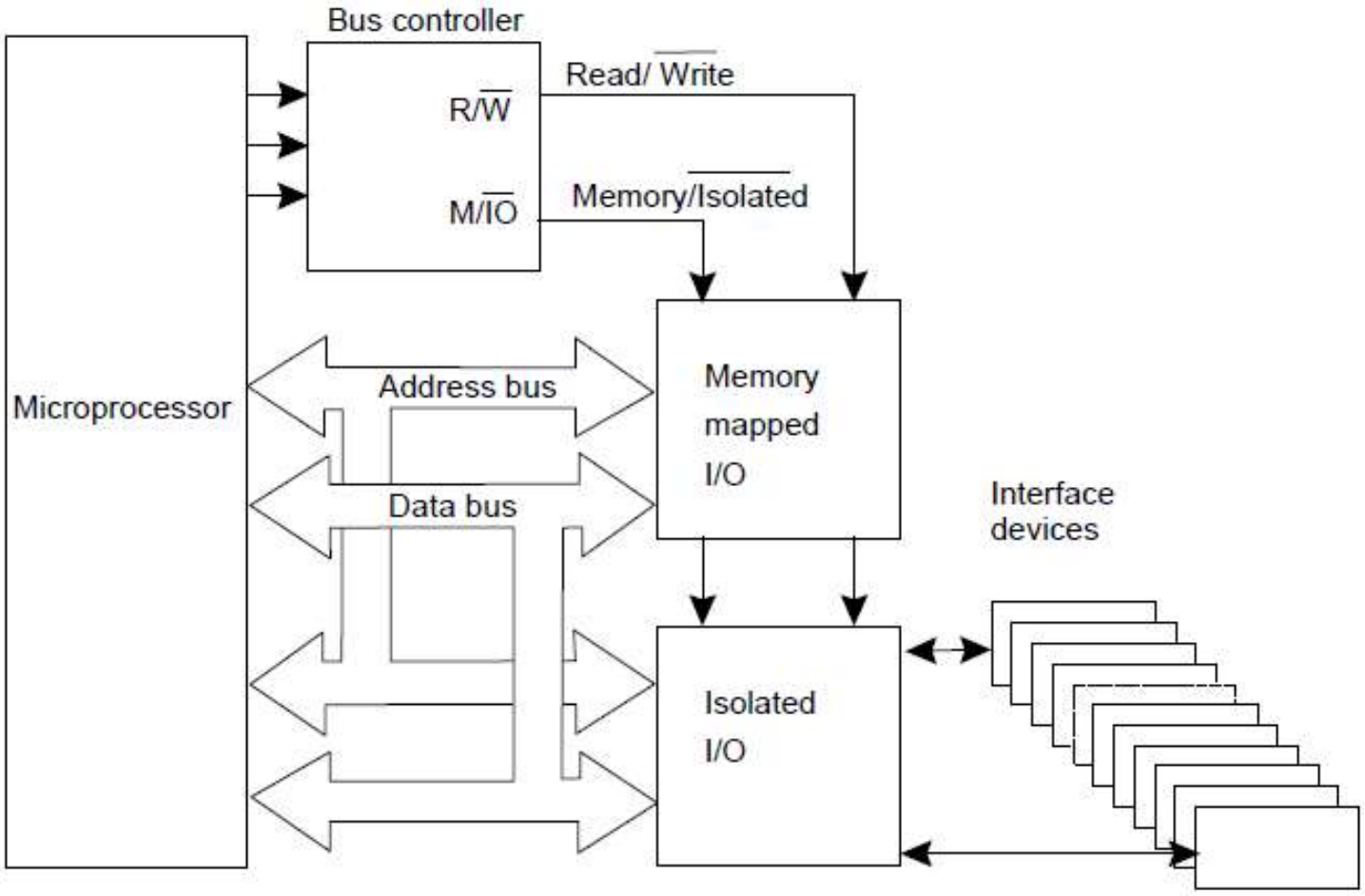
Access memory mapped and isolated I/O



Access memory mapped and isolated I/O

Memory mapped I/O

- Interface devices can map directly onto the system address and data bus.
- In a PC-compatible system the address bus is 20 bits wide, from address 00000h to FFFFFh (1 MB).
  - If the PC is being used in an enhanced mode (such as with MS Windows) it can access the area of memory above 1 MB.
  - If it uses 16-bit software (such as Microsoft Windows 3.1) then it can address up to 16 MB of physical memory, from 000000h to FFFFFFh.
  - If it uses 32-bit software (such as MS Windows 95/98/NT/ 2000) then the software can address up to 4 GB of physical memory, from 00000000h to FFFFFFFFh.

Processor Buses

- Like the CPU buses, the processor's buses interconnect the processor's major internal components (in this case the CPU, memory and I/O) together, carrying signals between the different components.
- A key feature of processor buses is their **width** which is (the number of bits that can be transmitted at any one time).
  - This can vary depending on both the buses implemented within the processor—for example: x86 contains bus widths of 16/32/64, 68K has 8/16/32/ 64 bit buses, MIPS 32 has 32 bit buses, and so forth.
- Each bus also has a **bus speed** (in MHz) that impacts the performance of the processor.
  - Buses implemented in real-world processor designs include the U, peripheral, and CPM buses in the MPC8xx family of processors, and the C and X buses in the x86 Geode.

Results

Specs/performance of motherboard bus types				
Bus type	**Frequency (MHz)**	**Bus-width (bits)**	**Clock-doubling technology**	**Theoretical bandwidth (MBytes per sec)**
ISA	8 (1)	16	N	8
PCI version 2.1	33	32	N	133
PCI-X	66/100/133	64	N	533/800/1066
PCI-X 2.0	133	64	4x	4,266
AGP (first implementation)	66	32	2x	266
AGP 8x	66	32	8x	2,133
PCI-Express single-lane	2,500	8	N	250 (2)
PCI-Express 4-lane	2,500	8	N	1,000 (2)
PCI-Express 8-lane	2,500	8	N	2,000 (2)
PCI-Express 16-lane	2,500	8	N	4,000 (2)
	(1) Some had synchronous frequencies up to 11MHz			(2) In each direction

Processor Performance

- There are several measures of processor performance, but are all based upon the processor's behavior over a given length of time.
- One of the most common definitions of processor performance is a **processor's throughput**, the amount of work the CPU completes in a given period of time.
- A processor's execution is ultimately synchronized by an external system or master clock, located on the board.
  - The master clock is simply an oscillator producing a fixed frequency sequence of regular on/off pulse signals that is usually divided or multiplied within the CPU's CU (control unit) to generate at least one internal clock signal running at a constant number of clock cycles per second, or clock rate, to control and coordinate the fetching, decoding, and execution of instructions.
  - The CPU's clock rate is expressed in MHz (megahertz).

Processor Performance

- At this point the total CPU's execution time can be determined by:
  - *CPU execution time in seconds per program = (total number of instructions per program or instruction count) * (CPI in number of cycle cycles/instruction) * (clock period in seconds per cycle) = ((instruction count) * (CPI in number of cycle cycles/instruction)) / (clock rate in MHz)*
- The processor's average execution rate, also referred to as **throughput** or **bandwidth**, reflects the amount of work the CPU does in a period of time and is the inverse of the CPU's execution time:
- *CPU throughput (in bytes/sec or MB/sec) = 1 / CPU execution time = CPU performance*

Other definitions of performance

- Other definitions of performance besides throughput include a processor's
  - responsiveness, or ***latency, which is the length of elapsed time a processor*** takes to respond to some event.
  - ***availability, which is the amount of time the processor runs normally*** without failure;
  - ***reliability, the average time between failures or MTBF (mean time*** between failures);
  - ***recoverability, the average time the CPU takes to recover from*** failure or MTTR (mean time to recover).

Final Note

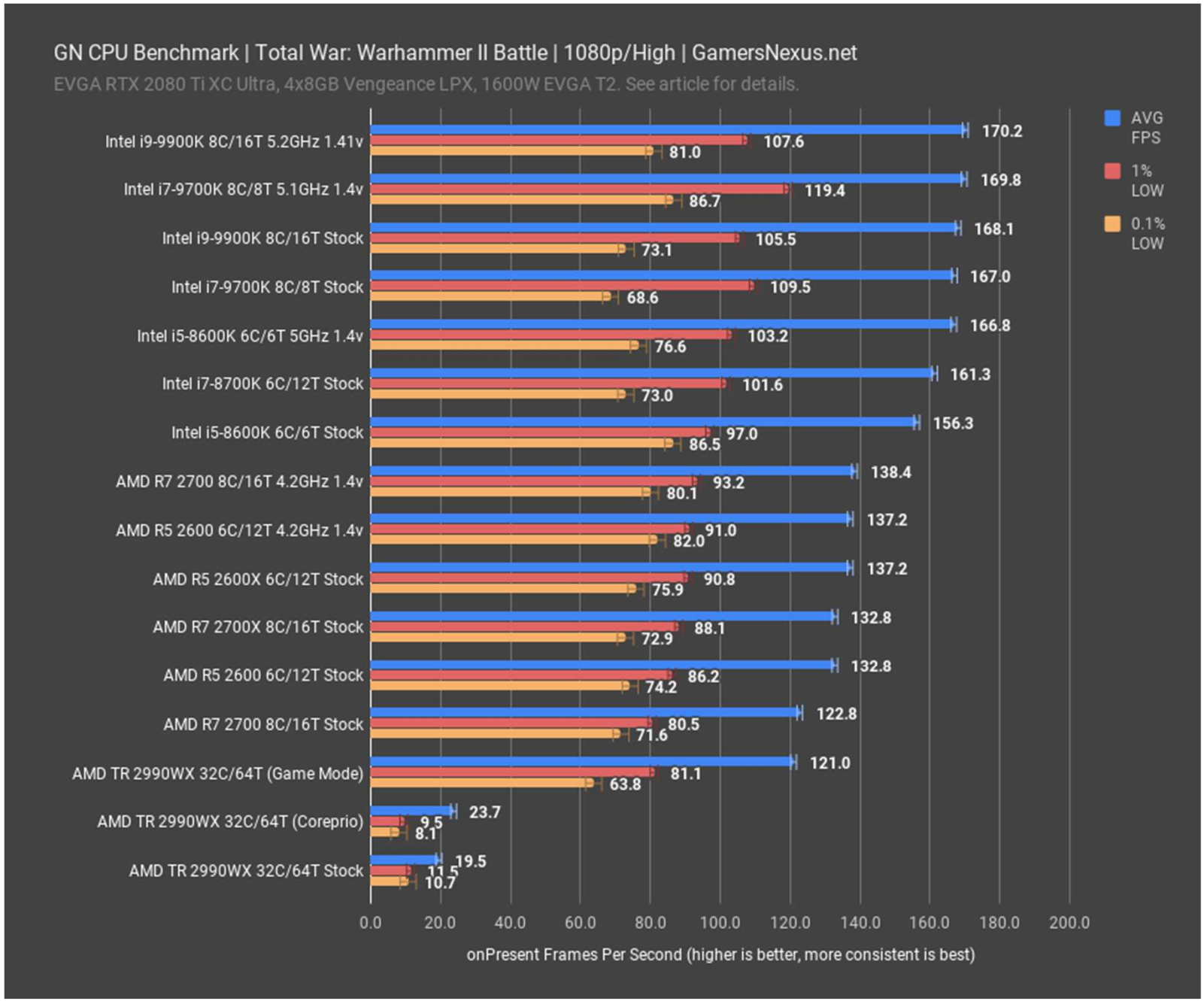
- A processor's internal design determines a processor's clock rate and the CPI; thus a processor's performance depends on which ISA is implemented and how the ISA is implemented.
  - For example, architectures that implement Instruction-level Parallelism ISA models have better performance over the application-specific and general-purpose based processors because of the parallelism that occurs within these architectures.
- Performance can be improved because of the actual physical implementations of the ISA within the processor, such as implementing pipelining in the ALU.
- The increasing gap between the performance of processors and memory can be improved by cache algorithms that implement instruction and data prefetching (especially algorithms that make use of branch prediction to reduce stall time), and lockup-free caching.
- Basically, any design feature that allows for either an increase in the clock rate or decrease in the CPI will increase the overall performance of a processor.

Benchmarks

- One of the most common performance measures used for processors in the embedded market is millions of instructions per seconds or **MIPS**.
  - MIPS = Instruction Count / (CPU execution time * 10^6) = Clock Rate / (CPI * 10^6)

MIPS

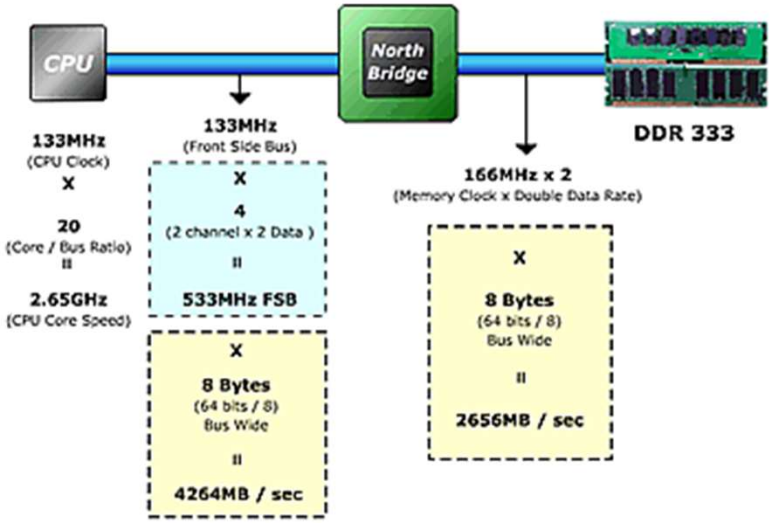
- The MIPS performance measure gives the impression that faster processors have higher MIPS values, since part of the MIPS formula is inversely proportional to the CPU's execution time.
- However, MIPS can be misleading when making this assumption for a number of reasons, including:
  - Instruction complexity and functionality aren't taken into consideration in the MIPS formula, so MIPS cannot compare the capabilities of processors with different ISAs.
  - MIPS can vary on the same processor when running different programs (with varying instruction count and different types of instructions).


GN CPU Benchmark | Total War: Warhammer II Battle | 1080p/High | GamersNexus.net
EVGA RTX 2080 Ti XC Ultra, 4x8GB Vengeance LPX, 1600W EVGA T2. See article for details.
AVG FPS
1% LOW
0.1% LOW
Intel i9-9900K 8C/16T 5.2GHz 1.41v
170.2
107.6
81.0
Intel i7-9700K 8C/8T 5.1GHz 1.4v
169.8
119.4
86.7
Intel i9-9900K 8C/16T Stock
168.1
105.5
73.1
Intel i7-9700K 8C/8T Stock
167.0
109.5
68.6
Intel i5-8600K 6C/6T 5GHz 1.4v
166.8
103.2
76.6
Intel i7-8700K 6C/12T Stock
161.3
101.6
73.0
Intel i5-8600K 6C/6T Stock
156.3
97.0
86.5
AMD R7 2700 8C/16T 4.2GHz 1.4v
138.4
93.2
80.1
AMD R5 2600 6C/12T 4.2GHz 1.4v
137.2
91.0
82.0
AMD R5 2600X 6C/12T Stock
137.2
90.8
75.9
AMD R7 2700X 8C/16T Stock
132.8
88.1
72.9
AMD R5 2600 6C/12T Stock
132.8
86.2
74.2
AMD R7 2700 8C/16T Stock
122.8
80.5
71.6
AMD TR 2990WX 32C/64T (Game Mode)
121.0
81.1
63.8
AMD TR 2990WX 32C/64T (Coreprio)
23.7
9.5
8.1
AMD TR 2990WX 32C/64T Stock
19.5
11.5
10.7
0.0
20.0
40.0
60.0
80.0
100.0
120.0
140.0
160.0
180.0
200.0
onPresent Frames Per Second (higher is better, more consistent is best)

Reading a Processor's Datasheet

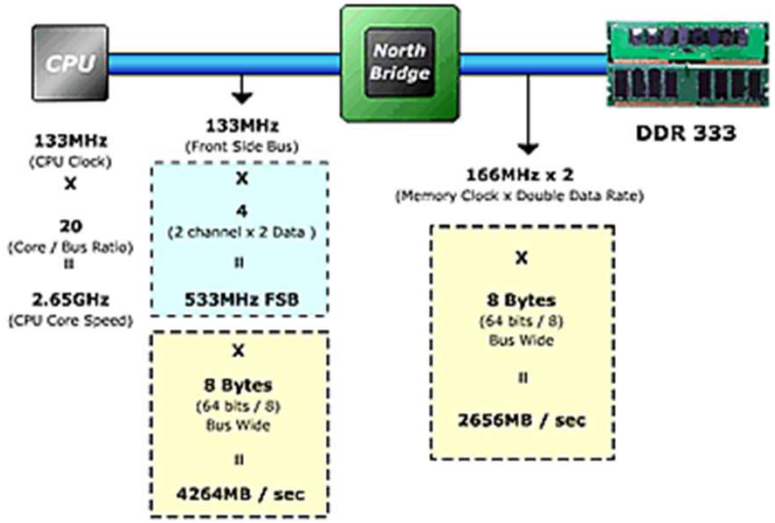
- A processor's datasheet provides key areas of useful processor information.
- Datasheets exist for almost any component, both hardware and software, and the information they contain varies between vendors.
- Some datasheets are a couple of pages long and list only the main features of a system, while others contain over 100 pages of technical information.
  - the MPC860EC rev. 6.3 datasheet is 80 pages.

Case Study



- **CPU**
- CPU clock is 133MHz to communicate with other components,
  - the CPU core speed is actually 2.65GHz (133MHz x 20 Core/Bus Ratio).
- **Front Side Bus**
- Data speed between CPU and North Bridge use the CPU clock and is called Front Side Bus (FSB).
  - FSB should equal to CPU clock, but nowadays we read/hear about 400MHz/533 FSB those are actually the efficiency by dual channels and double data rate.
    - For example, the FSB is 133(MHz) x2 (dual channels) x (double data rate) = 533 MHz (virtual working efficiency).
- **Memory Speed and data speed**
- Memory speed is also measured in memory clock. DDR (Double Data Rate) will twice the data transaction.
  - Nowadays most of bus wide are 64 bits base that means the bus can path 8 Bytes data (64 bits/8 = 8 Bytes).
  - How to calculate data the memory can carry through in one second?
  - Taking an example of DDR 333 (which is actually 166MHz x 2) to multiply 8 Bytes (64bits' Bus) will get 2656MB/sec.

Case Study



- **Memory performance comparing with FSB**
- If we calculate the data traveling speed between North Bridge and CPU, we have to use the efficiency of FBS to multiply bus wide's data.
  - For example, the 133MHz CPU Clock (FSB is also 133MHz) which virtual working efficiency FSB between CPU and North Bridge is 533MHz (133MHz x 2 x2) multiplies 8 Bytes (64 Bits' bus) we will get 4264MB/sec.
  - Comparing the data speed of DDR333 can carry through is still far behind the 533MHz FSB can take. So, there always is room to improve the speed of memory.

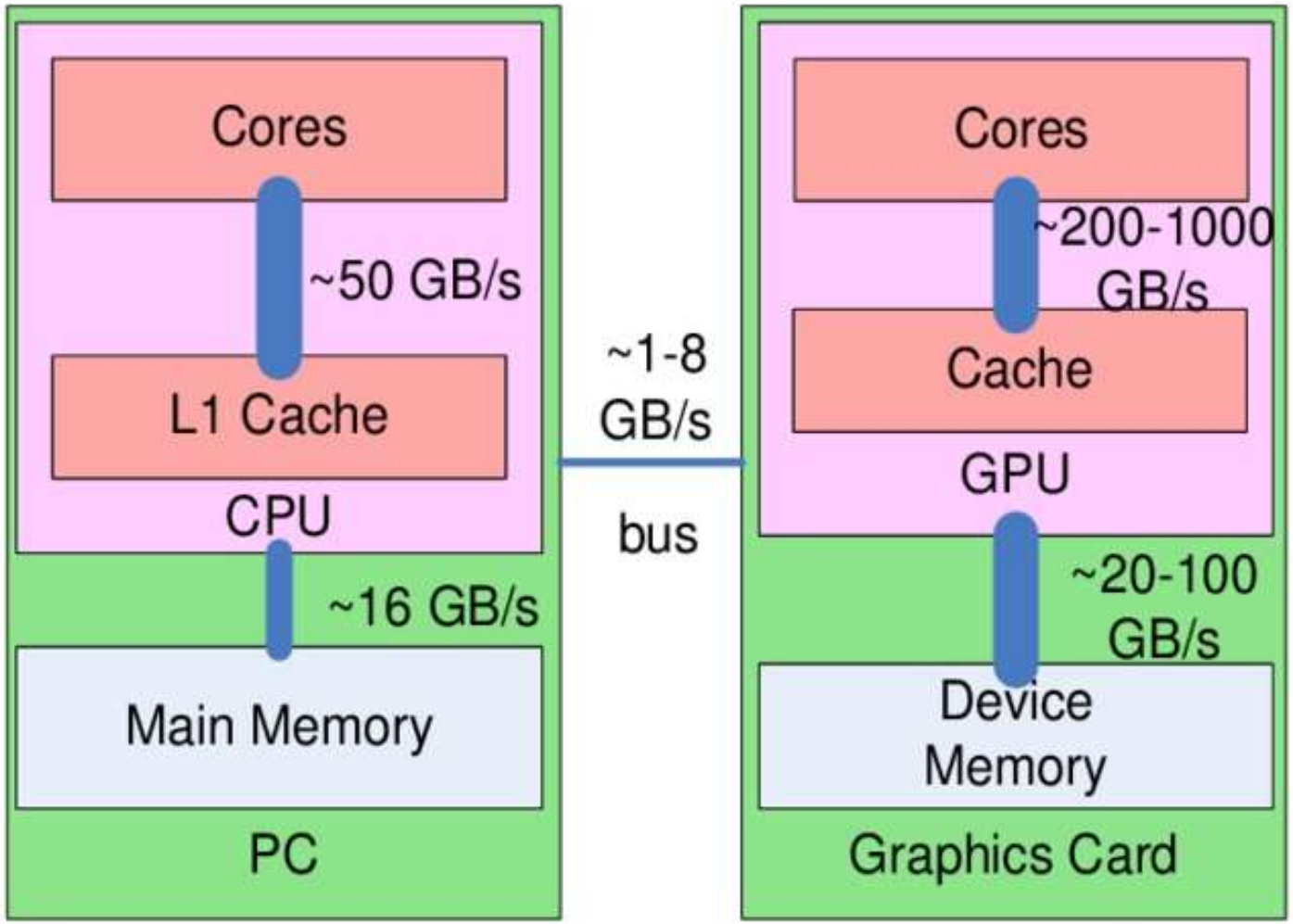
The Graphic Card

- powerful graphics accelerators have been introduced since the mid 1990s.
  - A graphic accelerator is the unit responsible for the 2D or 3D output
- the graphics unit usually is for the sake of interchangeability,
  - an add-in card
  - plugged into an expansion slot on the motherboard inside your PC.
- Low-cost graphics solutions are often integrated with the core components on a motherboard.
  - reserved for 2D graphics or displaying Windows or text

The Graphics Processor

- The graphics processor is very much the heart of the graphics card, just like the CPU is the brain of a computer.
- In most cases the graphics processor itself is hidden behind a cooler. The graphics processor is typically the largest - and hottest - component on the graphics card.
- The graphics processor is the single most important part of the graphics card. Almost all hardware specifications, such as pixel shaders, vertex shaders, pipelines, and component clock speeds refer to the architecture and capabilities of the graphics processor.
- The only other specifications of note are associated with the graphics card memory, which works hand in hand with the graphics processor to help deliver performance in bandwidth-intensive application such as 3D games.

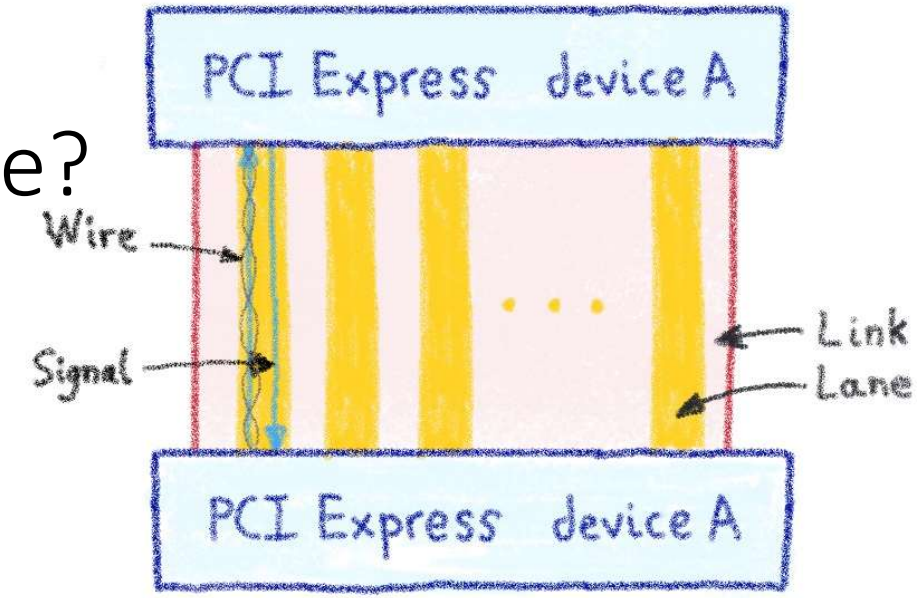
Various Speeds



PCI Express

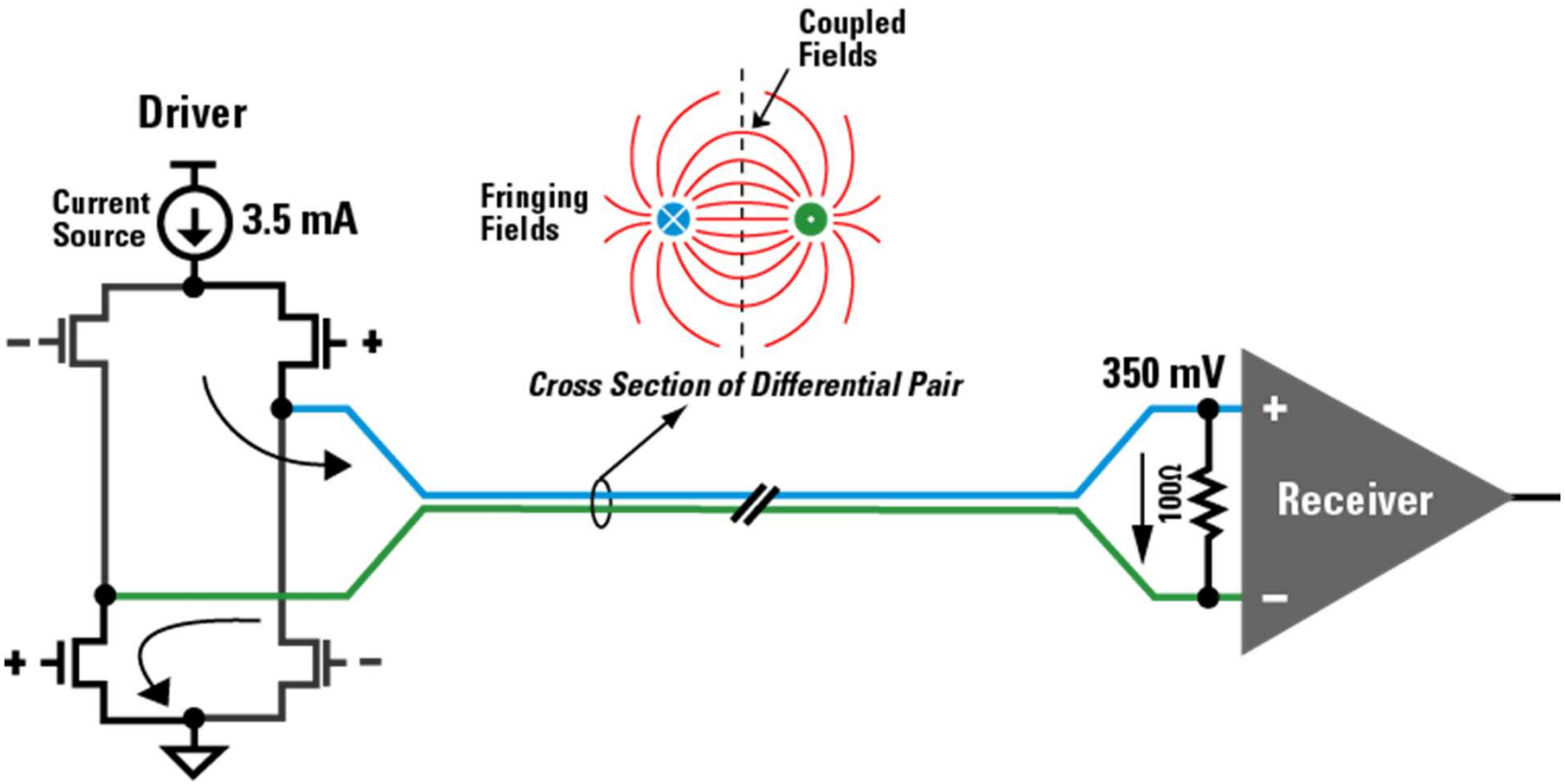
- In contrast to ISA, PCI and AGP, PCI Express is a serial interface subsystem.
  - it runs with very few connections.
  - Different from parallel buses, the total bandwidth is available for every device.
- PCI Express works on the basis of multiplying as many single links (or lanes) as required to lineup the desired bandwidth.
- PCI Express x16 (16 links) offers a bandwidth of 4 GB/s up and down or 8 GB/s total.
  - The inferior slot options (x8, x4, x1) are not used for graphics.
  - A mechanical x16 slot does not necessarily have to run at 16 connected PCI Express lanes, though.

What's a PCI Express lane?



- PCIe lanes are used to communicate between PCIe Devices or between PCIe and CPU.
- A lane is composed of 2 wires: one for inbound communications and one, which has double the traffic bandwidth, for outbound.
- Lane communications are similar to network Layer 1 communications
- However, the technique used for PCIe Link is a bit different as the PCIe device is composed of xN lanes. In our previous example N=16 but it could be any power of 2 from 1 to 16 (1/2/4/8/16).

Low-voltage differential signaling



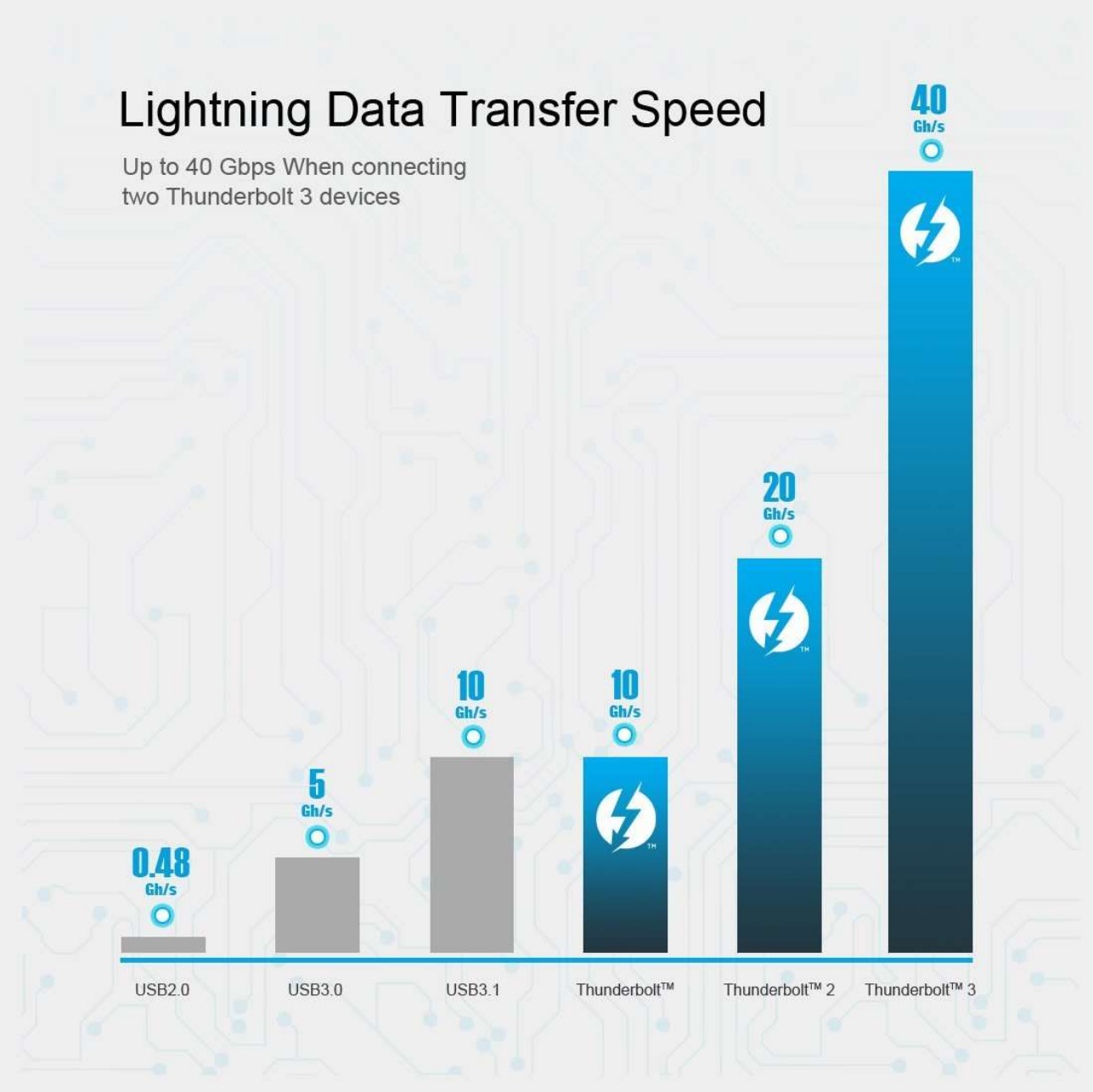
SATA

- Serial ATA is a new interface technology defined by Intel Corporation and backed by all major PC manufacturers, Hard disk manufacturers, and many leading companies in the computer industry.
- This technology gradually replaces the old (E)IDE interface.
  - 2003: first generation of SATA performs at 150 MBytes/sec (1.5 Gbits/sec).
  - 2004: second generation of SATA doubles the data throughput to 3 Gbits/sec.
  - 2008: performance doubled again to 6 Gbits/sec (to be verified).
- Serial ATA also improves the system air flow by utilizing a new thin cabling system to simplify the connection between the host controller and the data storage devices.
- To learn more about Serial ATA, please visit the Serial ATA Working Group's web site at www.serialata.org.

SATA - Advantages

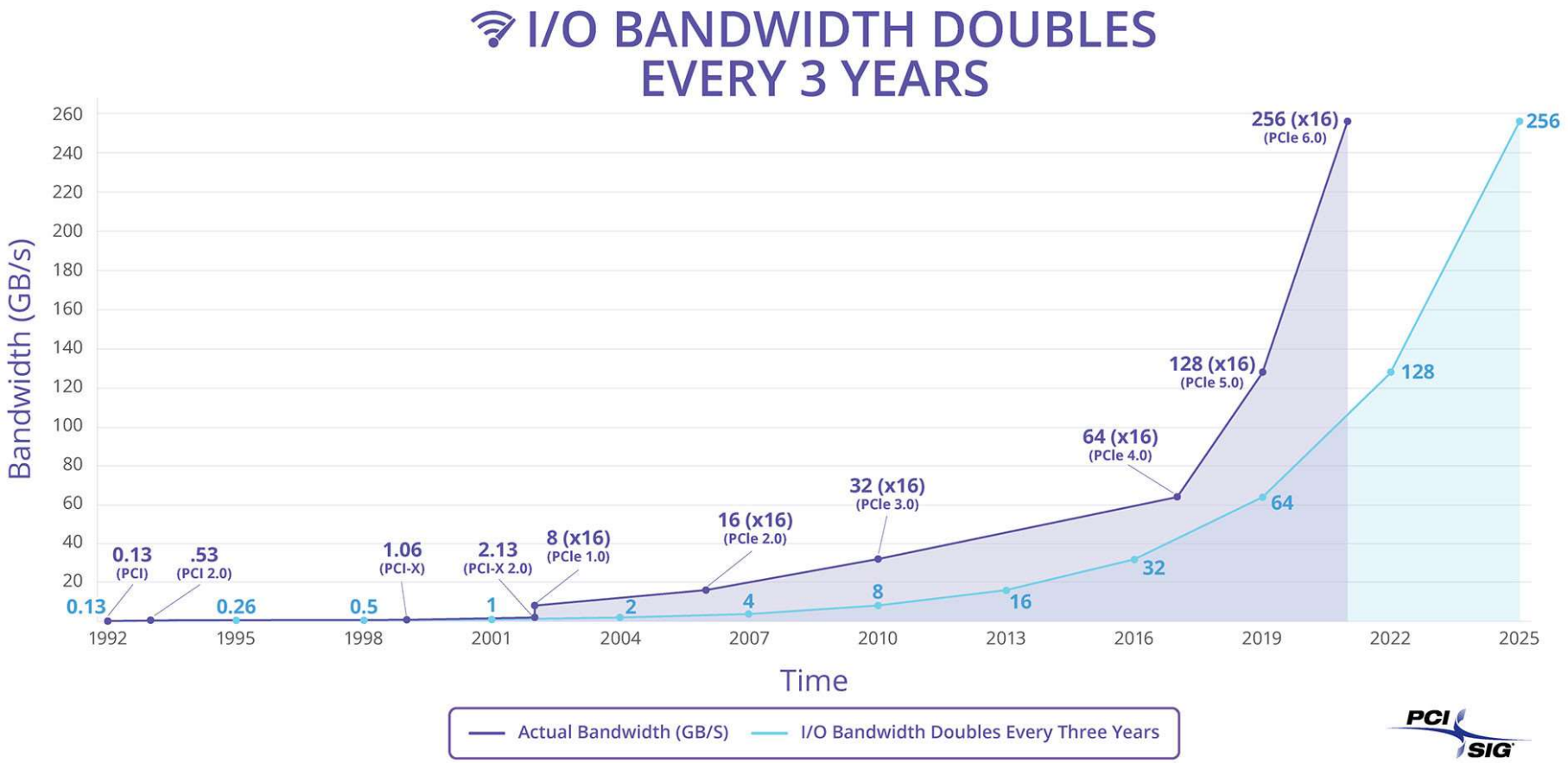
- High speed data transfer performance for each drive within an array
  - 1.5 Gbits/sec for SATA I and 3Gbits/sec for SATA II.
- Device can be hot swapped without shutting down or restarting the system
- Scalability: ATA is a point-to-point connection and allows multiple ports to be aggregated into a single controller.
- Plug and Play under Windows 98SE/Me/2000/XP/Vista/7.
- OS transparent. No driver is needed. Works under DOS and all Windows.
- Simple cabling system.
  - Maximum distance between device and the host is 1 meter (3 feet) for internal connection and 2 meters (6 feet) for external eSATA connection.
- Built-in RAID support
- Low cost. SATA was created with desktop PC prices in mind.

Maximum data transfer speeds



Content

Moore again


I/O BANDWIDTH DOUBLES EVERY 3 YEARS
Bandwidth (GB/s)
260
240
220
200
180
160
140
120
100
80
60
40
20
0.13 (PCI)
.53 (PCI 2.0)
1.06 (PCI-X)
2.13 (PCI-X 2.0)
8 (x16) (PCIe 1.0)
16 (x16) (PCIe 2.0)
32 (x16) (PCIe 3.0)
64 (x16) (PCIe 4.0)
128 (x16) (PCIe 5.0)
256 (x16) (PCIe 6.0)
0.13
0.26
0.5
1
2
4
8
16
32
64
128
256
1992
1995
1998
2001
2004
2007
2010
2013
2016
2019
2022
2025
Time
Actual Bandwidth (GB/S)
I/O Bandwidth Doubles Every Three Years

Blu-ray – Supplementary Reading

- Blu-ray
  - http://en.wikipedia.org/wiki/Blu-ray (well written article)

Oppo UDP 203 4k Blu Ray Player

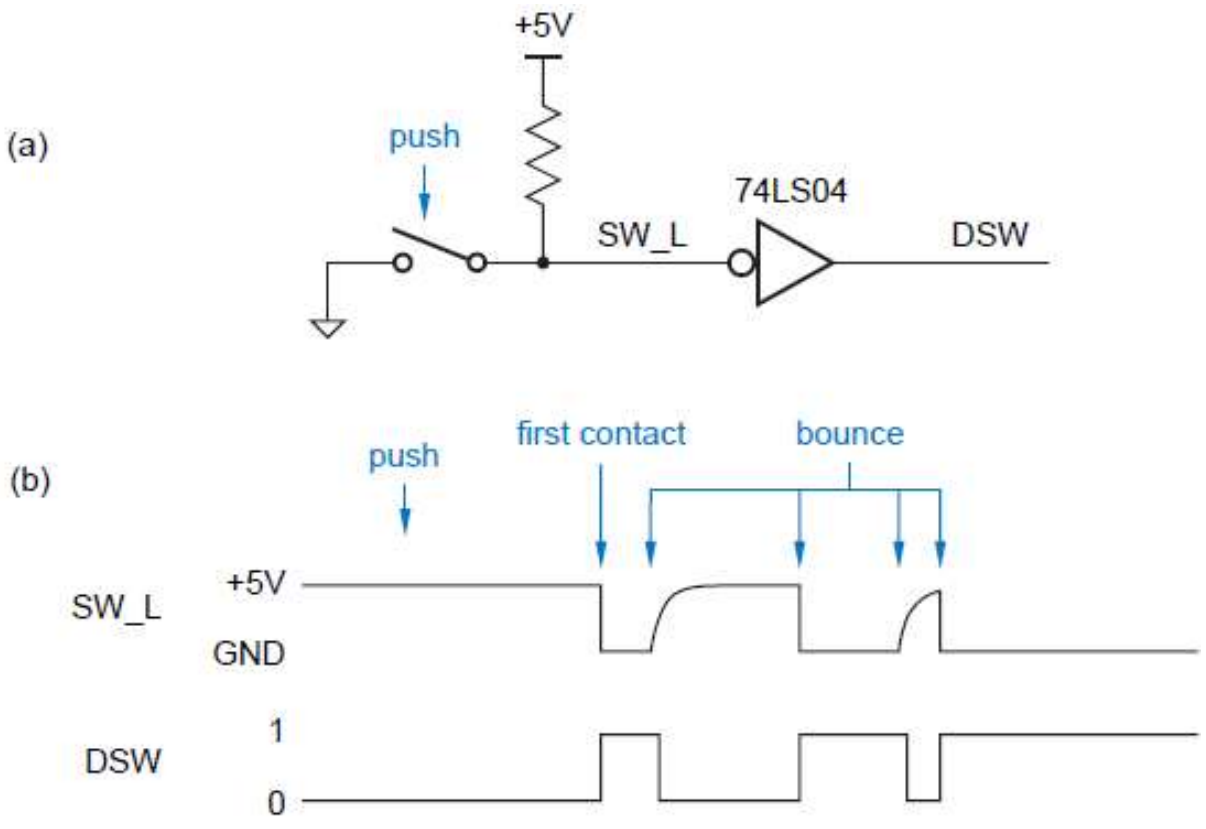
Content

Keys and Keyboard

Switch Debouncing

- A common application of simple bistables and latches is switch debouncing.
- We're all familiar with electrical switches from experience with lights, garbage disposals, and other appliances.
- Switches connected to sources of constant logic 0 and 1 are often used in digital systems to supply "user inputs."
- However, in digital logic applications we must consider another aspect of switch operation, the time dimension.
- A simple make or break operation, which occurs instantly as far as we slow-moving humans are concerned, actually has several phases that are discernible by high-speed digital logic.

Switch Debouncing



Switch Debouncing

- Figure (a) shows how a single-pole, single-throw (SPST) switch might be used to generate a single logic input.
- A pull-up resistor provides a logic-1 value when the switch is opened, and the switch contact is tied to ground to provide a logic-1 value when the switch is pushed.
- As shown in (b), it takes a while after a push for the wiper to hit the bottom contact.
- Once it hits, it doesn't stay there for long; it bounces a few times before finally settling.
- The result is that several transitions are seen on the SW_L and DSW logic signals for each single switch push. This behavior is called *contact bounce.*
- *Typical switches bounce for 10–20 ms, a very long time compared to the* switching speeds of logic gates.

Switch Debouncing

- Contact bounce may or may not be a problem, depending on the switch application.
- For example, some computers have configuration information specified by small switches, called *DIP switches because they are the same size* as a dual in-line package (DIP).
- Since DIP switches are normally changed only when the computer is inactive, there's no problem.

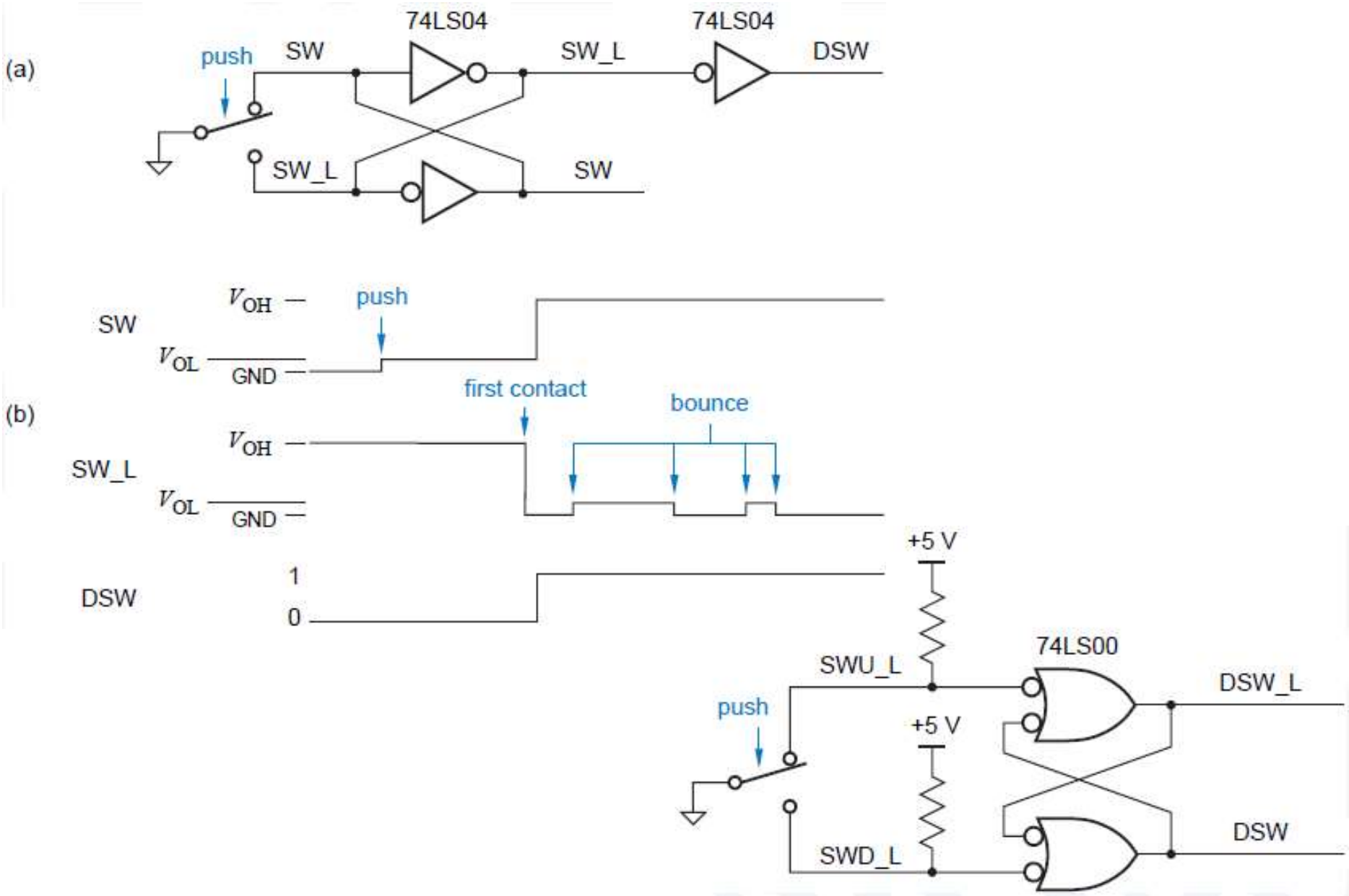
Switch Debouncing

- Contact bounce *is a problem* if a switch is being used to count or signal some event (e.g., laps in a race).
- Then we must provide a circuit (or, in microprocessor-based systems, software) to *debounce the switch—to provide just one signal change or pulse for each* external event.

The Simplest Switch Debouncer

- Switch debouncing is a good application for the simplest sequential circuit, the bistable element.
- This circuit uses a single-pole, double-throw (SPDT) switch.
- The switch contacts and wiper have a “break before make” behavior, so the wiper terminal is “floating” at some time halfway through the switch depression.
- Before the button is pushed, the top contact holds SW at 0 V, a valid logic 0, and the top inverter produces a logic 1 on SW_L and on the bottom contact.
- When the button is pushed and contact is broken, feedback in the bistable holds SW at V_{OL} *(≤ 0.5 V for LS-TTL), still a valid logic 0.*

Switch Debouncing



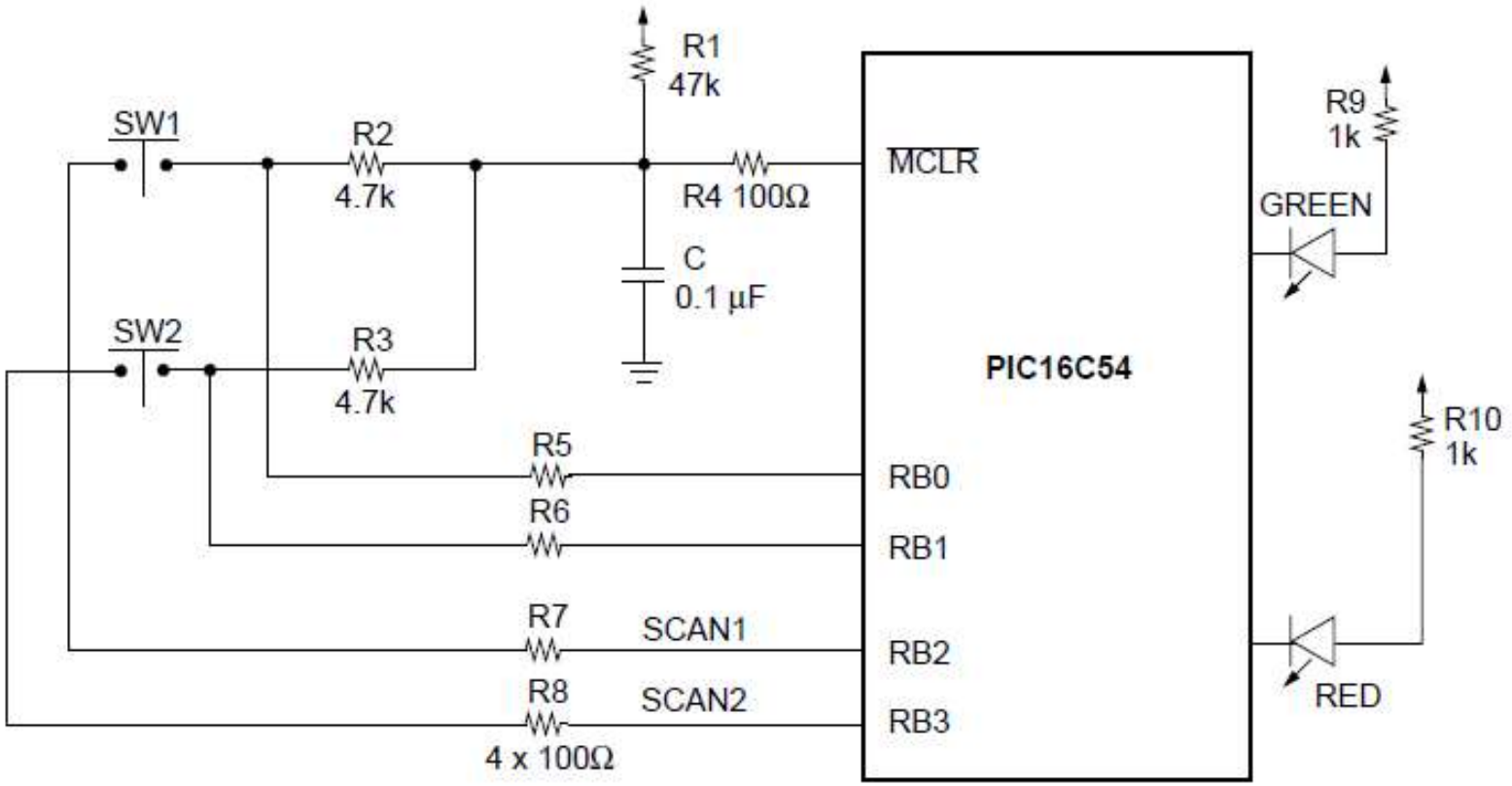
Switch Debouncing

- Next, when the wiper hits the bottom contact, the circuit operates quite unconventionally for a moment.
- The top inverter in the bistable is trying to maintain a logic 1 on the SW_L signal; the top transistor in its totem-pole output is “on” and connecting SW_L through a small resistance to +5 V.
- Suddenly, the switch contact makes a metallic connection of SW_L to ground, 0.0 V. Not surprisingly, the switch contact wins.

Switch Debouncing

- A short time later (30 ns for the 74LS04), the forced logic 0 on SW_L propagates through the two inverters of the bistable, so that the top inverter gives up its vain attempt to drive a 1, and instead drives a logic 0 onto SW_L.
- At this point, the top inverter output is no longer shorted to ground, and feedback in the bistable maintains the logic 0 on SW_L even if the wiper bounces off the bottom contact, as it does. (It does not bounce far enough to touch the top contact again.)

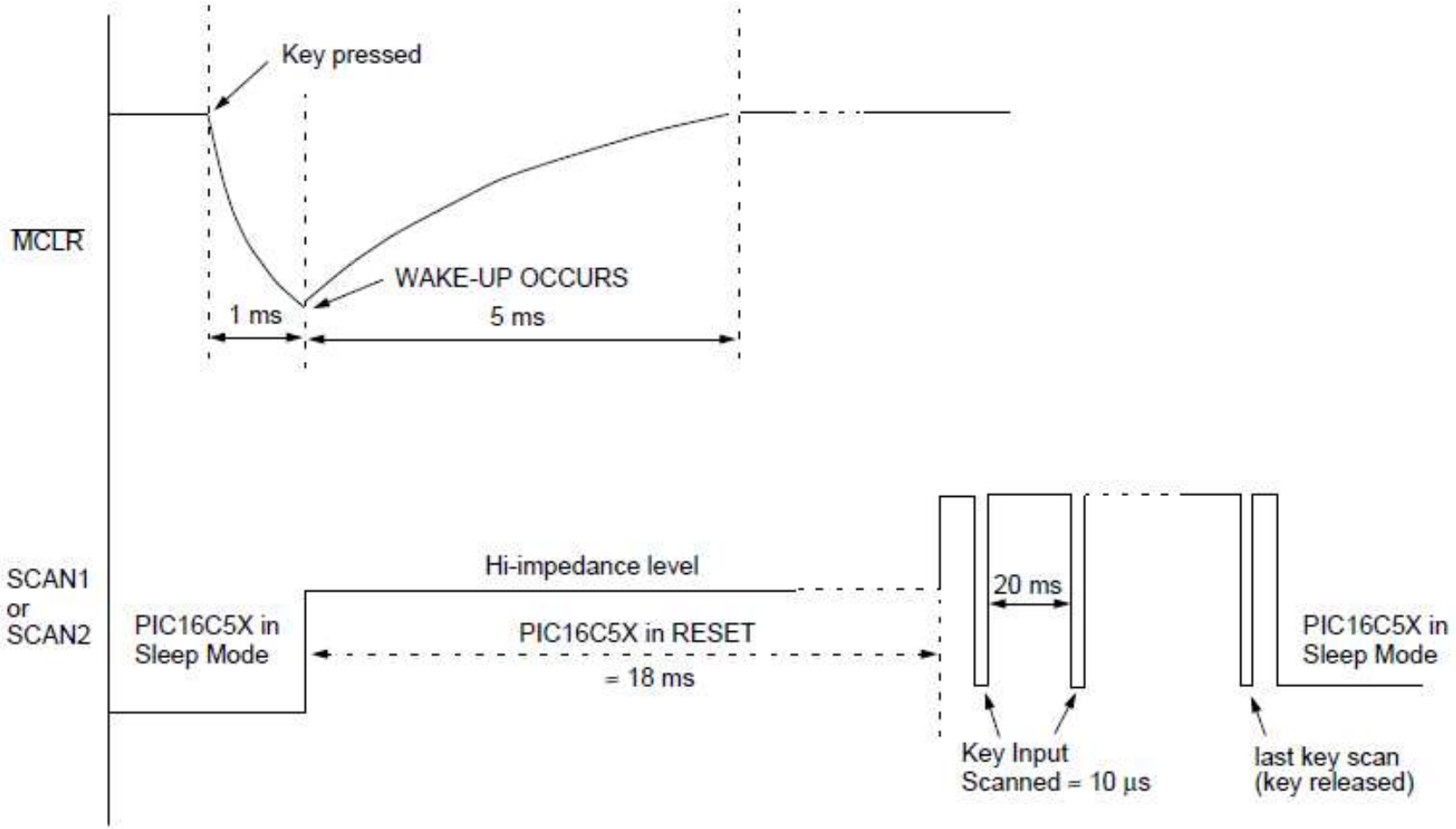
2 keys + Wake-up



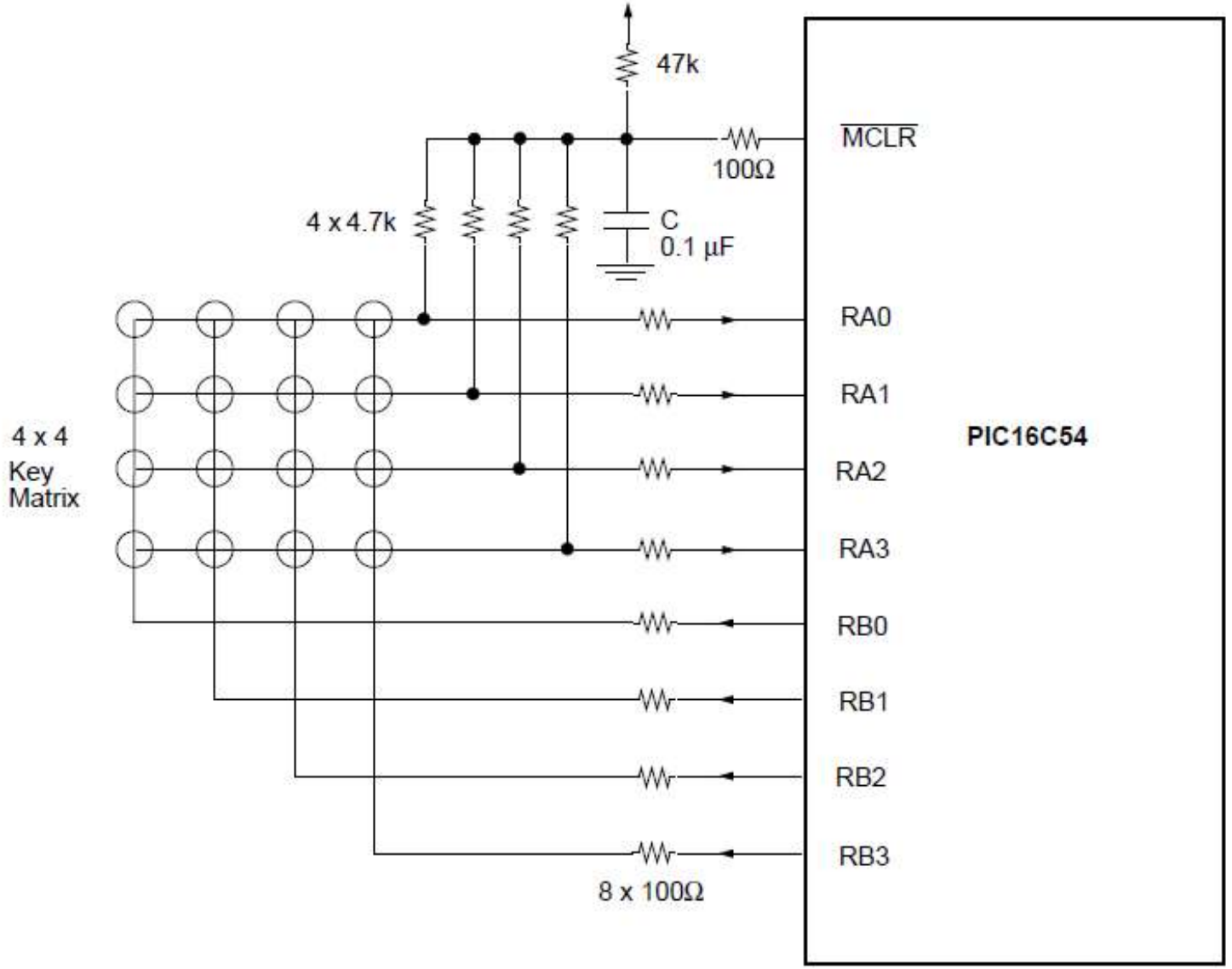
2 keys + Wake-up

- The PIC is normally in SLEEP mode consuming very little operating current
- If either of the two keys is pressed, the PIC16C5X “wakes up”, scans the keys and turns on one or both of the LED’s.
- In sleep mode, the scan outputs (SCAN1 and SCAN2) are both set to a low logic level. C is fully charged and MCLR = 1.

Timing Diagram

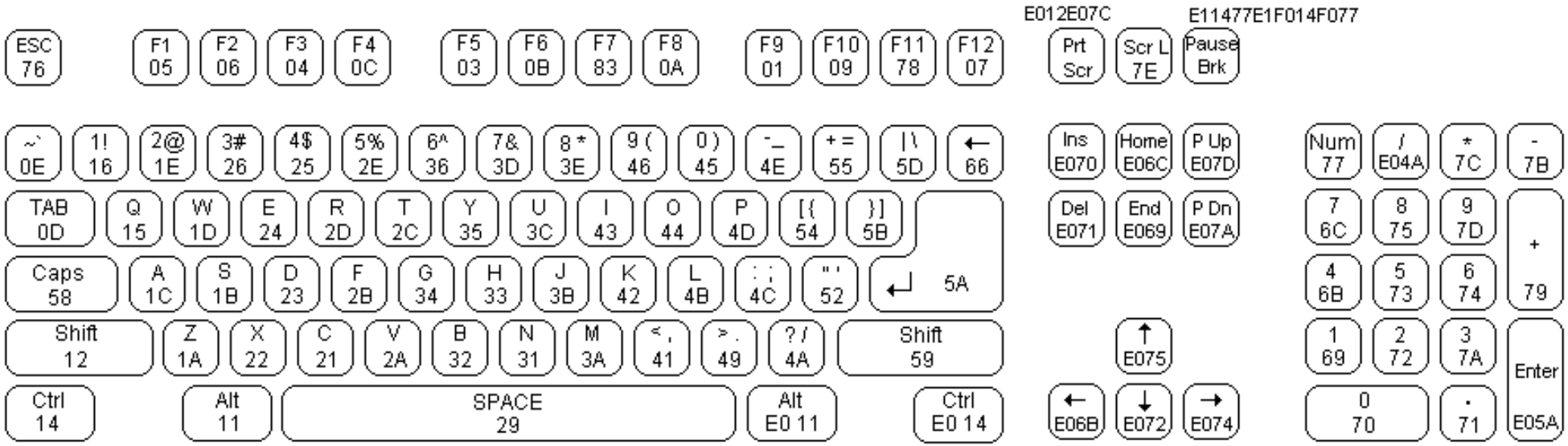


4 x 4 Key Matrix Interface



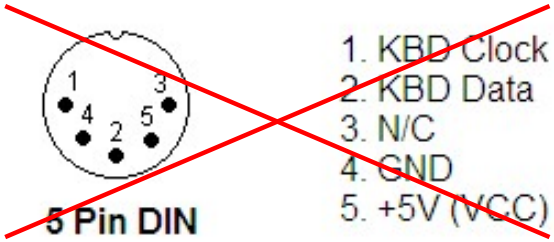
PC AT Scan Codes

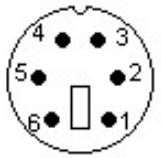
- The Scan code is shown on the bottom of the key.
- E.g. The Scan Code for ESC is 76.
- All the scan codes are shown in Hex.
- the scan code assignments are quite random.
- In many cases the easiest way to convert the scan code to ASCII would be to use a look up table.



The Keyboard's Connector

- The PC's AT Keyboard is connected to external equipment using four wires.
- These wires are shown below for the 5 Pin DIN Male Plug & PS/2 Plug.
- Both the KBD Clock and KBD Data are Open Collector bi-directional I/O Lines.
  - If desired, the Host can talk to the keyboard using these lines.
- Note:
  - Most keyboards are specified to drain a maximum 300mA. This will need to be considered when powering your devices





The Keyboard's Protocol

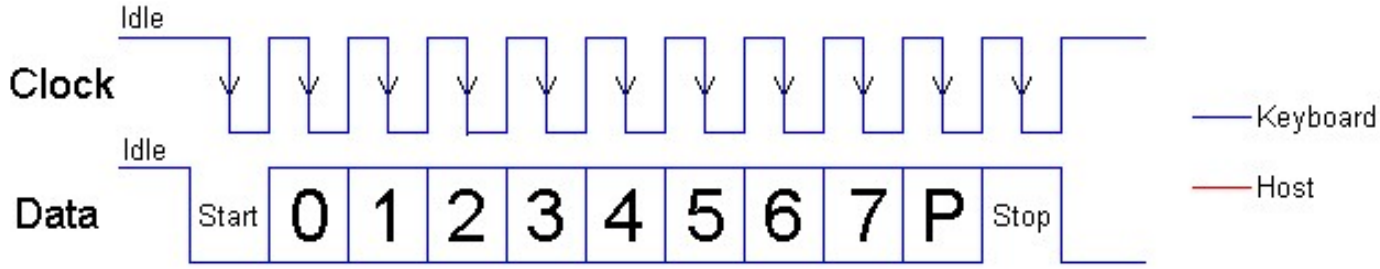
- **Keyboard to Host**
- **Host to Keyboard**

Keyboard to Host

- the PC's keyboard implements a bi-directional protocol.
- The keyboard can send data to the Host and the Host can send data to the Keyboard.
  - The Host has the ultimate priority over direction.
  - It can at anytime (although the not recommended) send a command to the keyboard.
- The keyboard is free to send data to the host when both the KBD Data and KBD Clock lines are high (Idle).
  - The KBD Clock line can be used as a Clear to Send line.
  - If the host takes the KBD Clock line low, the keyboard will buffer any data until the KBD Clock is released, ie goes high.
  - Should the Host take the KBD Data line low, then the keyboard will prepare to accept a command from the host.

Keyboard to Host

- The transmission of data (Keyboard to Host) is done with a frame of 11 bits.
- The first bit is a Start Bit (Logic 0) followed by 8 data bits (LSB First), one Parity Bit (Odd Parity) and a Stop Bit (Logic 1).
- Each bit should be read on the falling edge of the clock.
- The data line only has to be valid on the falling edge of the clock. The Keyboard will generate the clock.
- The frequency of the clock signal typically ranges from 20 to 30 KHz.



Host to Keyboard

- The Host to Keyboard Protocol is initiated by taking the KBD data line low.
  - to prevent the keyboard from sending data at the same time that you attempt to send the keyboard data, it is common to take the KBD Clock line low for more than 60us.
  - This is more than one bit length.
  - Then the KBD data line is taken low, while the KBD clock line is released.
- The keyboard will start generating a clock signal on it's KBD clock line.
- This process can take up to 10ms.
- After the first falling edge has been detected, one can load the first data bit on the KBD Data line.
  - This bit will be read into the keyboard on the next falling edge, after which one can place the next bit of data.
  - This process is repeated for the 8 data bits. After the data bits come an Odd Parity Bit.
  - Once the Parity Bit has been sent and the KBD Data Line is in a idle (High) state for the next clock cycle, the keyboard will acknowledge the reception of the new data.
  - The keyboard does this by taking the KBD Data line low for the next clock transition.
- If the KBD Data line is not idle after the 10th bit (Start, 8 Data bits + Parity), the keyboard will continue to send a KBD Clock signal until the KBD Data line becomes idle.

Host to Keyboard