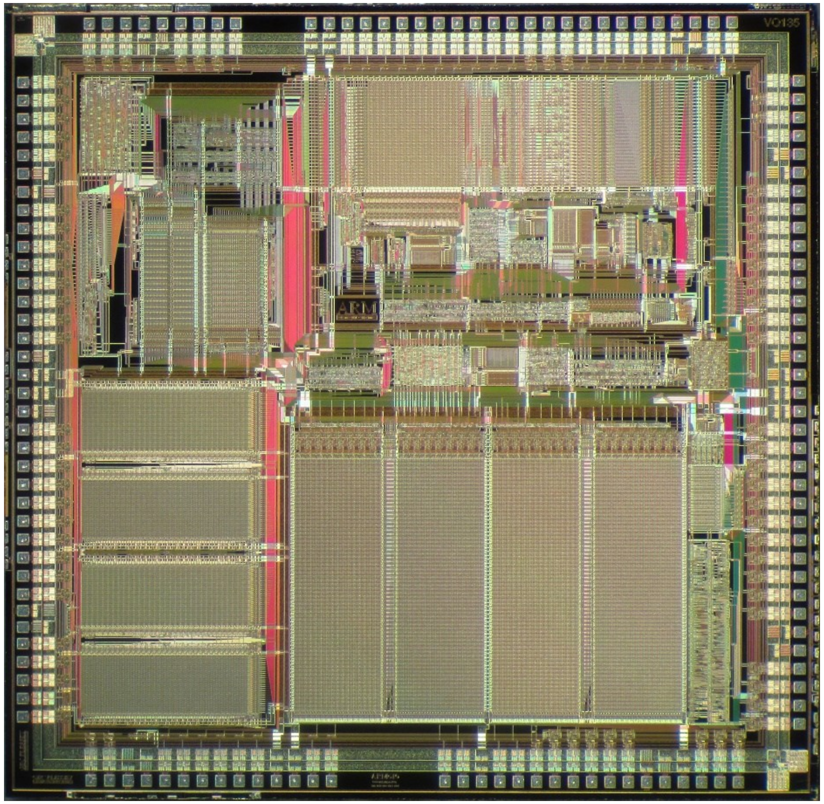
Acorn RISC Machine

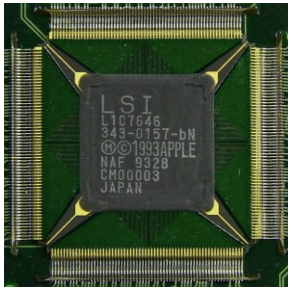
- The official *Acorn RISC Machine* project started in October 1983.
- VLSI Technology as the *silicon partner*
- The first samples of ARM silicon worked properly when first received and tested on 26 April 1985

Instruction Set Architecture Wars

Acorn RISC Machine

Advanced RISC Machine





Apple Newton was based on the ARM 610 RISC processor

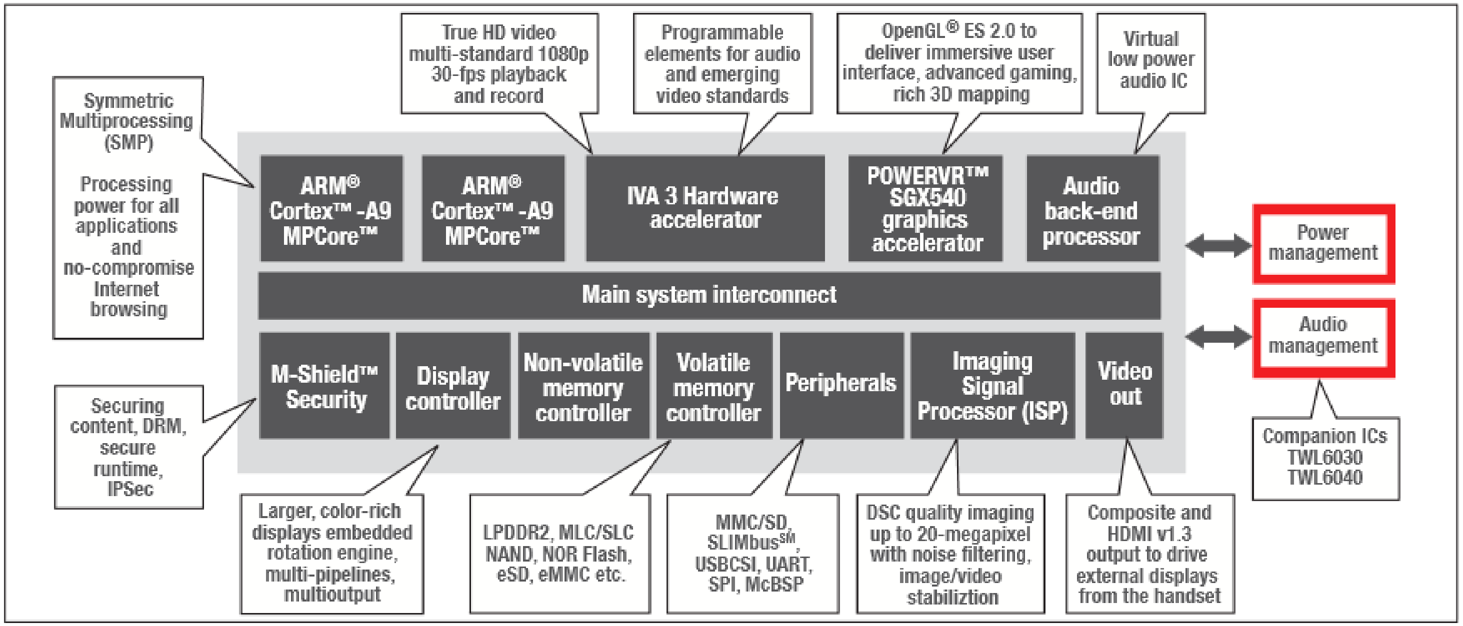
RISC CPU

MIPS TECHNOLOGIES

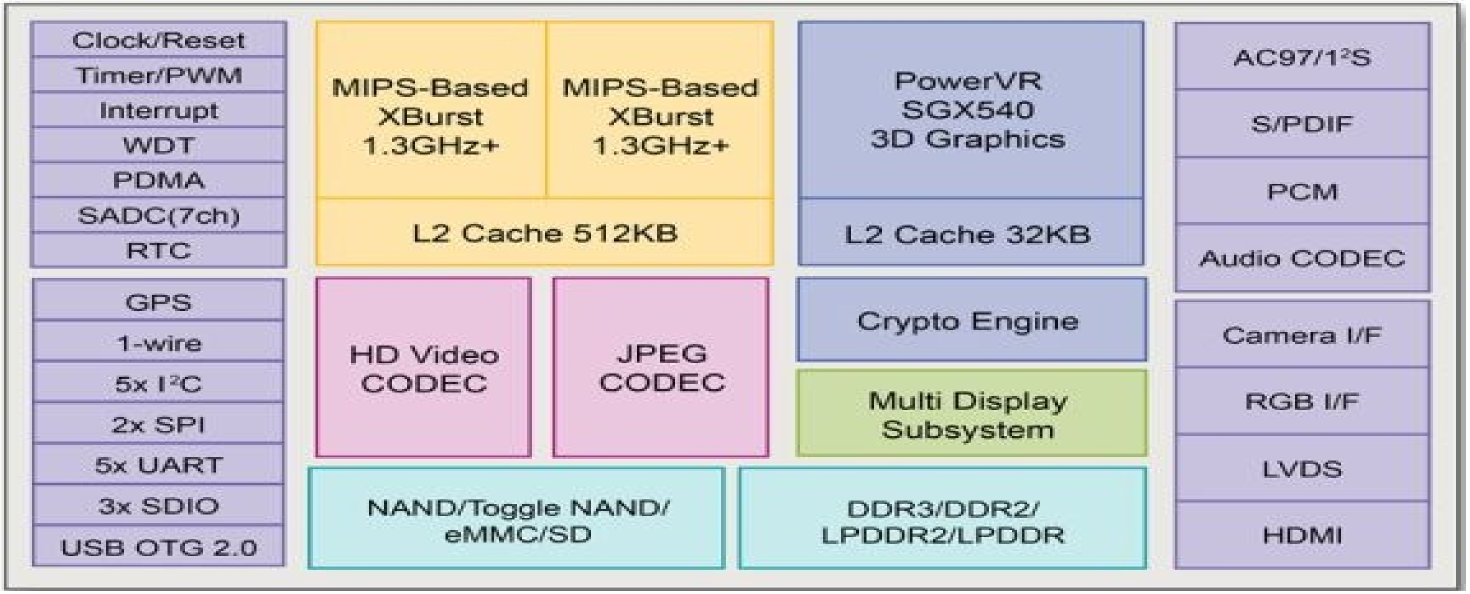
ARM®

RISC-V

MIPS & ARM



ARM

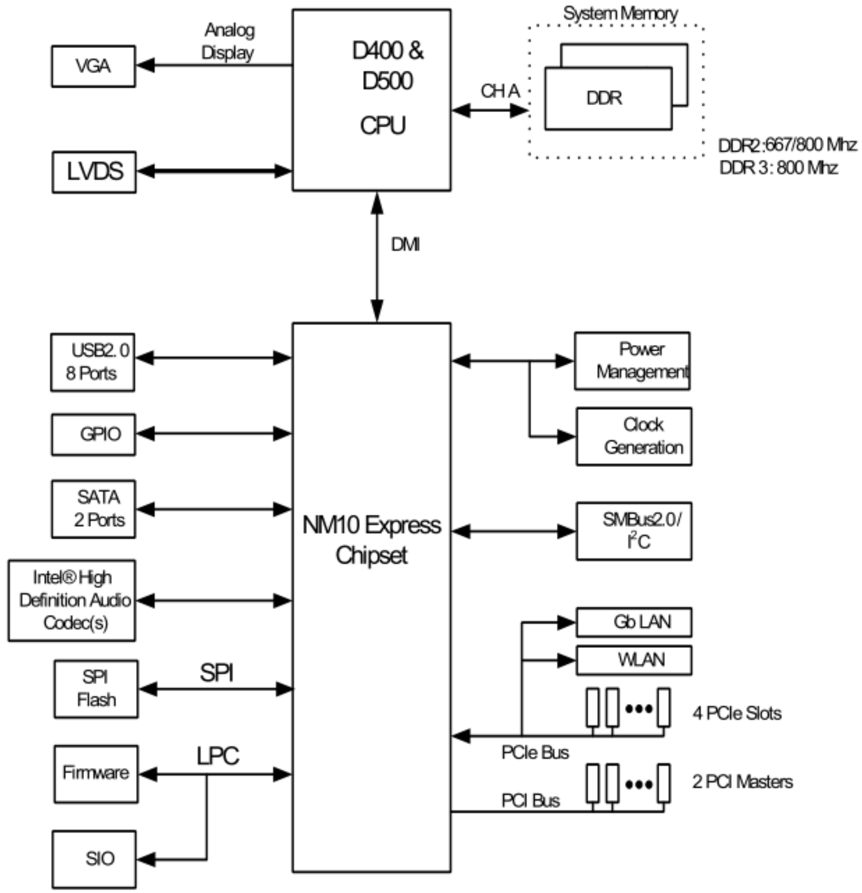


MIPS

MIPS & ARM

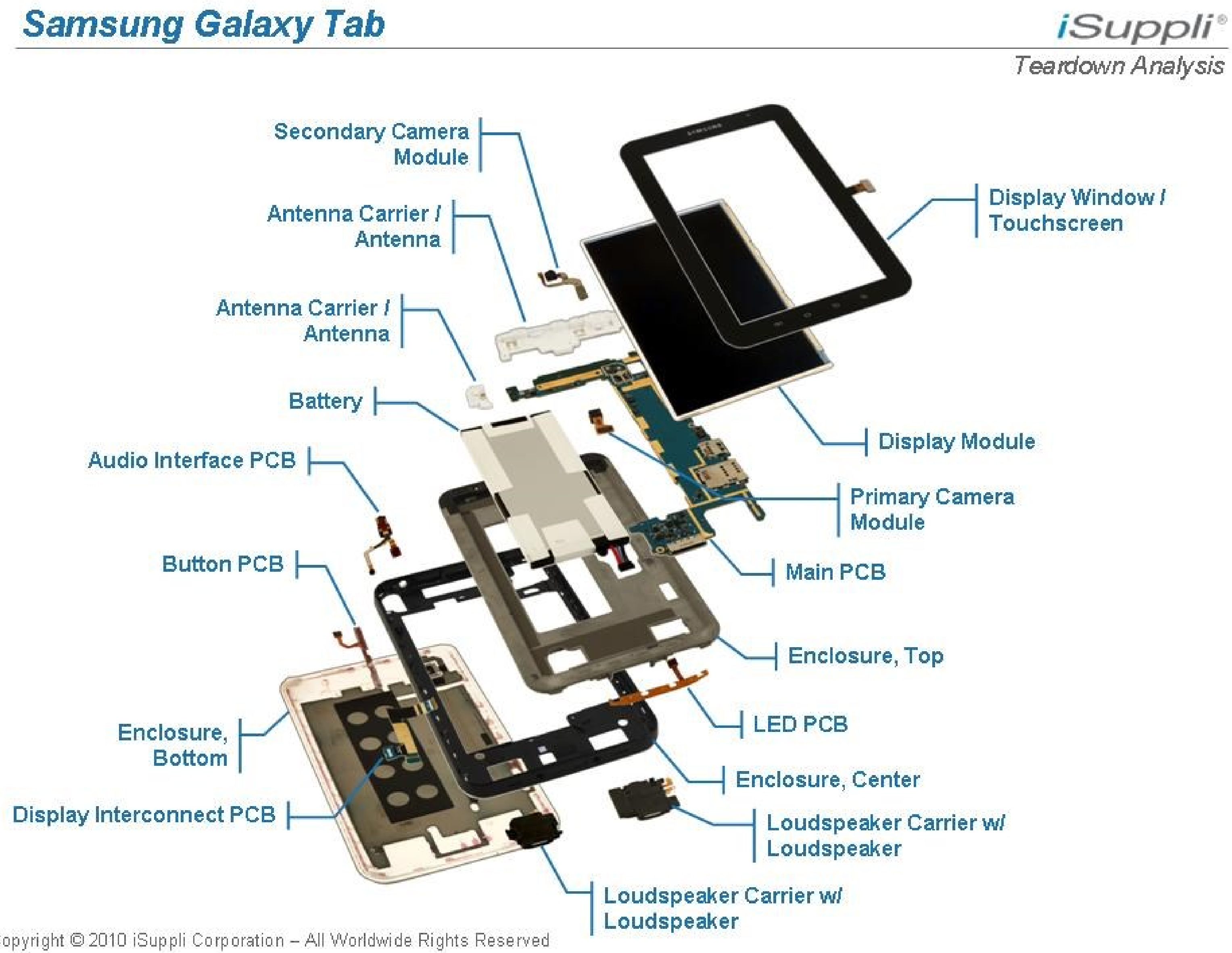
MIPS (Microprocessor without Interlocked Pipeline Stages)	ARM (Advanced RISC Machines)
Introduced 1981	Introduced 1985
John L. Hennessy at Stanford University	ARM Holdings
Reduced Instruction Set Computer (RISC)	
32 Registers **hard-wired-to-zero Register ($0)**	16 Registers Program Counter as a GPR
Compare only bew < , > need special instruction to set flag	cmp with condition flags (x86-style) If then else style
Printer, Set top box, Router…	Cell phone, Tablet

ATOM

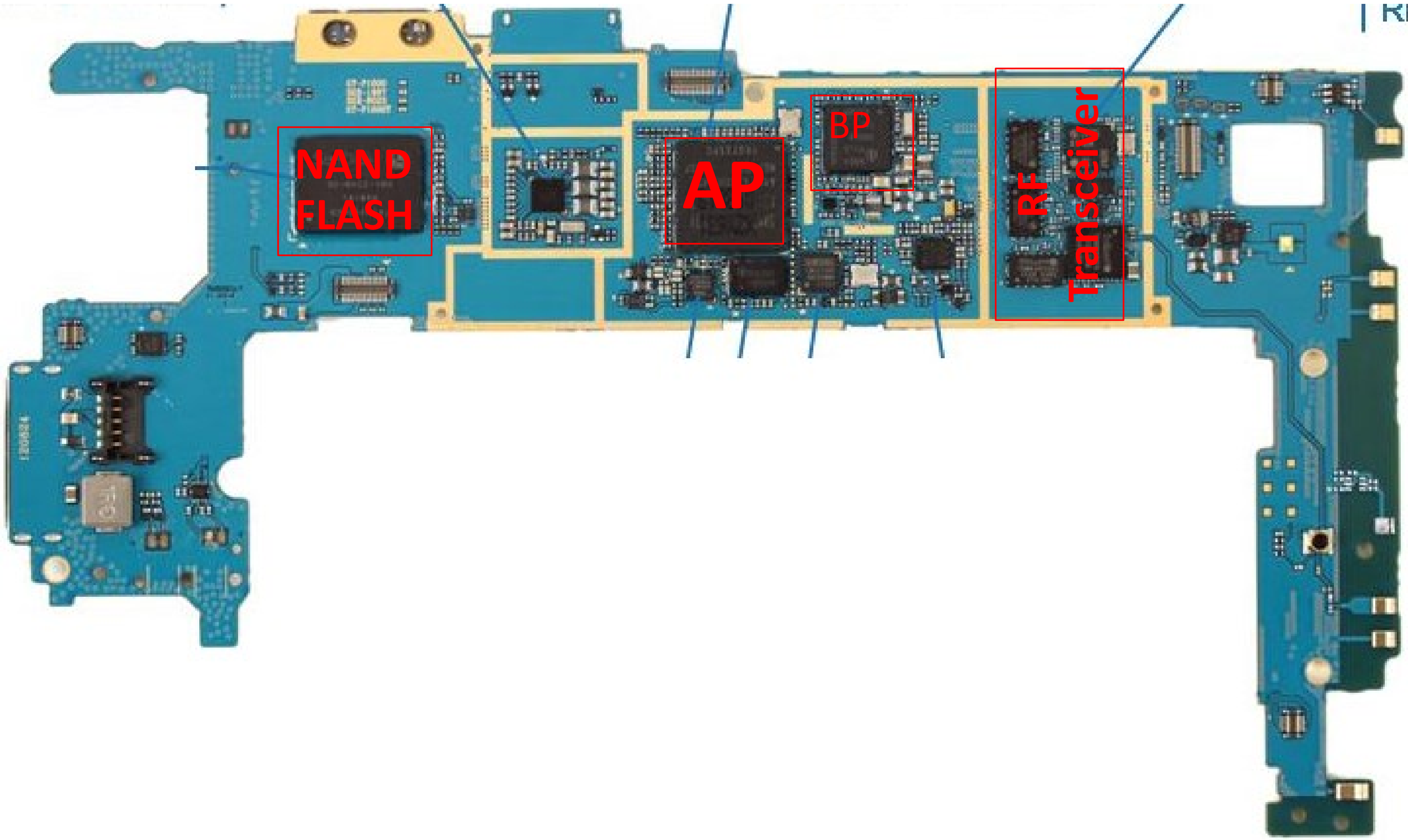


CISC, complex, expensive, need more energy
X64, PC computable and it’s run Windows!

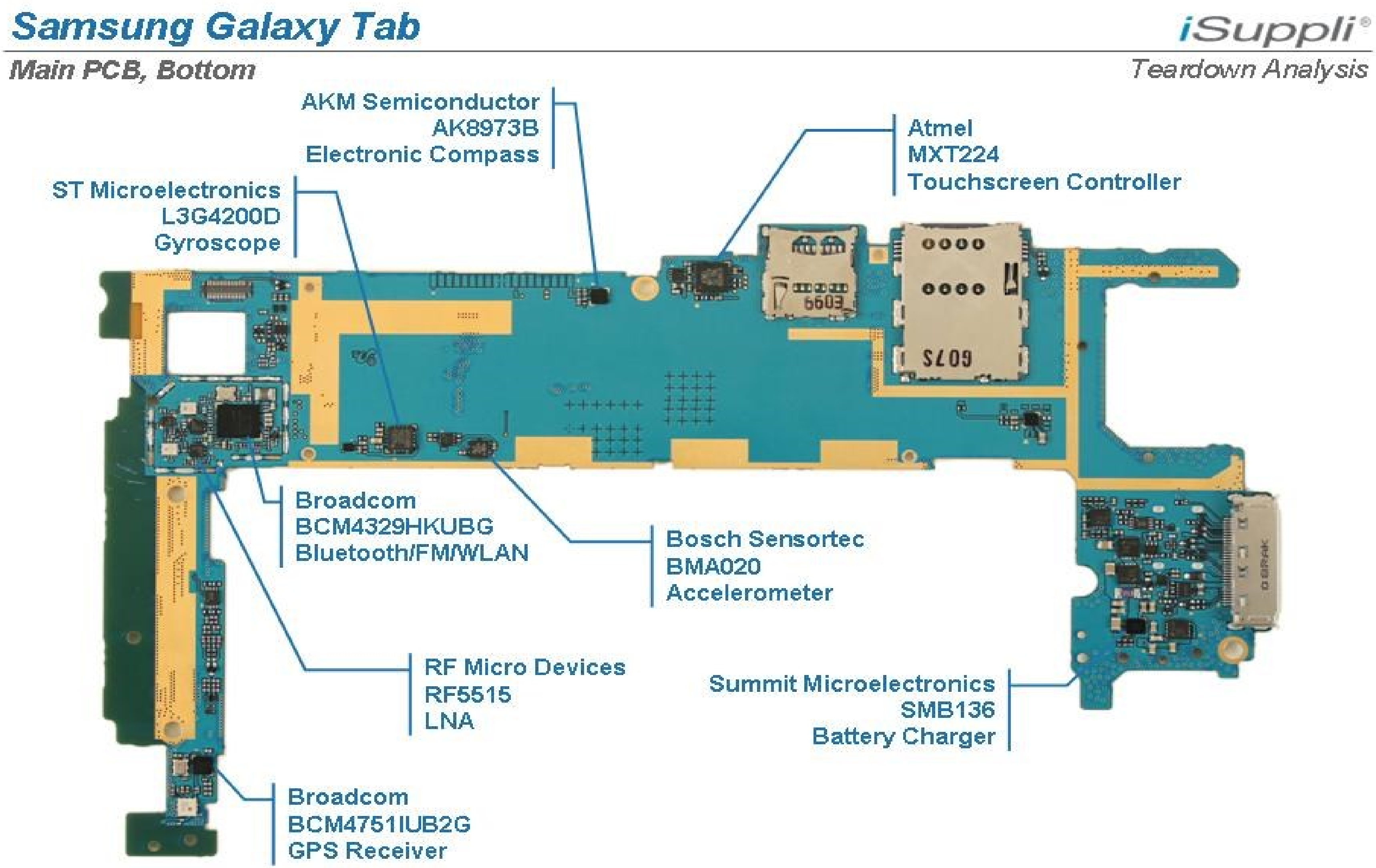
GALAXY TAB


Samsung Galaxy Tab
iSuppli
Teardown Analysis
Secondary Camera Module
Antenna Carrier / Antenna
Antenna Carrier / Antenna
Battery
Audio Interface PCB
Button PCB
Enclosure, Bottom
Display Interconnect PCB
Display Window / Touchscreen
Display Module
Primary Camera Module
Main PCB
Enclosure, Top
LED PCB
Enclosure, Center
Loudspeaker Carrier w/ Loudspeaker
Loudspeaker Carrier w/ Loudspeaker
Copyright © 2010 iSuppli Corporation – All Worldwide Rights Reserved

GALAXY TAB



GALAXY TAB



Copyright © 2010 iSuppli Corporation – All Worldwide Rights Reserved

Conclusion

-Smart phone
- 2 CPU (BP,AP)
- 2 OS (Baseband RTOS, Application OS)

-Feature phone
-Application run on Baseband processor
-No Application operating system

-Generation
-1G Analog
-2G Digital
-3G Faster Digital
-4G Faster Digital without voice channel