Click to verify

For the taxonomic method, see DNA barcoding. For a code of conduct for barristers, see Legal ethics. A barcode or bar code is a method of representing data in a visual, machine-readable form. Initially, barcodes represented data by varying the widths, spacings and sizes of parallel lines. These barcodes, now commonly referred to as linear or one-dimensional (1D), can be scanned by special optical scanners, called barcode readers, of which there are several types. A UPC-A barcode Later, two-dimensional (2D) variants were developed, using rectangles, dots, hexagons and other patterns, called 2D barcodes or matrix codes, although they do not use bars as such. Both can be read using purpose-built 2D optical scanners, which exist in a few different forms. Matrix codes can also be read by a digital camera connected to a microcomputer running software that takes a photographic image of the barcode and analyzes the image to deconstruct and decode the code. A mobile device with a built-in camera, such as a smartphone, can function as the latter type of barcode reader using specialized application software and is suitable for both 1D and 2D codes. Barcoded rolling stock in the UK, 1962 The barcode was invented by Norman Joseph Woodland and Bernard Silver and patented in the US in 1952.[1] The invention was based on Morse code[2] that was extended to thin and thick bars. However, it took over twenty years before this invention became commercially successful. UK magazine Modern Railways December 1962 pages 387–389 record how British Railways had already perfected a barcode-reading system capable of correctly reading rolling stock travelling at 100 mph (160 km/h) with no mistakes. An early use of one type of barcode in an industrial context was sponsored by the Association of American Railroads in the late 1960s. Developed by General Telephone and Electronics (GTE) and called KarTrak ACI (Automatic Car Identification), this scheme involved placing colored stripes in various combinations on steel plates which were affixed to the sides of railroad rolling stock. Two plates were used per car, one on each side, with the arrangement of the colored stripes encoding information such as ownership, type of equipment, and identification number.[3] The plates were read by a trackside scanner located, for instance, at the entrance to a classification yard, while the car was moving past.[4] The project was abandoned after about ten years because the system proved unreliable after long-term use.[3] Barcodes became commercially successful when they were used to automate supermarket checkout systems, a task for which they have become almost universal. The Uniform Grocery Product Code Council had chosen, in 1973, the barcode design developed by George Laurer. Laurer's barcode, with vertical bars, printed better than the circular barcode developed by Woodland and Silver.[5] Their use has spread to many other tasks that are generically referred to as automatic identification and data capture (AIDC). The first successful system using barcodes was in the UK supermarket group Sainsbury's in 1972 using shelf-mounted barcodes which were developed by Plessey.[6][7] In June 1974, Marsh supermarket in Troy, Ohio used a scanner made by Photographic Sciences Corporation to scan the Universal Product Code (UPC) barcode on a pack of Wrigley's chewing gum.[8][5] QR codes, a specific type of 2D barcode, rose in popularity in the second decade of the 2000s due to the growth in smartphone ownership.[9] Other systems have made inroads in the AIDC market, but the simplicity, universality and low cost of barcodes has limited the role of these other systems, particularly before technologies such as radio-frequency identification (RFID) became available after 2023. This article duplicates the scope of other articles, specifically Universal Product Code#History. Please discuss this issue and help introduce a summary style to the article. (December 2013) In 1948, Bernard Silver, a graduate student at Drexel Institute of Technology in Philadelphia, Pennsylvania, US overheard the president of the local food chain, Food Fair, asking one of the deans to research a system to automatically read product information during checkout.[10] Silver told his friend Norman Joseph Woodland about the request, and they started working on a variety of systems. Their first working system used ultraviolet ink, but the ink faded too easily and was expensive.[11] Convinced that the system was workable with further development, Woodland left Drexel, moved into his father's apartment in Florida, and continued working on the system. His next inspiration came from Morse code, and he formed his first barcode from sand on the beach. "I just extended the dots and dashes downwards and made narrow lines and wide lines out of them."[11] To read them, he adapted technology from optical soundtracks in movies, using a 500-watt incandescent light bulb shining through the paper onto an RCA935 photomultiplier tube (from a movie projector) on the far side. He later decided that the system would work better if it were printed as a circle instead of a line, allowing it to be scanned in any direction. On 20 October 1949 Woodland and Silver filed a patent application for "Classifying Apparatus and Method", in which they described both the linear and bull's eye printing patterns, as well as the mechanical and electronic systems needed to read the code. The patent was issued on 7 October 1952 as US Patent 2,612,994.[1] In 1951, Woodland moved to IBM and continually tried to interest IBM in developing the system. The company eventually commissioned a report on the idea, which concluded that it was both feasible and interesting, but that processing the resulting information would require equipment that was some time off in the future. IBM offered to buy the patent, but the offer was not accepted. Philco purchased the patent in 1962 and then sold it to RCA sometime later.[11] During his time as an undergraduate, David Jarrett Collins worked at the Pennsylvania Railroad and became aware of the need to automatically identify railroad cars. Immediately after receiving his master's degree from MIT in 1959, he started work at GTE Sylvania and began addressing the problem. He developed a system called KarTrak using blue, white and red reflective stripes attached to the side of the cars, encoding a four-digit company identifier and a six-digit car number.[11] Light reflected off the colored stripes was read by photomultiplier vacuum tubes.[12] The Boston and Maine Railroad tested the KarTrak system on their gravel cars in 1961. The tests continued until 1967, when the Association of American Railroads (AAR) selected it as a standard, automatic car identification, across the entire North American fleet. The installations began on 10 October 1967. However, the economic downturn and rash of bankruptcies in the industry in the early 1970s greatly slowed the rollout, and it was not until 1974 that 95% of the fleet was labeled. To add to its woes, the system was found to be easily fooled by dirt in certain applications, which greatly affected accuracy. The AAR abandoned the system in the late 1970s, and it was not until the mid-1980s that they introduced a similar system, this time based on radio tags.[13] The railway project had failed, but a toll bridge in New Jersey requested a similar system so that it could quickly scan for cars that had purchased a monthly pass. Then the US Post Office requested a system to track trucks entering and leaving their facilities. These applications required special retroreflector labels. Finally, Kal Kan asked the Sylvania team for a simpler (and cheaper) version which they could put on cases of pet food for inventory control. In 1967, with the railway system maturing, Collins went to management looking for funding for a project to develop a black-and-white version of the code for other industries. They declined, saying that the railway project was large enough, and they saw no need to branch out so quickly. Collins then quit Sylvania and formed the Computer Identics Corporation.[11] As its first innovations, Computer Identics moved from using incandescent light bulbs in its systems, replacing them with helium–neon lasers, and incorporated a mirror as well, making it capable of locating a barcode up to a meter (3 feet) in front of the scanner. This made the entire process much simpler and more reliable, and typically enabled these devices to deal with damaged labels, as well, by recognizing and reading the intact portions. Computer Identics Corporation installed one of its first two scanning systems in the spring of 1969 at a General Motors (Buick) factory in Flint, Michigan.[11] The system was used to identify a dozen types of transmissions moving on an overhead conveyor from production to shipping. The other scanning system was installed at General Trading Company's distribution center in Carlstadt, New Jersey to direct shipments to the proper loading bay. Main article: Universal Product Code In 1966 the National Association of Food Chains (NAFC) held a meeting on the idea of automated checkout systems. RCA, which had purchased the rights to the original Woodland patent, attended the meeting and initiated an internal project to develop a system based on the bullseye code. The Kroger grocery chain volunteered to test it. In the mid-1970s the NAFC established the Ad-Hoc Committee for U.S. Supermarkets on a Uniform Grocery-Product Code to set guidelines for barcode development. In addition, it created a symbol-selection subcommittee to help standardize the approach. In cooperation with consulting firm, McKinsey & Co., they developed a standardized 11-digit code for identifying products. The committee then sent out a contract tender to develop a barcode system to print and read the code. The request went to Singer, National Cash Register (NCR), Litton Industries, RCA, Pitney-Bowes, IBM and many others.[14] A wide variety of barcode approaches was studied, including linear codes, RCA's bullseye concentric circle code, starburst patterns and others. In the spring of 1971 RCA demonstrated their bullseye code at another industry meeting. IBM executives at the meeting noticed the crowds at the RCA booth and immediately developed their own system. IBM marketing specialist Alec Jablonover remembered that the company still employed Woodland, and he established a new facility in Research Triangle Park to lead development. In July 1972 RCA began an 18-month test in a Kroger store in Cincinnati. Barcodes were printed on small pieces of adhesive paper, and attached by hand by store employees when they were adding price tags. The code proved to have a serious problem; the printers would sometimes smear ink, rendering the code unreadable in most orientations. However, a linear code, like the one being developed by Woodland at IBM, was printed in the direction of the stripes, so extra ink would simply make the code "taller" while remaining readable. So on 3 April 1973 the IBM UPC was selected as the NAFC standard. IBM had designed five versions of UPC symbology for future industry requirements: UPC A, B, C, D, and E.[15] NCR installed a testbed system at Marsh's Supermarket in Troy, Ohio, near the factory that was producing the equipment. On 26 June 1974, a 10-pack of Wrigley's Juicy Fruit gum was scanned, registering the first commercial use of the UPC.[16] In 1971 an IBM team was assembled for an intensive planning session, threshing out, 12 to 18 hours a day, how the technology would be deployed and operate cohesively across the system, and scheduling a roll-out plan. By 1973, the team were meeting with grocery manufacturers to introduce the symbol that would need to be printed on the packaging or labels of all of their products. There were no cost savings for a grocery to use it, unless at least 70% of the grocery's products had the barcode printed on the product by the manufacturer. IBM projected that 75% would be needed in 1975. Economic studies conducted for the grocery industry committee projected over $40 million in savings to the industry from scanning by the mid-1970s. Those numbers were not achieved in that time-frame and some predicted the demise of barcode scanning. The usefulness of the barcode required the adoption of expensive scanners by a critical mass of retailers while manufacturers simultaneously adopted barcode labels. Neither wanted to move first and results were not promising for the first couple of years, with Business Week proclaiming "The Supermarket Scanner That Failed" in a 1976 article.[16][17] Sims Supermarkets were the first location in Australia to use barcodes, starting in 1979.[18] A barcode system is a network of hardware and software, consisting primarily of mobile computers, printers, handheld scanners, infrastructure, and supporting software. Barcode systems are used to automate data collection where hand recording is neither timely nor cost effective. Despite often being provided by the same company, Barcoding systems are not radio-frequency identification (RFID) systems. Many companies use both technologies as part of larger resource management systems. A typical barcode system consist of some infrastructure, either wired or wireless that connects some number of mobile computers, handheld scanners, and printers to one or many databases that store and analyze the data collected by the system. At some level there must be some software to manage the system. The software may be as simple as code that manages the connection between the hardware and the database or as complex as an ERP, MRP, or some other inventory management software. A wide range of hardware is manufactured for use in barcode systems by such manufacturers as Datalogic, Intermec, HHP (Hand Held Products), Microscan Systems, Unitech, Metrologic, PSC, and PANMOBIL, with the best known brand of handheld scanners and mobile computers being produced by Symbol,[citation needed] a division of Motorola. Some ERP, MRP, and other inventory management software have built in support for barcode reading. Alternatively, custom interfaces can be created using a language such as C++, C#, Java, Visual Basic.NET, and many others. In addition, software development kits are produced to aid the process. In 1981 the United States Department of Defense adopted the use of Code 39 for marking all products sold to the United States military. This system, Logistics Applications of Automated Marking and Reading Symbols (LOGMARS), is still used by DoD and is widely viewed as the catalyst for widespread adoption of barcoding in industrial uses.[19] Snack vendor on the Shinkansen train scans a barcode. EAN-13 ISBN barcode Barcode on a patient identification wristband Barcoded parcel Barcodes are widely used around the world in many contexts. In stores, UPC barcodes are pre-printed on most items other than fresh produce from a grocery store. This speeds up processing at check-outs and helps track items and also reduces instances of shoplifting involving price tag swapping, although shoplifters can now print their own barcodes.[20] Barcodes that encode a book's ISBN are also widely pre-printed on books, journals and other printed materials. In addition, retail chain membership cards use barcodes to identify customers, allowing for customized marketing and greater understanding of individual consumer shopping patterns. At the point of sale, shoppers can get product discounts or special marketing offers through the address or e-mail address provided at registration. Barcodes are widely used in healthcare and hospital settings, ranging from patient identification (to access patient data, including medical history, drug allergies, etc.) to creating SOAP notes[21] with barcodes to medication management. They are also used to facilitate the separation and indexing of documents that have been imaged in batch scanning applications, track the organization of species in biology,[22] and integrate with in-motion checkweighers to identify the item being weighed in a conveyor line for data collection. They can also be used to keep track of objects and people; they are used to keep track of rental cars, airline luggage, nuclear waste, express mail, and parcels. Barcoded tickets (which may be printed by the customer on their home printer, or stored on their mobile device) allow the holder to enter sports arenas, cinemas, theatres, fairgrounds, and transportation, and are used to record the arrival and departure of vehicles from rental facilities etc. This can allow proprietors to identify duplicate or fraudulent tickets more easily. Barcodes are widely used in shop floor control applications software where employees can scan work orders and track the time spent on a job. Barcodes are also used in some kinds of non-contact 1D and 2D position sensors. A series of barcodes are used in some kinds of absolute 1D linear encoder. The barcodes are packed close enough together that the reader always has one or two barcodes in its field of view. As a kind of fiducial marker, the relative position of the barcode in the field of view of the reader gives incremental precise positioning, in some cases with sub-pixel resolution. The data decoded from the barcode gives the absolute coarse position. An "address carpet", used in digital paper, such as Howell's binary pattern and the Anoto dot pattern, is a 2D barcode designed so that a reader, even though only a tiny portion of the complete carpet is in the field of view of the reader, can find its absolute X, Y position and rotation in the carpet.[23][24] Matrix codes can embed a hyperlink to a web page. A mobile device with a built-in camera might be used to read the pattern and browse the linked website, which can help a shopper find the best price for an item in the vicinity. Since 2005, airlines use an IATA-standard 2D barcode on boarding passes (Bar Coded Boarding Pass (BCBP)), and since 2008 2D barcodes sent to mobile phones enable electronic boarding passes.[25] Some applications for barcodes have fallen out of use. In the 1970s and 1980s, software source code was occasionally encoded in a barcode and printed on paper (Cauzin Softstrip and Paperbyte[26] are barcode symbologies specifically designed for this application), and the 1991 Barcode Battler computer game system used any standard barcode to generate combat statistics. Artists have used barcodes in art, such as Scott Blake's Barcode Jesus, as part of the post-modernism movement. The mapping between messages and barcodes is called a symbology. The specification of a symbology includes the encoding of the message into bars and spaces, any required start and stop markers, the size of the quiet zone required to be before and after the barcode, and the computation of a checksum. Linear symbologies can be classified mainly by two properties: Continuous vs. discrete Characters in discrete symbologies are composed of n bars and n − 1 spaces. There is an additional space between characters, but it does not convey information, and may have any width as long as it is not confused with the end of the code. Characters in continuous symbologies are composed of n bars and n spaces, and usually abut, with one character ending with a space and the next beginning with a bar, or vice versa. A special end pattern that has bars on both ends is required to end the code. Two-width vs. many-width A two-width, also called a binary bar code, contains bars and spaces of two widths, "wide" and "narrow". The precise width of the wide bars and spaces is not critical; typically, it is permitted to be anywhere between 2 and 3 times the width of the narrow equivalents. Some other symbologies use bars of two different heights (POSTNET), or the presence or absence of bars (CPC Binary Barcode). These are normally also considered binary bar codes. Bars and spaces in many-width symbologies are all multiples of a basic width called the module; most such codes use four widths of 1, 2, 3 and 4 modules. Some symbologies use interleaving. The first character is encoded using black bars of varying width. The second character is then encoded by varying the width of the white spaces between these bars. Thus, characters are encoded in pairs over the same section of the barcode. Interleaved 2 of 5 is an example of this. Stacked symbologies repeat a given linear symbology vertically. The most common among the many 2D symbologies are matrix codes, which feature square or dot-shaped modules arranged on a grid pattern. 2D symbologies also come in circular and other patterns and may employ steganography, hiding modules within an image (for example, DataGlyphs). Linear symbologies are optimized for laser scanners, which sweep a light beam across the barcode in a straight line, reading a slice of the barcode light-dark patterns. Scanning at an angle makes the modules appear wider, but does not change the width ratios. Stacked symbologies are also optimized for laser scanning, with the laser making multiple passes across the barcode. In the 1990s development of charge-coupled device (CCD) imagers to read barcodes was pioneered by Welch Allyn. Imaging does not require moving parts, as a laser scanner does. In 2007, linear imaging had begun to supplant laser scanning as the preferred scan engine for its performance and durability. 2D symbologies cannot be read by a laser, as there is typically no sweep pattern that can encompass the entire symbol. They must be scanned by an image-based scanner employing a CCD or other digital camera sensor technology. Main article: Barcode reader GTIN barcodes on Coca-Cola bottles. The images at right show how the laser of barcode readers "see" the images behind a red filter. The earliest, and still[when?] the cheapest, barcode scanners are built from a fixed light and a single photosensor that is manually moved across the barcode. Barcode scanners can be classified into three categories based on their connection to the computer. The older type is the RS-232 barcode scanner. This type requires special programming for transferring the input data to the application program. Keyboard interface scanners connect to a computer using a PS/2 or AT keyboard–compatible adaptor cable (a "keyboard wedge"). The barcode's data is sent to the computer as if it had been typed on the keyboard. Like the keyboard interface scanner, USB scanners do not need custom code for transferring input data to the application program. On PCs running Windows the human interface device emulates the data merging action of a hardware "keyboard wedge", and the scanner automatically behaves like an additional keyboard. Most modern smartphones are able to decode barcode using their built-in camera. Google's mobile Android operating system can use their own Google Lens application to scan QR codes, or third-party apps like Barcode Scanner to read both one-dimensional barcodes and QR codes. Google's Pixel devices can natively read QR codes inside the default Pixel Camera app. Nokia's Symbian operating system featured a barcode scanner,[27] while mbarcode[28] is a QR code reader for the Maemo operating system. In Apple iOS 11, the native camera app can decode QR codes and can link to URLs, join wireless networks, or perform other operations depending on the QR Code contents.[29] Other paid and free apps are available with scanning capabilities for other symbologies or for earlier iOS versions.[30] With BlackBerry devices, the App World application can natively scan barcodes and load any recognized Web URLs on the device's Web browser. Windows Phone 7.5 is able to scan barcodes through the Bing search app. However, these devices are not designed specifically for the capturing of barcodes. As a result, they do not decode nearly as quickly or accurately as a dedicated barcode scanner or portable data terminal.[citation needed] It is common for producers and users of bar codes to have a quality management system which includes verification and validation of bar codes.[31] Barcode verification examines scanability and the quality of the barcode in comparison to industry standards and specifications.[32] Barcode verifiers are primarily used by businesses that print and use barcodes. Any trading partner in the supply chain can test barcode quality. It is important to verify a barcode to ensure that any reader in the supply chain can successfully interpret a barcode with a low error rate. Retailers levy large penalties for non-compliant barcodes. These chargebacks can reduce a manufacturer's revenue by 2% to 10%.[33] A barcode verifier works the way a reader does, but instead of simply decoding a barcode, a verifier performs a series of tests. For linear barcodes these tests are: Edge contrast (EC)[34] The difference between the space reflectance (Rs) and adjoining bar reflectance (Rb). EC=Rs-Rb Minimum bar reflectance (Rb)[34] The smallest reflectance value in a bar. Minimum space reflectance (Rs)[34] The smallest reflectance value in a space. Symbol contrast (SC)[34] Symbol contrast is the difference in reflectance values of the lightest space (including the quiet zone) and the darkest bar of the symbol. The greater the difference, the higher the grade. The parameter is graded as either A, B, C, D, or F. SC=Rmax-Rmin Minimum edge contrast (ECmin)[34] The difference between the space reflectance (Rs) and adjoining bar reflectance (Rb). EC=Rs-Rb Modulation (MOD)[34] The parameter is graded either A, B, C, D, or F. This grade is based on the relationship between minimum edge contrast (ECmin) and symbol contrast (SC). MOD=ECmin/SC The greater the difference between minimum edge contrast and symbol contrast, the lower the grade. Scanners and verifiers perceive the narrower bars and spaces to have less intensity than wider bars and spaces; the comparison of the lesser intensity of narrow elements to the wide elements is called modulation. This condition is affected by aperture size. Inter-character gap[34] In discrete barcodes, the space that disconnects the two contiguous characters. When present, inter-character gaps are considered spaces (elements) for purposes of edge determination and reflectance parameter grades. Defects Decode[34] Extracting the information which has been encoded in a bar code symbol. Decodability[34] Can be graded as A, B, C, D, or F. The Decodability grade indicates the amount of error in the width of the most deviant element in the symbol. The less deviation in the symbology, the higher the grade. Decodability is a measure of print accuracy using the symbology reference decode algorithm. 2D matrix symbols look at the parameters: Symbol contrast[34] Modulation[34] Decode[34] Unused error correction Fixed (finder) pattern damage Grid non-uniformity Axial non-uniformity[35] Depending on the parameter, each ANSI test is graded from 0.0 to 4.0 (F to A), or given a pass or fail mark. Each grade is determined by analyzing the scan reflectance profile (SRP), an analog graph of a single scan line across the entire symbol. The lowest of the 8 grades is the scan grade, and the overall ISO symbol grade is the average of the individual scan grades. For most applications a 2.5 (C) is the minimal acceptable symbol grade.[36] Compared with a reader, a verifier measures a barcode's optical characteristics to international and industry standards. The measurement must be repeatable and consistent. Doing so requires constant conditions such as distance, illumination angle, sensor angle and verifier aperture. Based on the verification results, the production process can be adjusted to print higher quality barcodes that will scan down the supply chain. Bar code validation may include evaluations after use (and abuse) testing such as sunlight, abrasion, impact, moisture, etc.[37] Barcode verifier standards are defined by the International Organization for Standardization (ISO), in ISO/IEC 15426-1 (linear) or ISO/IEC 15426-2 (2D).[citation needed] The current international barcode quality specification is ISO/IEC 15416 (linear) and ISO/IEC 15415 (2D).[citation needed] The European Standard EN 1635 has been withdrawn and replaced by ISO/IEC 15416. The original U.S. barcode quality specification was ANSI X3.182. (UPCs used in the US – ANSI/UCC5).[citation needed] As of 2011 the ISO workgroup JTC1 SC31 was developing a Direct Part Marking (DPM) quality standard: ISO/IEC TR 29158.[38] In point-of-sale management, barcode systems can provide detailed up-to-date information on the business, accelerating decisions and with more confidence. For example: Fast-selling items can be identified quickly and automatically reordered. Slow-selling items can be identified, preventing inventory build-up. The effects of merchandising changes can be monitored, allowing fast-moving, more profitable items to occupy the best space. Historical data can be used to predict seasonal fluctuations very accurately. Items may be repriced on the shelf to reflect both sale prices and price increases. This technology also enables the profiling of individual consumers, typically through a voluntary registration of discount cards. While pitched as a benefit to the consumer, this practice is considered to be potentially dangerous by privacy advocates.[which?] Besides sales and inventory tracking, barcodes are very useful in logistics and supply chain management. When a manufacturer packs a box for shipment, a unique identifying number (UID) can be assigned to the box. A database can link the UID to relevant information about the box; such as order number, items packed, quantity packed, destination, etc. The information can be transmitted through a communication system such as electronic data interchange (EDI) so the retailer has the information about a shipment before it arrives. Shipments that are sent to a distribution center (DC) are tracked before forwarding. When the shipment reaches its final destination, the UID gets scanned, so the store knows the shipment's source, contents, and cost. Barcode scanners are relatively low cost and extremely accurate compared to key-entry, with only about 1 substitution error in 15,000 to 36 trillion characters entered.[39][unreliable source?] The exact error rate depends on the type of barcode. A first generation, "one dimensional" barcode that is made up of lines and spaces of various widths or sizes that create specific patterns. Example Symbology Continuous or discrete Bar type Uses Codabar Discrete Two Old format used in libraries and blood banks and on airbills (out of date, but still widely used in libraries) Code 25 – Non-interleaved 2 of 5 Continuous Two Industrial Code 25 – Interleaved 2 of 5 Continuous Two Wholesale, libraries International standard ISO/IEC 16390 Code 11 Discrete Two Telephones (out of date) Farmacode or Code 32 Discrete Two Italian pharmacode – use Code 39 (no international standard available) Code 39 Discrete Two Various – international standard ISO/IEC 16388 Code 93 Continuous Many Various Code 128 Continuous Many Various – International Standard ISO/IEC 15417 CPC Binary Discrete Two Data Logic 2 of 5 Discrete Two Datalogic 2 of 5 can encode digits 0–9 and was used mostly in Chinese Postal Services. EAN 2 Continuous Many Addon code (magazines), GS1-approved – not an own symbology – to be used only with an EAN/UPC according to ISO/IEC 15420 EAN 5 Continuous Many Addon code (books), GS1-approved – not an own symbology – to be used only with an EAN/UPC according to ISO/IEC 15420 EAN-8, EAN-13 Continuous Many Worldwide retail, GS1-approved – International Standard ISO/IEC 15420 || | || Facing Identification Mark Discrete Two USPS business reply mail GS1-128 (formerly named UCC/EAN-128), incorrectly referenced as EAN 128 and UCC 128 Continuous Many Various, GS1-approved – just an application of the Code 128 (ISO/IEC 15417) using the ANS MH10.8.2 AI Datastructures. It is not a separate symbology. GS1 DataBar, formerly Reduced Space Symbology (RSS) Continuous Many Various, GS1-approved IATA 2 of 5 Discrete Two IATA 2 of 5 version of Industrial 2 of 5 is used by International Air Transport Association had fixed 17 digits length with 16 valuable package identification digit and 17-th check digit. Industrial 2 of 5 Discrete Two Industrial 2 of 5 can encode only digits 0–9 and at this time has only historical value. ITF-14 Continuous Two Non-retail packaging levels, GS1-approved – is just an Interleaved 2/5 Code (ISO/IEC 16390) with a few additional specifications, according to the GS1 General Specifications ITF-6 Continuous Two Interleaved 2 of 5 barcode to encode an addon to ITF-14 and ITF-16 barcodes. The code is used to encode additional data such as items quantity or container weight JAN Continuous Many Used in Japan, similar to and compatible with EAN-13 (ISO/IEC 15420) Japan Post barcode Discrete 4 bar heights Japan Post Matrix 2 of 5 Discrete Two Matrix 2 of 5 can encode digits 0–9 and was uses for warehouse sorting, photo finishing, and airline ticket marking. MSI Continuous Two Used for warehouse shelves and inventory Pharmacode Discrete Two Pharmaceutical packaging (no international standard available) PLANET Continuous Tall/short United States Postal Service (no international standard available) Plessey Continuous Two Catalogs, store shelves, inventory (no international standard available) Telepen Continuous Two Libraries (UK) Universal Product Code (UPC-A and UPC-E) Continuous Many Worldwide retail, GS1-approved – International Standard ISO/IEC 15420 2D barcodes consist of bars, but use both dimensions for encoding. Example Symbology Continuous or discrete Bar type Uses Australia Post barcode Discrete 4 bar heights An Australia Post 4-state barcode as used on a business reply paid envelope and applied by automated sorting machines to other mail when initially processed in fluorescent ink.[40] Codablock Continuous Many Codablock is a family of stacked 1D barcodes (in some cases counted as stacked 2D barcodes) which are used in health care industry (HIBC). Code 49 Continuous Many Various Code 16K The Code 16K (1988) is a multi-row bar code developed by Ted Williams at Laserlight Systems (USA) in 1992. In the US and France, the code is used in the electronics industry to identify chips and printed circuit boards. Medical applications in the USA are well known. Williams also developed Code 128, and the structure of 16K is based on Code 128. Not coincidentally, 128 squared happened to equal 16,384 or 16K for short. Code 16K resolved an inherent problem with Code 49. Code 49's structure requires a large amount of memory for encoding and decoding tables and algorithms. 16K is a stacked symbology.[41][42] DX film edge barcode Neither Tall/short Color print film Intelligent Mail barcode Discrete 4 bar heights United States Postal Service, replaces both POSTNET and PLANET symbols (formerly named OneCode) KarTrak ACI Discrete Coloured bars Used in North America on railroad rolling equipment PostBar Discrete 4 bar heights Canadian Post office POSTNET Discrete Tall/short United States Postal Service (no international standard available) RM4SCC / KIX Discrete 4 bar heights Royal Mail / PostNL RM Mailmark C Discrete 4 bar heights Royal Mail RM Mailmark L Discrete 4 bar heights Royal Mail Spotify codes Discrete 23 bar heights Spotify codes point to artists, songs, podcasts, playlists, and albums. The information is encoded in the height of the bars;[43] so as long as the bar heights are maintained, the code can be handwritten and can vary in color.[44] Patented under EP3444755. A matrix code or simply a 2D code, is a two-dimensional way to represent information. It can represent more data per unit area. Apart from dots various other patterns can be used. Example Name Notes App Clip Code Apple-proprietary code for launching "App Clips", a type of applet. 5 concentric rings of three colors (light, dark, middle).[45] ArUco code ArUco markers are black-and-white square patterns used as visual tags that can be easily detected and identified by a camera. They are commonly used in augmented reality, robotics, and camera calibration to determine the position and orientation of objects. Their design includes error correction, making them reliable even under partial occlusion or in challenging lighting conditions.[46] AR Code A type of marker used for placing content inside augmented reality applications. Some AR Codes can contain QR codes inside, so that AR content can be linked to.[47] See also ARTag. Aztec Code Designed by Andrew Longacre at Welch Allyn (now Honeywell Scanning and Mobility). Public domain. – International Standard: ISO/IEC 24778 bCode A matrix designed for the study of insect behavior.[48] Encodes an 11 bit identifier and 16 bits of read error detection and error correction information. Predominantly used for marking honey bees, but can also be applied to other animals. BEEtag A 25 bit (5x5) code matrix of black and white pixels that is unique to each tag surrounded by a white pixel border and a black pixel border. The 25-bit matrix consists of a 15-bit identity code, and a 10-bit error check.[49] It is designed to be a low-cost, image-based tracking system for the study of animal behavior and locomotion. BeeTagg A 2D code with honeycomb structures suitable for mobile tagging and was developed by the Swiss company connvision AG. Bokode A type of data tag which holds much more information than a barcode over the same area. They were developed by a team led by Ramesh Raskar at the MIT Media Lab. The bokode pattern is a tiled series of Data Matrix codes. Boxing A high-capacity 2D code is used on piqlFilm by Piql AS[50] Cauzin Softstrip Softstrip code was used in the 1980s to encode software, which could be transferred by special scanners from printed journals into computer hardware. Code 1 Public domain. Code 1 is currently used in the health care industry for medicine labels and the recycling industry to encode container content for sorting.[51] ColorCode ColorZip[52] developed colour barcodes that can be read by camera phones from TV screens; mainly used in Korea.[53] Color Construct Code Color Construct Code is one of the few code symbologies designed to take advantage of multiple colors.[54][55] Cronto Visual Cryptogram The Cronto Visual Cryptogram (also called photoTAN) is a specialized color barcode, spun out from research at the University of Cambridge by Igor Drokov, Steven Murdoch, and Elena Punskaya.[56] It is used for transaction signing in e-banking; the barcode contains encrypted transaction data which is then used as a challenge to compute a transaction authentication number using a security token.[57] CyberCode CyberCode is a visual tagging system utilizing 2D barcodes, designed for recognition by standard cameras, enabling the identification and 3D positioning of tagged objects. Its design incorporates visual fiduciary markers, allowing computers to determine both the identity and orientation of objects, making it suitable for augmented reality applications. However, its data capacity is limited to 24 bits, restricting the amount of information each tag can convey. From Sony. d-touch readable when printed on deformable gloves and stretched and distorted[58][59] DataGlyphs From Palo Alto Research Center (also termed Xerox PARC).[60] Patented.[61] DataGlyphs can be embedded into a half-tone image or background shading pattern in a way that is almost perceptually invisible, similar to steganography.[62][63] Data Matrix From Microscan Systems, formerly RVSI Acuity CiMatrix/Siemens. Public domain. Increasingly used throughout the United States. Single segment Data Matrix is also termed Semacode. – International Standard: ISO/IEC 16022. Datastrip Code From Datastrip, Inc. Digimarc code The Digimarc Code is a unique identifier, or code, based on imperceptible patterns that can be applied to marketing materials, including packaging, displays, ads in magazines, circulars, radio and television[64] digital paper patterned paper used in conjunction with a digital pen to create handwritten digital documents. The printed dot pattern uniquely identifies the position coordinates on the paper. Dolby Digital Digital sound code for printing on cinematic film between the threading holes DotCode Standardized as ISS DotCode Symbology Specification 4.0. Public domain. Extended 2D replacement of Code 128 barcode. At this time is used to track individual cigarette and pharmaceutical packages. Dot Code A Also known as Philips Dot Code.[65] Patented in 1988.[66] DWCode Introduced by GS1 US and GS1 Germany, the DWCode is a unique, imperceptible data carrier that is repeated across the entire graphics design of a package[67] EZcode Designed for decoding by cameraphones;[68] from ScanLife.[69] Han Xin code Code designed to encode Chinese characters, invented in 2007 by Chinese company The Article Numbering Center of China, introduced by Association for Automatic Identification and Mobility in 2011 and published as ISO/IEC 20830:2021 in 2021. High Capacity Color Barcode HCCB was developed by Microsoft; licensed by ISAN-IA. HueCode From Robot Design Associates. Uses greyscale or colour.[70] InterCode From Iconlab, Inc. The standard 2D Code in South Korea. All 3 South Korean mobile carriers put the scanner program of this code into their handsets to access mobile internet, as a default embedded program. JAB Code Just Another Bar Code is a colored 2D Code. Square or rectangle. License free MaxiCode Used by United Parcel Service. Now public domain. mCode Designed by NextCode Corporation, specifically to work with mobile phones and mobile services.[71] It is implementing an independent error detection technique preventing false decoding, it uses a variable-size error correction polynomial, which depends on the exact size of the code.[72] Messenger Codes Proprietary ring-shaped code for Facebook Messenger. Defunct as of 2019, replaced by standard QR codes. Micro QR code Micro QR code is a smaller version of the QR code standard for applications where symbol size is limited. Micro PDF417 MicroPDF417 is a restricted size barcode, similar to PDF417, which is used to add additional data to linear barcodes. MMCC Designed to disseminate high capacity mobile phone content via existing colour print and electronic media, without the need for network connectivity NaviLens NaviLens is used by MTA NYC Transit. NexCode NexCode is developed and patented by S5 Systems. Nintendo Dot code Developed by Olympus Corporation to store songs, images, and mini-games for Game Boy Advance on Pokémon trading cards. PDF417 Originated by Symbol Technologies. Public domain. – International standard: ISO/IEC 15438 Ocode A proprietary matrix code in hexagonal shape.[73] Qode American proprietary and patented 2D Code from NeoMedia Technologies, Inc.[69] QR code Initially developed, patented and owned by Denso Wave for automotive components management; they have chosen not to exercise their patent rights. Can encode Latin and Japanese Kanji and Kana characters, music, images, URLs, emails. De facto standard for most modern smartphones. Used with BlackBerry Messenger to pick up contacts rather than using a PIN code. The most frequently used type of code to scan with smartphones, and one of the most widely used 2D Codes.[74] Public domain. – International standard: ISO/IEC 18004 Rectangular Micro QR Code (rMQR Code) Rectangular extension of QR Code Originated by Denso Wave. Public domain. – International standard: ISO/IEC 23941 Screencode Developed and patented[75][76] by Hewlett-Packard Labs. A time-varying 2D pattern using to encode data via brightness fluctuations in an image, for the purpose of high bandwidth data transfer from computer displays to smartphones via smartphone camera input. Inventors Timothy Kindberg and John Collomosse, publicly disclosed at ACM HotMobile 2008.[77] ShotCode Circular pattern codes for camera phones. Originally from High Energy Magic Ltd in name Spotcode. Before that most likely termed TRIPCode. Snapcode, also called Boo-R code Used by Snapchat, Spectacles, etc. US9111164B1[78][79][80] Snowflake Code A proprietary code developed by Electronic Automation Ltd. in 1981. It is possible to encode more than 100 numeric digits in a space of only 5mm x 5mm. User selectable error correction allows up to 40% of the code to be destroyed and still remain readable. The code is used in the pharmaceutical industry and has an advantage that it can be applied to products and materials in a wide variety of ways, including printed labels, ink-jet printing, laser-etching, indenting or hole punching.[41][81][82] SPARQCode QR code encoding standard from MSKYNET, Inc. TLC39 This is a combination of the two barcodes Code 39 and MicroPDF417, forming a 2D pattern. It is also known as Telecommunications Industry Forum (TCIF) Code 39 or TCIF Linked Code 39.[83] Trillcode Designed for mobile phone scanning.[84] Developed by Lark Computer, a Romanian company.[72] VOICEYE Developed and patented by VOICEYE, Inc. in South Korea, it aims to allow blind and visually impaired people to access printed information. It also claims to be the 2D Code that has the world's largest storage capacity. WeChat Mini Program code A circular code with outward-projecting lines.[85] First, second and third generation barcodes GTIN-12 number encoded in UPC-A barcode symbol. First and last digit are always placed outside the symbol to indicate Quiet Zones that are necessary for barcode scanners to work properly EAN-13 (GTIN-13) number encoded in EAN-13 barcode symbol. First digit is always placed outside the symbol, additionally right quiet zone indicator (>) is used to indicate Quiet Zones that are necessary for barcode scanners to work properly "Wikipedia" encoded in Code 93 "*WIKI39*" encoded in Code 39 "Wikipedia" encoded in Code 128 An example of a stacked barcode. Specifically a "Codablock" barcode. PDF417 sample Lorem ipsum boilerplate text as four segment Data Matrix 2D "This is an example Aztec symbol for Wikipedia" encoded in Aztec Code Text 'EZcode' High Capacity Color Barcode of the URL for Wikipedia's article on High Capacity Color Barcode "Wikipedia, The Free Encyclopedia" in several languages encoded in DataGlyphs Two different 2D barcodes used in film: Dolby Digital between the sprocket holes with the "Double-D" logo in the middle, and Sony Dynamic Digital Sound in the blue area to the left of the sprocket holes The QR code for the Wikipedia URL. "Quick Response", the most popular 2D barcode. It is open in that the specification is disclosed and the patent is not exercised.[86] MaxiCode example. This encodes the string "Wikipedia, The Free Encyclopedia" ShotCode sample detail of Twibright Optar scan from laser printed paper, carrying 32 kbit/s Ogg Vorbis digital music (48 seconds per A4 page) A KarTrak railroad Automatic Equipment Identification label on a caboose in Florida In architecture, a building in Lingang New City by German architects Gerkan, Marg and Partners incorporates a barcode design,[87] as does a shopping mall called Shtrikh-kod (Russian for barcode) in Narodnaya ulitsa ("People's Street") in the Nevskiy district of St. Petersburg, Russia.[88] In media, in 2011, the National Film Board of Canada and ARTE France launched a web documentary entitled Barcode.tv, which allows users to view films about everyday objects by scanning the product's barcode with their iPhone camera.[89][90] In professional wrestling, the WWE stable D-Generation X incorporated a barcode into their entrance video, as well as on a T-shirt.[91][92] In video games, the protagonist of the Hitman video game series has a barcode tattoo on the back of his head; QR codes can also be scanned in a side mission in Watch Dogs. The 2018 videogame Judgment features QR Codes that protagonist Takayuki Yagami can photograph with his phone camera. These are mostly to unlock parts for Yagami's Drone.[93] Interactive Textbooks were first published by Harcourt College Publishers to Expand Education Technology with Interactive Textbooks.[94] Some companies integrate custom designs into barcodes on their consumer products without impairing their readability. Some have regarded barcodes to be an intrusive surveillance technology. Some Christians, pioneered by a 1982 book The New Money System 666 by Mary Stewart Relfe, believe the codes hide the number 666, representing the "Number of the beast".[95] Old Believers, a separation of the Russian Orthodox Church, believe barcodes are the stamp of the Antichrist.[96] Television host Phil Donahue described barcodes as a "corporate plot against consumers".[97] Automated identification and data capture (AIDC) Barcode printer Campus card European Article Numbering-Uniform Code Council Global Trade Item Number Identifier Inventory control system Object hyperlinking Semacode SPARQCode (QR code) List of GS1 country codes

^ a b US patent 2612994 ^ "How Barcodes Work". Stuff You Should Know. 4 June 2019. Archived from the original on 5 June 2019. Retrieved 5 June 2019. ^ a b Cranstone, Ian. "A guide to ACI (Automatic Car Identification)/KarTrak". Canadian Freight Cars A resource page for the Canadian Freight Car Enthusiast. Archived from the original on 27 August 2011. Retrieved 26 May 2013. ^ Keyes, John (22 August 2003). "KarTrak". John Keyes Boston photoblogger. Images from Boston, New England, and beyond. John Keyes. Archived from the original on 10 March 2014. Retrieved 26 May 2013. ^ a b Roberts, Sam (11 December 2019). "George Laurer, Who Developed the Bar Code, Is Dead at 94". The New York Times. Archived from the original on 22 June 2020. Retrieved 13 December 2019. ^ Brown, Derrick (Spring 2023). "The Birth of the Barcode". The Journal of the Computer Conservation Society (101). ISSN 0958-7403. ^ Brown, Derrick (20 March 2023). "The Birth of the Barcode". British Computer Society. Archived from the original on 6 August 2024. Retrieved 6 August 2024. ^ Fox, Margalit (15 June 2011). "Alan Haberman, Who Ushered in the Bar Code, Dies at 81". The New York Times. Archived from the original on 24 June 2017. Retrieved 24 February 2017. ^ G. F. (2 November 2017). "Why QR codes are on the rise". The Economist. Archived from the original on 5 February 2018. Retrieved 5 February 2018. ^ Fishman, Charles (1 August 2001). "The Killer App – Bar None". American Way. Archived from the original on 12 January 2010. Retrieved 19 April 2010. ^ a b c d e f Seideman, Tony (Spring 1993), "Barcodes Sweep the World", Wonders of Modern Technology, archived from the original on 16 October 2016 ^ Dunn, Peter (20 October 2015). "David Collins, SM '59: Making his mark on the world with bar codes". technologyreview.com. MIT. Archived from the original on 10 November 2018. Retrieved 2 December 2019. ^ Graham-White, Sean (August 1999). "Do You Know Where Your Boxcar Is?". Trains. 59 (8): 48–53. ^ Laurer, George. "Development of the U.P.C. Symbol". Archived from the original on 25 September 2008. ^ Nelson, Benjamin (1997). Punched Cards To Bar Codes: A 200-year journey. Peterborough, N.H.: Helmers. ISBN 9780911261127. ^ a b Varchaver, Nicholas (31 May 2004). "Scanning the Globe". Fortune. Archived from the original on 14 November 2006. Retrieved 27 November 2006. ^ Rawsthorn, Alice (23 February 2010). "Scan Artists". New York Times. Archived from the original on 18 November 2016. Retrieved 31 July 2015. ^ "World hails barcode on important birthday". ATN. 1 July 2014. Archived from the original on 23 July 2014. Retrieved 15 February 2017. ^ "A Short History of Bar Code". BarCode 1. Adams Communications. Archived from the original on 2 May 2010. Retrieved 28 November 2011. ^ "Barcode". iWatch Systems. 2 May 2011. Archived from the original on 9 January 2012. Retrieved 28 November 2011. ^ Oberfield, Craig. "QNotes Barcode System". US Patented #5296688. Quick Notes Inc. Archived from the original on 31 December 2012. Retrieved 15 December 2012. ^ National Geographic, May 2010, page 30 ^ Hecht, David L. (March 2001). "Printed Embedded Data Graphical User Interfaces" (PDF). IEEE Computer. 34 (3). Xerox Palo Alto Research Center: 47–55. doi:10.1109/2.910893. Archived from the original (PDF) on 3 June 2013. ^ Howell, Jon; Kotay, Keith (March 2000). "Landmarks for absolute localization" (PDF). Dartmouth Computer Science Technical Report TR2000-364. Archived from the original on 1 October 2020. ^ "IATA.org". IATA.org. 21 November 2011. Archived from the original on 4 January 2012. Retrieved 28 November 2011. ^ "Paperbyte Bar Codes for Waduzitdo". Byte magazine. September 1978. p. 172. Archived from the original on 4 July 2017. Retrieved 6 February 2009. ^ "Nokia N80 Support". Nokia Europe. Archived from the original on 14 July 2011. ^ "package overview for mbarcode". Maemo.org. Archived from the original on 7 April 2019. Retrieved 28 July 2010. ^ Sargent, Mikah (24 September 2017). "How to use QR codes in iOS 11". iMore. Archived from the original on 2 October 2017. Retrieved 1 October 2017. ^ "15+ Best Barcode Scanner iPhone Applications". iPhoneness. 3 March 2017. Archived from the original on 2 October 2017. Retrieved 1 October 2017. ^ David, H (28 November 2018), "Barcodes – Validation vs Verification in GS1", Labeling News, archived from the original on 7 June 2020, retrieved 6 June 2020 ^ "Layman's Guide to ANSI, CEN, and ISO Barcode Print Quality Documents" (PDF). Association for Automatic Identification and Data Capture Technologies (AIM). 2002. Archived from the original (PDF) on 10 September 2016. Retrieved 23 November 2017. ^ Zieger, Anne (October 2003). "Retailer chargebacks: is there an upside? Retailer compliance initiatives can lead to efficiency". Frontline Solutions. Archived from the original on 8 July 2012. ^ a b c d e f g h i j k l Corp, Express. "Barcode Glossary | Express". Express Corp. Archived from the original on 11 December 2019. Retrieved 11 December 2019. ^ Bar Code Verification Best Practice work team (May 2010). "GS1 DataMatrix: An introduction and technical overview of the most advanced GS1 Application Identifiers compliant symbology" (PDF). Global Standards 1. 1 (17): 34–36. Archived (PDF) from the original on 20 July 2011. Retrieved 2 August 2011. ^ GS1 Bar Code Verification Best Practice work team (May 2009). "GS1 Bar Code Verification for Linear Symbols" (PDF). Global Standards 1. 4 (3): 23–32. Archived (PDF) from the original on 27 September 2011. Retrieved 2 August 2011.{{cite journal}}: CS1 maint: numeric names: authors list (link) ^ Garner, J (2019), Results of Data Matrix Barcode Testing for Field Applications, Oak Ridge National Laboratory, retrieved 6 June 2020 ^ "Technical committees – JTC 1/SC 31 – Automatic identification and data capture techniques". ISO. 4 December 2008. Archived from the original on 18 October 2011. Retrieved 28 November 2011. ^ Harmon, Craig K.; Adams, Russ (1989). Reading Between The Lines:An Introduction to Bar Code Technology. Peterborough, NH: Helmers. p. 13. ISBN 0-911261-00-1. ^ "Barcoding fact sheet" (PDF). Australia Post. October 2014. Archived from the original (PDF) on 26 November 2024. ^ a b "2-Dimensional Bar Code Page". www.adams1.com. Archived from the original on 7 July 2011. Retrieved 12 January 2019. ^ "Code 16K Specs" (PDF). www.gomaro.ch. Archived (PDF) from the original on 13 July 2018. Retrieved 12 January 2019. ^ Boone, Peter (13 November 2020). "How do Spotify Codes work?". boonepeter.github.io. Archived from the original on 3 May 2023. Retrieved 3 May 2023. ^ "Scan these new QR-style Spotify Codes to instantly play a song". TechCrunch. 5 May 2017. Retrieved 21 August 2022. ^ "Creating App Clip Codes". Apple Developer Documentation. ^ "OpenCV: Detection of ArUco Markers". Open Source Computer Vision. ^ ""AR Code Generator"". Archived from the original on 10 June 2018. Retrieved 29 April 2017. ^ Gernat, Tim; Rao, Vikyath D.; Middendorf, Martin; Dankowicz, Harry; Goldenfeld, Nigel; Robinson, Gene E. (13 February 2018). "Automated monitoring of behavior reveals bursty interaction patterns and rapid spreading dynamics in honeybee social networks". Proceedings of the National Academy of Sciences. 115 (7): 1433–1438. Bibcode:2018PNAS..115.1433G. doi:10.1073/pnas.1713568115. ISSN 0027-8424. PMC 5816157. PMID 29378954. ^ Combes, Stacey A.; Mountcastle, Andrew M.; Gravish, Nick; Crall, James D. (2 September 2015). "BEEtag: A Low-Cost, Image-Based Tracking System for the Study of Animal Behavior and Locomotion". PLOS ONE. 10 (9): e0136487. Bibcode:2015PLoSO..1036487C. doi:10.1371/journal.pone.0136487. ISSN 1932-6203. PMC 4558030. PMID 26332211. ^ "GitHub – piql/Boxing: High capacity 2D barcode format". GitHub. 4 November 2021. Archived from the original on 21 December 2020. Retrieved 26 March 2021. ^ Adams, Russ (15 June 2009). "2-Dimensional Bar Code Page". Archived from the original on 7 July 2011. Retrieved 6 June 2011. ^ "Colorzip.com". Colorzip.com. Archived from the original on 16 December 2014. Retrieved 28 November 2011. ^ "Barcodes for TV Commercials". Adverlab. 31 January 2006. Archived from the original on 8 December 2009. Retrieved 10 June 2009. ^ "About". Colour Code Technologies. Archived from the original on 29 August 2012. Retrieved 4 November 2012. ^ "Frequently Asked Questions". ColorCCode. Archived from the original on 21 February 2013. Retrieved 4 November 2012. ^ "New system to combat online banking fraud". University of Cambridge. University of Cambridge. 18 April 2013. Archived from the original on 26 July 2020. Retrieved 21 January 2020. ^ Cronto Visual Transaction Signing, OneSpan, archived from the original on 6 December 2019, retrieved 6 December 2019 ^ d-touch topological fiducial recognition, MIT, archived from the original on 2 March 2008. ^ d-touch markers are applied to deformable gloves, MIT, archived from the original on 21 June 2008. ^ See Xerox.com Archived 7 June 2009 at the Wayback Machine for details. ^ "DataGlyphs: Embedding Digital Data". Microglyphs. 3 May 2006. Archived from the original on 26 February 2014. Retrieved 10 March 2014. ^ ""DataGlyph" Embedded Digital Data". Tauzero. Archived from the original on 22 November 2013. Retrieved 10 March 2014. ^ "DataGlyphs". Xerox. Archived from the original on 23 November 2012. Retrieved 10 March 2014. ^ "Better Barcodes, Better Business" (PDF). Archived from the original (PDF) on 22 November 2016. Retrieved 19 June 2017. ^ Dot Code A Archived 9 May 2017 at the Wayback Machine at barcode.ro ^ "Dot Code A Patent" (PDF). Archived (PDF) from the original on 10 March 2016. Retrieved 1 September 2017. ^ "GS1 Germany and Digimarc Announce Collaboration to Bring DWCode to the German Market" (Press release). ^ "Scanbuy". Archived from the original on 20 August 2008. Retrieved 28 November 2011. ^ a b Steeman, Jeroen. "Online QR Code Decoder". Archived from the original on 9 January 2014. Retrieved 9 January 2014. ^ "BarCode-1 2-Dimensional Bar Code Page". Adams. Archived from the original on 3 November 2008. Retrieved 10 June 2009. ^ "Global Research Solutions – 2D Barcodes". grs.weebly.com. Archived from the original on 13 January 2019. Retrieved 12 January 2019. ^ a b Kato, Hiroko; Tan, Keng T.; Chai, Douglas (8 April 2010). Barcodes for Mobile Devices. Cambridge University Press. ISBN 9781139487511. ^ "Ocode – Authentifiez vos produits par le marquage d'un code unique". www.ocode.fr (in French). Retrieved 27 November 2023. ^ Chen, Rongjun; Yu, Yongxing; Xu, Xiansheng; Wang, Leijun; Zhao, Huimin; Tan, Hong-Zhou (11 December 2019). "Adaptive Binarization of QR Code Images for Fast Automatic Sorting in Warehouse Systems". Sensors. 19 (24): 5466. Bibcode:2019Senso..19.5466C. doi:10.3390/s19245466. PMC 6960674. PMID 31835866. ^ ""US Patent 9270846: Content encoded luminosity modulation"". Archived from the original on 2 December 2018. Retrieved 1 December 2018. ^ ""US Patent 8180163: Encoder and decoder and methods of encoding and decoding sequence information with inserted monitor flags"". Archived from the original on 2 December 2018. Retrieved 1 December 2018. ^ ""Screen Codes: Visual Hyperlinks for Displays"" (PDF). Archived (PDF) from the original on 11 December 2019. Retrieved 1 December 2018. ^ ""Snapchat is changing the way you watch snaps and add friends"". July 2015. Archived from the original on 27 January 2021. Retrieved 30 August 2017. ^ ""Snapchat Lets You Add People Via QR Snaptags Thanks To Secret Scan.me Acquisition"". 28 January 2015. Archived from the original on 24 February 2017. Retrieved 24 February 2017.

^ ""How Snapchat Made QR Codes Cool Again"". 4 May 2015. Archived from the original on 14 September 2016. Retrieved 24 February 2017. ^ US 5825015, Chan, John Paul & GB, "Machine readable binary codes", issued 20 October 1998 ^ "US Patent 5825015". pdfpiw.uspto.gov. 20 October 1998. Archived from the original on 13 January 2019. Retrieved 12 January 2019. ^ "Understanding TLC-39 Barcodes: All You Need to Know". 9 August 2023. Retrieved 27 November 2023. ^ "Trillcode Barcode". Barcoding, Inc. 17 February 2009. Archived from the original on 26 July 2020. Retrieved 12 January 2019. ^ "Getting Mini Program Code". Weixin public doc. ^ (株)デンソーウェーブ Archived 7 June 2012 at the Wayback Machine, denso-wave.com (in Japanese) Copyright ^ "Barcode Halls, Standard Facades for Manufacturing Buildings – Projects – gmp Architekten". www.gmp.de. 2009. Archived from the original on 16 December 2023. Retrieved 16 December 2023. ^ "image". Peterburg2.ru. Archived from the original on 10 November 2011. Retrieved 28 November 2011. ^ Lavigne, Anne-Marie (5 October 2011). "Introducing Barcode.tv, a new interactive doc about the objects that surround us". NFB Blog. National Film Board of Canada. Archived from the original on 11 October 2011. Retrieved 7 October 2011. ^ Anderson, Kelly (6 October 2011). "NFB, ARTE France launch 'Bar Code'". Reelscreen. Archived from the original on 10 October 2011. Retrieved 7 October 2011. ^ [1] Archived 16 March 2015 at the Wayback Machine ^ "Dx theme song 2009–2010". YouTube. 19 December 2009. Archived from the original on 11 December 2021. Retrieved 10 March 2014. ^ Diego Agruello (27 June 2019). "Judgment QR code locations to upgrade Drone Parts explained • Eurogamer.net". Eurogamer. Archived from the original on 28 August 2019. Retrieved 3 August 2019. ^ "CueCat History". CueCat History. Archived from the original on 12 November 2019. Retrieved 12 November 2019. ^ "What about barcodes and 666: The Mark of the Beast?". Av1611.org. 1999. Archived from the original on 27 November 2013. Retrieved 14 March 2014. ^ Serafino, Jay (26 July 2018). "The Russian Family That Cut Itself Off From Civilization for More Than 40 Years". Mental Floss. Archived from the original on 7 May 2020. Retrieved 6 May 2020. ^ Bishop, Tricia (5 July 2004). "UPC bar code has been in use 30 years". SFgate.com. Archived from the original on 23 August 2004. Retrieved 22 December 2009. Wikimedia Commons has media related to Barcode. Automating Management Information Systems: Barcode Engineering and Implementation – Harry E. Burke, Thomson Learning, ISBN 0-442-20712-3 Automating Management Information Systems: Principles of Barcode Applications – Harry E. Burke, Thomson Learning, ISBN 0-442-20667-4 The Bar Code Book – Roger C. Palmer, Helmers Publishing, ISBN 0-911261-09-5, 386 pages The Bar Code Manual – Eugene F. Brighan, Thompson Learning, ISBN 0-03-016173-8 Handbook of Bar Coding Systems – Harry E. Burke, Van Nostrand Reinhold Company, ISBN 978-0-442-21430-2, 219 pages Information Technology for Retail:Automatic Identification & Data Capture Systems – Girdhar Joshi, Oxford University Press, ISBN 0-19-569796-0, 416 pages Lines of Communication – Craig K. Harmon, Helmers Publishing, ISBN 0-911261-07-9, 425 pages Punched Cards to Bar Codes – Benjamin Nelson, Helmers Publishing, ISBN 0-911261-12-5, 434 pages Revolution at the Checkout Counter: The Explosion of the Bar Code – Stephen A. Brown, Harvard University Press, ISBN 0-674-76720-9 Reading Between The Lines – Craig K. Harmon and Russ Adams, Helmers Publishing, ISBN 0-911261-00-1, 297 pages The Black and White Solution: Bar Code and the IBM PC – Russ Adams and Joyce Lane, Helmers Publishing, ISBN 0-911261-01-X, 169 pages Sourcebook of Automatic Identification and Data Collection – Russ Adams, Van Nostrand Reinhold, ISBN 0-442-31850-2, 298 pages Inside Out: The Wonders of Modern Technology – Carol J. Amato, Smithmark Pub, ISBN 0831746572, 1993 Retrieved from " 2 "PS/2 keyboard" and "PS/2 mouse" redirect here. For the keyboards normally supplied with IBM PS/2 computers, see Model M keyboard. For peripherals and game conversions for the video game console, see PlayStation 2. The PS/2 port is a 6-pin mini-DIN connector used for connecting keyboards and mice to a PC compatible computer system. Its name comes from the IBM Personal System/2 series of personal computers, with which it was introduced in 1987. The PS/2 mouse connector generally replaced the older DE-9 RS-232 "serial mouse" connector, while the PS/2 keyboard connector replaced the larger 5-pin/180° DIN connector used in the IBM PC/AT design. The PS/2 keyboard port is electrically and logically identical to the IBM AT keyboard port, differing only in the type of electrical connector used. The PS/2 platform introduced a second port with the same design as the keyboard port for use to connect a mouse; thus the PS/2-style keyboard and mouse interfaces are electrically similar and employ the same communication protocol. However, unlike the otherwise similar Apple Desktop Bus connector used by Apple, a given system's keyboard and mouse port may not be interchangeable since the two devices use different sets of commands and the device drivers generally are hard-coded to communicate with each device at the address of the port that is conventionally assigned to that device. (That is, keyboard drivers are written to use the first port, and mouse drivers are written to use the second port.[1]) PS/2 port The color-coded PS/2 connection ports (purple for keyboard and green for mouse)Type Keyboard and computer mouse data connectorProduction historyDesigner IBMDesigned 1987; 38 years ago (1987)Superseded DIN connector, DE-9 connector and Mini-DIN-9 InPortSuperseded by USB (USB human interface device class)General specificationsHot pluggable NoExternal YesCable 4 wires plus shieldPins 6Connector Mini-DIN-6ElectricalSignal 5 V DCEarth Dedicated pin and shieldMax. voltage 5.0±0.5 VMax. current 275 mADataData signal Serial data at 10.0–16.7 kHz with 1 start bit, 8 data bits (LSB first), 1 parity bit (odd), 1 stop bit, [1 ack bit (if host-to-device)]Bitrate 7–12 kbit/sMax. devices 1 or 2[a]Protocol SerialPinout Female connector from the frontPin 1 +DATA DataPin 2 Not connected[b]Pin 3 GND GroundPin 4 Vcc +5 V DC at 275 mAPin 5 +CLK ClockPin 6 Not connected[c] ^ Keyboard and mouse ports may be combined into a single port which can be used to connect both by splitter cable. ^ Sometimes, mouse Data for splitter cable. ^ Sometimes, mouse Clock for splitter cable. Each port implements a bidirectional synchronous serial channel.[2] The channel is slightly asymmetrical: it favors transmission from the input device to the computer, which is the majority case. The bidirectional IBM AT and PS/2 keyboard interface is a development of the unidirectional IBM PC keyboard interface, using the same signal lines but adding capability to send data back to the keyboard from the computer; this explains the asymmetry.[3] The interface has two main signal lines, Data and Clock. These are single-ended signals driven by open-collector drivers at each end. Normally, the transmission is from the device to the host. To transmit a byte, the device simply outputs a serial frame of data (including 8 bits of data and a parity bit) on the Data line serially as it toggles the Clock line once for each bit. The host controls the direction of communication using the Clock line; when the host pulls it low, communication from the attached device is inhibited. The host can interrupt the device by pulling Clock low while the device is transmitting; the device can detect this by Clock staying low when the device releases it to go high as the device-generated clock signal toggles. When the host pulls Clock low, the device must immediately stop transmitting and release Clock and Data to both float high. (So far, all of this is the same as the unidirectional communication protocol of the IBM PC keyboard port, though the serial frame formats differ.) The host can use this state of the interface simply to inhibit the device from transmitting when the host is not ready to receive. (For the IBM PC keyboard port, this was the only normal use of signalling from the computer to the keyboard. The keyboard could not be commanded to retransmit a keyboard scan code after it had been sent, since there was no reverse data channel to carry commands to the keyboard, so the only way to avoid losing scan codes when the computer was too busy to receive them was to inhibit the keyboard from sending them until the computer was ready. This mode of operation is still an option on the IBM AT and PS/2 keyboard port.)[4] To send a byte of data back to the device, the host pulls Clock low, waits briefly, pulls Data low and releases the Clock line again. The device then generates a Clock signal while the host outputs a frame of bits on the Data line, one bit per Clock pulse, similar to what the attached device would do to transmit in the other direction. However, while device-to-host transmission reads bits on falling Clock edges, transmission in the other direction reads bits on rising edges. After the data byte, the host releases the Data line, and the device will pull the Data line low for one clock period to indicate successful reception. A keyboard normally interprets the received byte as a command or a parameter for a preceding command. The device will not attempt to transmit to the host until both Clock and Data have been high for a minimum period of time.[5] Transmission from the device to the host is favored because from the normal idle state, the device does not have to seize the channel before it can transmit—the device just begins transmitting immediately. In contrast, the host must seize the channel by pulling first the Clock line and then the Data line low and waiting for the device to have time to release the channel and prepare to receive; only then can the host begin to transmit data. PS/2 dualport, corresponding splitter (Y-cable) and pinout (female) Older laptops and some contemporary motherboards have a single port that supports either a keyboard or a mouse. Sometimes the port also allows one of the devices to be connected to the two normally unused pins in the connector to allow both to be connected at once through a special splitter cable.[6] This configuration is common on IBM/Lenovo Thinkpad notebooks among many others. The PS/2 keyboard interface is electrically the same as the 5-pin DIN connector on earlier AT keyboards, and keyboards designed for one can be connected to the other with a simple wiring adapter. Such wiring adapters and adapter cables were once commonly available for sale. Note that IBM PC and PC XT keyboards use a different unidirectional protocol with the same DIN connector as AT keyboards, so though a PC or XT keyboard can be connected to PS/2 port using a wiring adapter intended for an AT keyboard, the earlier keyboard will not work with the PS/2 port. (At least, it cannot work with normal PS/2 keyboard driver software, including the system BIOS keyboard driver.) In contrast to this, the PS/2 mouse interface is substantially different from RS-232 (which was generally used for mice on PCs without PS/2 ports), but nonetheless many mice were made that could operate on both with a simple passive wiring adapter, where the mice would detect the presence of the adapter based on its wiring and then switch protocols accordingly. PS/2 mouse and keyboard connectors have also been used in non-IBM PC-compatible computer systems, such as the DEC AlphaStation line, early IBM RS/6000 CHRP machines and SGI Indy, Indigo 2, and newer (Octane, etc.) computers.[7] Macintosh clone computers based on the "LPX-40" logic board design featured PS/2 mouse and keyboard ports, including the Motorola StarMax and the Power Computing PowerBase.[8] PS/2 is now considered a legacy port, with USB ports now normally preferred for connecting keyboards and mice. This dates back at least as far as the Intel/Microsoft PC 2001 specification of 2000. However, as of 2023, although PS/2 ports are rarely included in off the shelf computer systems, they continue to be included on some computer motherboards and are favored by some users for various reasons including the following: PS/2 ports may be favored for security reasons in a corporate environment as they allow USB ports to be totally disabled, preventing the connection of any USB removable disks and malicious USB devices.[9] The PS/2 interface provides no restriction on key rollover, although USB keyboards have no such restriction either, unless operated in BOOT mode, which is the exception. To free USB ports for other uses like removable USB devices. Some USB keyboards may not be able to operate the BIOS on certain motherboards due to driver issues or lack of support. The PS/2 interface has near-universal compatibility with BIOS. The simplicity of PS/2 makes it useful for embedded systems and retro computers that may lack USB host peripheral. This also takes advantage of the wide availability of USB keyboard and mouses with PS/2 compatibility mode. [10] USB mice send data more quickly than PS/2 mice because standard USB mice are polled at a default rate of 125 Hz while standard PS/2 mice send interrupts at a default rate of 100 Hz when they have data to send to the computer.[11][12] Also, USB mice do not cause the USB controller to interrupt the system when they have no status change to report according to the USB HID specification's default profile for mice.[13] Both PS/2 and USB allow the sample rate to be overridden, with PS/2 supporting a sampling rate of up to 200 Hz[2] and USB supporting a polling rate up to 1 kHz[11] as long as the mouse runs at full-speed USB speeds or higher, while USB 2.0 capable devices can support up to 8 kHz polling rates. The USB HID keyboard interface requires that it explicitly handle key rollover, with the full HID keyboard class supporting n-key rollover. However, the USB boot keyboard class (designed to allow the BIOS to easily provide a keyboard in the absence of OS USB HID support) only allows 6-key rollover. Some keyboard peripherals support only the latter class, and some OSes may fail to switch to using the full HID keyboard class with a device after boot.[14] Passive PS/2 to USB adapters Active USB to PS/2 converter Many keyboards and mice were specifically designed to support both the USB and the PS/2 interfaces and protocols, selecting the appropriate connection type at power-on. Such devices are generally equipped with a USB connector and ship with a passive wiring adapter to allow connection to a PS/2 port. Such passive adapters may be specific to the devices they came with; however, the most common configuration involves sharing the GND and Vcc (+5 V) pins of both protocols, connecting the USB D+ pin to the PS/2 +CLK pin, and connecting the USB D- pin to the PS/2 +DATA pin. Using such adapters requires a dual-mode controller on the keyboard or mouse side.[15][16][17] USB to PS/2 passive adapter pinout[15] USB pin Number USB pin Name PS/2 pin Number PS/2 pin Name 1 VBUS 4 Vcc 2 D− 1 +DATA 3 D+ 5 +CLK 4 GND 3 GND Older PS/2-only peripherals can be connected to a USB port via an active converter, which generally provides a pair of PS/2 ports (which may be designated as one keyboard and one mouse, even though both ports may support both protocols) at the cost of one USB port on the host computer.[18] Main article: PC System Design Guide § Color-coding scheme for connectors and ports Color / Pantone Function Connector on PC Green / 3395C PS/2 mouse / pointing device 6-pin mini-DIN female Purple / 2715C PS/2 keyboard Original PS/2 connectors were black or had the same color as the connecting cable (mainly white). Later, the PC 97 standard introduced a color code: the keyboard port, and the plugs on compliant keyboards, were purple; mouse ports and plugs were green. (Some vendors initially used a different color code; Logitech used the color orange for the keyboard connector for a short period, but soon switched to purple.) Today this code is still used on most PCs. The pinouts of the connectors are the same, but most computers will not recognize devices connected to the wrong port. This section: Hardware issues needs additional citations for verification. Please help improve this article by adding citations to reliable sources in this section: Hardware issues. Unsourced material may be challenged and removed.Find sources: "PS/2 port" – news · newspapers · books · scholar · JSTOR (March 2011) (Learn how and when to remove this message) This section is missing information about i8042 issues as described in Glasgow. Please expand the section to include this information. Further details may exist on the talk page. (December 2020) PS/2 ports are designed to connect the digital I/O lines of the microcontroller in the external device directly to the digital lines of the microcontroller on the motherboard. They are not designed to be hot swappable. Hot swapping PS/2 devices usually does not cause damage because more modern microcontrollers tend to have more robust I/O lines built into them which are harder to damage than those of older controllers;[19] however, hot swapping can still potentially cause damage on older machines, or machines with less robust port implementations. If they are hot swapped, the devices must be similar enough that the driver running on the host system recognizes and can be used with the new device. Otherwise, the new device will not function properly. While this is seldom an issue with standard keyboard devices, the host system rarely recognizes the new device attached to the PS/2 mouse port. In practice most keyboards can be hot swapped but this should be avoided. PS/2 connectors are not designed to be plugged in and out very often, which can lead to bent or broken pins. Additionally, PS/2 connectors only insert in one direction and must be rotated correctly before attempting connection. (If a user attempts to insert the connector in the wrong orientation and then tries to rotate it to the correct orientation without first pulling it out, then bent pins could result.) Most but not all connectors include an arrow or flat section which is usually aligned to the right or top of the jack before being plugged in. The exact direction may vary on older or non-ATX computers and care should be taken to avoid damaged or bent pins when connecting devices. This issue is slightly alleviated in modern times with the advent of the PS/2-to-USB adapter: users can just leave a PS/2 connector plugged into the PS/2-to-USB adapter at all times and not risk damaging the pins this way. A USB-to-PS/2 adapter does not have this problem. In a standard implementation both PS/2 ports are usually controlled by a single microcontroller on the motherboard. This makes design and manufacturing extremely simple and cheap. However, a rare side effect of this design is that a malfunctioning device can cause the controller to become confused, resulting in both devices acting erratically. (A well designed and programmed controller will not behave in this way.) The resulting problems can be difficult to troubleshoot (e.g., a bad mouse can cause problems that appear to be the fault of the keyboard and vice versa). BIOS interrupt call DIN connector on IBM PC keyboards Bus mouse Connections on mice DE-9 connector USB ^ There is actually no technical reason that either port could not work with either type of device, if appropriate software was written to support that arrangement. ^ a b "The PS/2 Mouse Interface". 1 April 2003. Archived from the original on 16 September 2008. ^ Compare the logic diagrams in the IBM Personal Computer Technical Reference manual with those in the IBM Personal Computer AT Technical Reference manual. ^ IBM Personal Computer Technical Reference, IBM Personal Computer AT Technical Reference ^ IBM Personal Computer AT Technical Reference ^ "PS/2 Keyboard (IBM Thinkpad) Y adapter". RU: Pinouts. Retrieved 14 June 2011. ^ Lenerz, Gerhard (7 November 2006). "Common Input Devices". Hardware. SGIstuff. Archived from the original on 26 June 2007. Retrieved 14 March 2007. ^ "Power Computing PowerBase". Low end Mac. Retrieved 4 April 2011. ^ Anthony, Sebastian (31 July 2014). "Massive, undetectable security flaw found in USB: It's time to get your PS/2 keyboard out of the cupboard". ExtremeTech. Ziff Davis, LLC. Retrieved 26 October 2015. ^ Eater, Ben. "So how does a PS/2 keyboard interface work?". www.youtube.com. YouTube. Retrieved 14 December 2024. ^ a b "Mouse Optimization Guide: Acceleration Fix and Polling Rate". eu.cybergamer.com. 26 July 2011. Retrieved 16 October 2024. ^ "LCOM 2012/2013: Lab5". web.fe.up.pt. Retrieved 16 October 2024. ^ "Device Class Definition for HID 1.11" (PDF). Archived from the original (PDF) on 11 August 2014. ^ "N-key Rollover via PS/2 and USB". Geek hack. Archived from the original on 25 December 2010. ^ a b "USB to PS/2 mouse or keyboard adapter pinout signals @ PinoutGuide.com". pinoutguide.com. Retrieved 28 October 2024. ^ enCoRe™ USB Combination Low-Speed USB and PS/2 Peripheral Controller. CY7C63722, CY7C63723, CY7C63743 (PDF). Cypress Semiconductor. 27 September 2004. p. 3. ^ USB and PS/2 Multimedia Keyboard Interface. Designer Reference Manual. M68HC08 Microcontrollers (PDF). Freescale Semiconductor. 2006. p. 8. ^ "The pros and cons of PS-2 to USB adapters and converters". TechTarget. ^ Adam Chapweske (5 September 2003). "The PS/2 Mouse/Keyboard Protocol". Archived from the original on 16 November 2016. Retrieved 26 November 2016. Wikimedia Commons has media related to PS/2 connector. "Keyboard and Auxiliary Device Controller" (PDF). Hardware Interface Technical Reference -Common Technical-. IBM. October 1990. PS/2 keyboard and mouse mini-DIN 6 connector pinouts, Burton sys. PS/2 In-depth information, Computer engineering, archived from the original on 1 September 2006, retrieved 11 September 2006. Technical information on Interfacing with the AT keyboard, Beyond logic, archived from the original on 30 August 2018, retrieved 25 March 2012. Retrieved from " 3Learn more about this pageThis article is rated C-class on Wikipedia's content assessment scale.It is of interest to the following WikiProjects:Computing Low-importanceThis article is within the scope of WikiProject Computing, a collaborative effort to improve the coverage of computers, computing, and information technology on Wikipedia. If you would like to participate, please visit the project page, where you can join the discussion and see a list of open tasks.ComputingWikipedia:WikiProject ComputingTemplate:WikiProject ComputingComputingLowThis article has been rated as Low-importance on the project's importance scale.This article is supported by Computer hardware task force (assessed as Mid-importance). Latest comment: 2 years ago4 comments4 people in discussion Yes, it really is true -- at least on some older motherboards, unplugging a PS/2 device while powered will occasionally at least semi-permanently break the motherboard, such as blowing a special fuse. -69.87.204.161 01:03, 19 March 2007 (UTC)Reply Really? I've done it dozens of times and never ever had a single computer fail because of that. I have also plugged mice in the keyboard plug, and the other way around, and I'd swear they worked, but I'm not sure on that one. — isilanes (talk|contribs) 09:45, 12 March 2008 (UTC)Reply As stated, it was primarily on older systems. I had a friend who co-op'd at the school board ~1996 where he did it with an IBM system (which was probably a few years older than that) and burned out the motherboard. On the other hand, the PS/2 connector on my EPoX EP-4B2A2 from 2002 eventually developed a bad connection after a few years and the keyboard would cut in and out, so I ended up having to disconnect and reconnect the keyboard countless times while the system was on, without hurting it. If I was in Windows, what would usually happen is that the typematic rates would get reset and I would have to change them back, otherwise nothing bad happened. Synetech (talk) 19:03, 20 June 2011 (UTC)Reply I used to work on PS/2 keyboard and mouse firmware in the 1990s and never saw a damaged motherboard due to hot plugging. All motherboards and interfaces, going right back to the original IBM systems, included protection devices for static discharge into the pins, overcurrent limiting and wrong polarity. Obviously the protection devices could be damaged with purposeful abuse, e.g. a high negative voltage for a long time into the data lines could overload the protection diodes, but normal use would rarely damage the ports electrically. I did see one motherboard with a fusible protection device on the power line which would require replacement on significant over current. None of these issues would be likely to occur through normal hot plugin. However, some BIOS/hardware could lock out the PS/2 interface if bad signal sequences were received (I saw many times while debugging kbd/mouse firmware). This would have either been a hardware shift register wrong state issue or bad BIOS code or a BIOS protection mechanism, ignoring apparently dodgy signals to prevent rubbish input for the user. Maybe this is what is being referred to? However this was never a hardware issue and caused no damage, only requiring reboot of the host system. Lkingscott (talk) 06:59, 29 September 2022 (UTC)Reply Latest comment: 5 months ago2 comments2 people in discussion I corrected some spelling errors. jpoke89 (talk) 11:49, 22 November 2007 (UTC)Reply Thank you SpiralSource (talk) SpiralSource (talk) 00:05, 24 December 2024 (UTC)Reply Latest comment: 5 years ago2 comments2 people in discussion Care should be taken to avoid connecting a keyboard to a mouse port or vice versa.[citation needed] That's BS. Electrically they're the same. They signal the same. On all modern motherboards both ports even go to the same chip. The data format is different, but you won't harm or break anything by plugging the wrong thing into the wrong port. —Preceding unsigned comment added by 72.29.167.236 (talk) 19:22, 20 September 2008 (UTC)Reply Interestingly the 1st reference on this subject appears to be a comment, rather than a reference... PrimalBlueWolf (talk) 10:59, 25 March 2020 (UTC)Reply Latest comment: 13 years ago3 comments3 people in discussion What was the connector used in the original IBM PC? Family Guy Guy (talk) 04:34, 19 November 2008 (UTC)Reply The IBM XT used a DIN connector - that used a different set of Scan codes. See this if you are interested in some technical details. Megapixie (talk) 04:43, 19 November 2008 (UTC)Reply PC keyboards use the same 5-pin DIN connector as AT keyboards and the pinout is the same as well (although not all ATs supported the RESET signal any more). However, the data going over the interface is different, PC keyboards use 9 bit frames, whereas AT keyboards use 11 bits. Also, PC keyboards send scan codes in what later became known as scan code set 1 (the only set supported by these keyboards), whereas AT keyboards use scan code set 2 by default (and MF keyboard introduced yet another set 3). For compatibility purposes, the keyboard controller on the mainboard translates these interface scancodes into codes according to set 1 for IRQ 1 (INT 9). There are very few exceptions (including some "obscure" IBM machines) where the scancodes are not translated by the keyboard controller ("pass-through mode"). In these cases, AT keyboards will have to be forced into scan code set 1 or the INT 9 interrupt handler of the keyboard driver will have to translate the codes itself. Scan code set 3 was meant to be used by PS/2 machines, but it is rarely used. While all MF keyboards supported all three scan code sets in the early 1990s, later low-quality keyboards were "optimized" to only support scan code set 2.--Matthiaspaul (talk) 01:26, 23 January 2012 (UTC)Reply Latest comment: 12 years ago4 comments3 people in discussion The article should contain a section in which the compatibility of PS/2-to-USB and vice versa is discussed. Simple adapters exist both for plugging PS/2-hardware into USB sockets, as well as for plugging USB-hardware into PS/2 sockets. -- Alexey Topol (talk) 16:31, 27 March 2010 (UTC)Reply Yes, a section about adapters is appropriate. I bought an inexpensive adapter to connect a Logitech M-S48 PS/2 mouse and a Microsoft PS/2 Intellimouse to USB sockets. Neither mouse worked with the adapter. After the Logitech mouse was connected through the adapter for about 10 minutes, the Y axis photodetector overheated enough to melt the window and melt three spokes of the encoder wheel. The MS mouse survived a shorter connection time. Only an adapter from a reputable supplier should be used. An inexpensive nameless adapter can ruin a mouse. Conceivably, it can even cause a fire. Regards, PeterEasthope (talk) 21:10, 26 April 2013 (UTC)Reply In case anyone is interested, this is the link to the faulty adapter. Hopefully it is removed from the market soon. Regards, ... PeterEasthope (talk) 02:40, 27 April 2013 (UTC)Reply This sort of thing really isn't WP's job. We're not a product review or advice site. And everything written in WP must be verifiable by reference to reliable sources, regardless of editors' personal knowledge. If you can find an RS that says something like "some PS/2 to USB adapters have damaged the mice plugged into them", that could be included with the cite of the RS. If you can find a review site that approves of one adapter or disapproves of another, a link thereto might find a place in the "External links" section. But I'm afraid your own report of the faulty adapter is, by WP standards, pure original research and therefore not acceptable for inclusion. Jeh (talk) 04:49, 27 April 2013 (UTC)Reply Latest comment: 12 years ago29 comments6 people in discussion Really? I've not bought a retail PC for years but almost no mobo's for self build had PS/2 ports when I was self building in '08. Anyone else think this is a bit out dated and that now almost all KB and mouse are usb as well as PS/2 being largely dropped from new pc's? —Preceding unsigned comment added by 77.197.60.204 (talk) 12:38, 25 August 2010 (UTC)Reply No, not really. I just looked through like 10-15 high-end Socket 1366 motherboards and not one was lacking a PS/2 port. It seems to be the same story with cheaper motherboards also. Retail PC's are another matter though and may leave them out for lack of space on the I/O panel. So "almost all desktop computers" is definitely an exaggeration but PS/2 (at least for keyboards) is alive and kicking and as it seems is now partly geared toward enthusiast users who are aware of the advantages of PS/2 that exist over the USB and who also might not want to give up old PS/2 keyboards they have owned for a long time, and still kept for compatibility reasons as well. There is the problem of USB keyboards sometimes not being usable at boot which in those cases makes PS/2 absolutely necessary. 83.226.206.82 (talk) 00:16, 5 October 2010 (UTC)Reply That, and who wants to waste a perfectly usable USB port when there's a PS/2 port available? USB ports are in high demand on my system, and I'm sure plenty of people agree. 207.144.204.209 (talk) 22:27, 7 January 2011 (UTC)Reply newegg PS/2 stats TOTAL PS/2 #WITHOUT %WITH AMD 148 3 98 95 Intel 317 23 93 I went through every AMD and Intel mobo listing at newegg.com the other day before I removed the (unreferenced) blurb about PS/2 ports "frequently" not being included (results at left).I didn't bother with the "Motherboard / CPU / VGA Combo" and "Server Motherboards" categories, but 100% of the "Top Sellers" in them have PS/2 ports.As for retail PCs, I bought a compact tower Acer from BestBuy last year that has seven USB ports and individual PS/2 ports for keyboard and mouse (as well as many other ports). ¦ Reisio (talk) 23:29, 7 January 2011 (UTC)Reply I've undid the change partly because it removes a lot of other useful information, and partly because it *is* legacy: PC 2001 (and probably even PC99) says "legacy devices such as PS/2-compatible mouse devices" and "legacy devices such as PS/2-compatible keyboard devices" and refers the reader to the Legacy Plug and Play Guidelines. ⇌Elektron 00:23, 31 January 2011 (UTC)Reply Some more research: PC 98 and 99 "prefer" USB to PS/2; only PC 2001 and the Legacy PnP Guidelines (dated before PC98) call PS/2 "legacy". I'm not aware of major advantages of PS/2 over USB except price, and that PS/2 communication is (by default) device-initiated, so you potentially get lower power consumption and latency (but the maximum PS/2 mouse "sample rate" is apparently 200 Hz, so it only really helps button-clicks and keypresses). The USB-keyboard-in-BIOS problem is not an issue if PS/2-less motherboards implement working USB support; I've only noticed problems on boards with PS/2 ports (e.g. the DG33BU does not support keyboards attached via hubs, and many keyboards have integrated hubs; the GA-D525TUD occasionally fails to reset USB at (re)boot, but there might be a BIOS fix; some boards let you disable USB keyboard support which might happen on a CMOS reset). Of course, anecdotally, a friend bought a PC and later noticed that it didn't specify any PS/2 ports, so he bought a converter (the computer arrived with PS/2 ports). ⇌Elektron 01:06, 31 January 2011 (UTC)Reply 1: The survey described above is a couple of years old, and furthermore is original research. 2: Motherboards sold by places like Newegg are one thing. Motherboards installed in preassembled PCs from Dell, HP, etc., are another. You can't make any judgments like "the majority of motherboards have PS/2 ports" without considerng them. A quick glance at the machines on display at Fry's last week told me that most of the mass market PCs they were offering had no PS/2 ports. The same is also true of laptops: very few laptops have PS/2 ports any more. My belief is that Newegg sells mostly to hobbyists and gamers who need (or think they need) PS/2 ports to support a beloved old Northgate or Model M keyboard (or for the latency advantage in gaming)... but that of course is WP:OR. 3: regarding the "cn" tag: WP:V says that "Even if you're sure something is true, it must be verifiable before you can add it." When someone writes "the question is how to cite it", that is not reason to remove the "cn" tag, it is reason to keep it in place. 3A: The link that was there previously, a broken link to a product description for one motherboard model, was most certainly not sufficient to back up the claim. 3B: WP:V also says "Attribute all quotations and any material challenged or likely to be challenged to a reliable, published source using an inline citation." Well, I'm challenging this material, so a citation is indeed needed. 3C: And finally, WP:V says "Any material lacking a reliable source directly supporting it may be removed." There wasn't and still isn't a WP:RS for this claim. So any editor would be within their rights to remove the claim outright. A cn tag is a completely reasonable interim step. If anyone wants this claim to stay on the page they need to step up and WP:PROVEIT. Jeh (talk) 07:52, 21 January 2013 (UTC)Reply You'd rather have information that's inaccurate by omission, I suppose? Don't care. Think that violates WP:VERIFY? The omission does also, because I can (and have) verified that the text without the information is inaccurate; and if that weren't enough I could always show you WP:IGNORE or WP:CONSENSUS (no, you against me [and all these other editors] does not mean you have a consensus, in case you were wondering; in fact it makes a consensus against your position). A simple matter of wording. If you want to change the wording from "motherboards" to "motherboards for individual sale" or the like, be my guest. The burden of proof is, however, on you as well as I; only my end has already been proven. See 1.I didn't add it, and have taken no action against its removal.See 1.See 1. ¦ Reisio (talk) 17:45, 21 January 2013 (UTC)Reply 1: As a matter of fact, yes, omission is preferable to an unreferenced, challenged claim. That is WP policy. "The standard for inclusion is verifiability, not truth." If it can't be verified then it doesn't belong here, no matter how strongly you "know" it's true or how important you think it is to the article. And besides... you might be wrong. 1A: Your claim that an omission violates WP:V, just because of your WP:OR, is absurd. WP:V says that all material here must be verifiable. It in no way says that anything that can be verified must be included. Material verified only by WP:OR? That's the absurd part. 2: I'm not making such a change because there is no support for that assertion either. Your WP:OR, a survey of one vendor's offerings to a particular market segment, is not sufficient and is WP:OR besides. 3: Consensus on WP is about deciding between possible alternative content, not whether or not to follow WP:V. Consensus cannot override WP:V nor any other core policy. Nor can WP:IGNORE. 4: Why do you object so strenuously to a CN tag? Jeh (talk) 07:25, 23 January 2013 (UTC)Reply Almost, consensus is general agreement, and it can be on any matter at all. There is no trump order amongst policies that I'm aware of, though it can safely be said that WP:IGNORE can trump any other policy (unless consensus is against it, obviously, because that's a numbers game).Because as already explained (and clearly evident to the majority), this issue is indisputable. If the tag didn't mean some yahoo would come by later and remove the assertion thinking it was utter nonsense, I wouldn't care; but since that does happen, and without it the article is inaccurate, I do.¦ Reisio (talk) 19:20, 28 January 2013 (UTC)Reply "Indisputable"? I'm sorry but WP does not recognize your or any editor as having the power to declare a matter under dispute as "indisputable." I'm disputing it, based on considerable experience with PCs sold as complete PCs. I'm talking in particular about the models that are installed on the high tens of millions of corporate desktops in the US alone. PS/2 ports are like hen's teeth in that environment and these machines vastly outsell motherboards sold as components or high-end PCs sold to gamers. Re. consensus, it is most decidedly not a "numbers game." WP:CONSENSUS is not based on voting, but on logical arguments: Consensus is determined by the quality of the arguments given on the various sides of an issue, as viewed through the lens of Wikipedia policy. See also WP:NOTDEMOCRACY: Wikipedia is not an experiment in democracy or any other political system. Its primary (though not exclusive) means of decision making and conflict resolution is editing and discussion leading to consensus—not voting. And as for consensus vs. policies, see again WP:CONSENSUS: Consensus among a limited group of editors, at one place and time, cannot override community consensus on a wider scale. For instance, unless they can convince the broader community that such action is right, participants in a WikiProject cannot decide that some generally accepted policy or guideline does not apply to articles within its scope. In short: Nobody has presented a RS for your claim, so it doesn't matter how many people agree with you. So far your argument has consisted of a two-year-old survey of specialty products from one online vendor, followed by numerous instances of "argument by vigorous assertion." WP:RS requires more than that. Jeh (talk) 03:58, 29 January 2013 (UTC)Reply "I'm sorry but..." To quote myself (because you aren't getting it): 'A simple matter of wording. If you want to change the wording from "motherboards" to "motherboards for individual sale" or the like, be my guest.' "Re. consensus..." It certainly is a numbers game, because it takes a certain number of more people in favor of one thing than a certain number of people in favor of another to produce a consensus... but that's irrelevant to the discussion, I just offered it as the only policy I knew of that could trump WP:IGNORE, as no amount of ignoring will override a majority of people against you (something you might try to realize, Jeh, because [again] there is a majority against you on this PS/2 on motherboards matter). "See also..." We're already splitting hairs on the policy trumping game, so this requires no specific comment. "And as..." See preceding. "In short..." Heh. One moment you argue that consensus matters, and the next you say it doesn't. :p Good show.Again, I'll take a two-year-old comprehensive survey of one online vendor over your complete lack of anything at all, and so will any other sensible person. ¦ Reisio (talk) 00:39, 30 January 2013 (UTC)Reply As I said before: I'm not changing it to "motherboards for individual sale" because that claim doesn't have a source either. That change with a CN tag would, however, seem to be appropriate. Of course, so would just adding the CN tag to the existing text. Consensus on Wikipedia is most decidedly not a numbers game nor a matter of voting or of majority rule. This appears to be a core misunderstanding on your part. From WP:CONSENSUS: "The quality of an argument is more important than whether it represents a minority or a majority view." That is, of course, a policy page. This principle appears in numerous other guidelines, for example WP:PNSD: "most decisions on Wikipedia are made on the basis on consensus, not on vote-counting or majority rule." As well as WP:NOTDEMOCRACY, which I quoted before. Clearly consensus is not the same as "the majority" here. They may be equivalent elsewhere, but not on Wikipedia. "Requires no specific comment". Let's review: WP:NOTDEMOCRACY directly counters your claim that a majority is necessary and sufficient to establish consensus, and WP:CONSENSUS specifically refutes your claim that consensus can override other core policies. WP:V is of course a core policy, so even if you had consensus on your side (you don't; see next item) you couldn't claim that you could therefore ignore WP:V. As I have shown a few lines above, what you claim as consensus—a claimed majority on your side—is not considered consensus here. So, yes, what you are calling "consensus" doesn't count. For that matter, you don't even have a majority, since nobody else has spoken up about this in two years; nor did any of the others who were present before offer an opinion on the simple addition of a CN tag, which is the point under discussion here. I'm afraid your two-year-old personally conducted survey of one site is exactly equivalent to "lack of anything at all" as far as WP:RS is concerned. You can't use original research—and it would be pretty tough for you to argue that a survey you yourself conducted isn't WP:OR—to establish verifiability. Finally: You stated before that you didn't want the CN tag because it was just an invitation for someone to come along and delete the disputed text. Well, yes, that is how WP works. "The article will then be inaccurate", you say? Once again, from WP:V: "The standard for inclusion in Wikipedia is verifiability, not truth." (And certainly not "I think the article would be inaccurate without it, so it has to stay in.") If you feel so strongly about this claim being in the article, then find a WP:RS that backs it up. If it's all that true (and notable) then it really ought to be possible to find a citation for it. Jeh (talk) 02:16, 30 January 2013 (UTC)Reply Skimmed that, looked like you were repeating yourself again. You don't seem remotely interested in reason to me. ¦ Reisio (talk) 04:53, 4 February 2013 (UTC)Reply What you think of as "reason," however "reasonable," is not allowed to override WP policy. Your position that the claim is "already proven" is not supported by any possible reading of WP:V or WP:RS. Nor is your position about WP:CONSENSUS. It appears that you have decided to maintain your position and ignore these key policies. Given that, I am taking this to the dispute resolution noticeboard. You'll have a notice on your talk page informing you of the discussion in a few minutes. I should note that you are far from alone - a very large number of editors come in here believing that it is perfectly fine to add to WP stuff they "know" to be true. I certainly did. It took me a while to learn that that just isn't acceptable here, no matter how "reasonable" the claims seem to be. Jeh (talk) 05:40, 4 February 2013 (UTC)Reply Yes, I believe I've already mentioned multiple times that you should note that consensus is against you... you just don't seem to. ¦ Reisio (talk) 18:36, 5 February 2013 (UTC)Reply I have explained above why consensus is absolutely not against me, per WP:CONSENSUS. My position is completely supported by WP policy and you have not offered any meaningful counter to any of those points. Not one. 02:21, 6 February 2013 (UTC) And I have explained to the contrary. You do see, I hope, the circuitousness. ¦ Reisio (talk) 00:54, 7 February 2013 (UTC)Reply Your "explanations to the contrary" consist of stating that you're going to simply ignore those aspects of WP policy that invalidate your position. It remains the case that policy is against you. Consensus here on WP is not established by a majority, and even if it were, consensus cannot override core policies like WP:V. Jeh (talk) 21:54, 7 February 2013 (UTC)Reply Would you accept a wording change to something like "PS/2 connectors are still present on some current PCs and component motherboards"? I believe that that claim is sufficiently noncontroversial that a citation is not needed. I would also ask that you not delete my add regarding availability of new PS/2 compatible mice and keyboards. No, it is not proven, but that is what a CN tag is for. Jeh (talk) 02:21, 6 February 2013 (UTC)Reply I have already twice suggested you could merely change the wording rather than utterly remove the text, but you seemed to have ignored that. I would accept a wording change to something like what I've already mentioned, but not the use of the phrase "present on some current PCs", because as has already been established, it is clear the majority of (at least individually sold) motherboards, and not merely "some", have PS/2 ports. Your addition expressly contradicts that which we have already been discussing, and I'm not sure tagging something {{citation needed}} while simultaneously adding it is not disingenuous; nevermind that you would apparently promote one assertion even though you have no source (and even less of what you call original research) while opposing another that is contradictory. ¦ Reisio (talk) 03:33, 6 February 2013 (UTC)Reply Far from ignoring your suggested alternate wording, I have replied twice that there is no WP:RS evident for your modified claim either. So I would insist on a CN tag there as well. "Some" still allows "most" as a possibility. It doesn't rule out "most". Please note that I am not trying to "utterly remove the text." I just want a CN tag on it. The fact that that claim seems "clear" to you does not establish verifiability as far as WP is concerned. All you have "established" is that your own two-year old survey of one web retailer that caters mostly to the enthusiast market supports the claim. But that result is not representative of all PCs, nor even of all individually sold motherboards, and it is OR besides. It's not something that can be cited, as it is WP:OR. Therefore, a citation to a WP:RS is still needed. And that's all the CN tag means. Jeh (talk) 08:36, 6 February 2013 (UTC)Reply ""Some" still..." Then it may as well be the more accurate "most". "Please note..." I've replied to this sentiment at the DNR discussion. You may be truly content to repeat yourself in not one but two different locations, but I have better things to do with my time (believe it or not :p). ¦ Reisio (talk) 01:03, 7 February 2013 (UTC)Reply But there is a citable source for "some", whereas there is none for "most". Jeh (talk) 21:54, 7 February 2013 (UTC)Reply CN means "it may be that this claim is true, but no reliable, published sources were given to verify it and the information is not considered common knowledge". That is a direct quote from the template description. It goes on to say that the template should be used "when there is a general question of the verifiability of a statement, or when an editor believes that a reference verifying the statement should be provided." This is why I insist on a CN tag for your "most." You see, the CN tag does not mean "I claim it's not true." It does not even mean "I'm doubtful about this." (There is a "doubtful" tag for that.) I'm saying it's not self-evident the way "the sky is blue" is, nor common knowledge like "the moon orbits the Earth" is, so by WP:V it needs to be cited. And the survey you did two years ago, while providing sufficient motivation for the inclusion of the claim, is not something that we can use as a citation nor sufficient to remove the CN tag. If I thought the "most motherboards" claim wasn't true I'd have used a "doubtful" tag instead. That validity of the claim is not what this dispute is about. The fact that it lacks a citation is. Jeh (talk) 08:36, 6 February 2013 (UTC)Reply You seem to be so caught up in bureaucracy that you ignore the meaning of your own words. Reliable. Published. Verifiable. These words have meanings, and had them long before random internet denizens on Wikipedia wrote up articles on them. The information I have provided is reliable, it is published, and it is verifiable (although I can only assume you have not taken the time to verify it). I won't again go into how WP:V does not trump other policies, but I will say I don't see why you bother taking a position no dictionary editor would. It may not be self-evident the way claiming "the sky is blue" is, but it is self-evident all the same, and if you don't see that, you have just been ignoring me this entire time. My survey might not be enough for you, but it is for me. :) As far as I can tell, it is not your clear distinction between {{citation needed}} and {{doubtful}}, but your stubborn inability to accept reality over bureaucracy that motivates you. The validity is the only thing that matters. It is valid, and I wish you'd accept that; and that despite the myriad of Wikipedia bylaws, there is not an appropriate paper to push in every single situation. ¦ Reisio (talk) 01:14, 7 February 2013 (UTC)Reply No, I'm not ignoring you. I'm trying to get you to see that you can't simply ignore WP policy. The "self-evidentiary" nature of your claim is not what is under discussion. What is relevant to inclusion on Wikipedia is whether it is verifiable. I'm afraid that words like "reliable source", "published", and "verifiable" do indeed have specific meanings in the context of WP policy. And regardless of how they are used elsewhere, when writing for WP and defending your work here, you must use WP's meanings. Otherwise your position here may be indefensible. In particular, you write "The validity is the only thing that matters." I tell you three times: WP policy disagrees with you. WP:V states that all material added to WP must be verifiable, according to WP's definition of "verifiable", from what WP considers "reliable, published sources." Otherwise it can't be here, no matter how true it is, no matter how true you "know" it is, no matter how satisfied you are from your own research. That's WP:V in a nutshell. And there is no provision I can find that would allow any other policy to override that. I really think you need to read all of WP:V, and as well, the essay "verifiability, not truth") for some of the rationale behind it. And then you need not to say "but, I can ignore all that, because I know this claim is true." Similarly the criteria for "reliable" generally excludes self-published sources (like online catalogs); see WP:RS. However! I think I found an "out". WP:RS does allow that "Self-published or questionable sources may be used as sources of information about themselves" (emph. added). So a statement along the lines of "At least one PS/2 port can be found on most of the component motherboards offered by one on-line vendor", with a link to Newegg's "motherboards" listings, seems to me to be acceptable. (It's too bad that PS/2 ports are not among their selection criteria for motherboards, as they are for keyboards and mice.) A much stronger statement would be to do this with three vendors and say "several vendors", leaving their names to the refs. Jeh (talk) 21:44, 7 February 2013 (UTC)Reply Since CN does not mean "I don't think this is true," adding and simultaneously tagging something as CN is not at all disingenuous. Adding new text to WP and adding a "doubtful" tag at the same time would have been disingenuous. But CN and "doubtful" do not mean the same thing. Regarding the claim itself: I can provide for the "most currently sold keyboards and mice are not PS/2 capable" claim the exact same level of OR that you provided for the "most PCs have PS/2 ports" claim... only mine is from today, not two years ago. If you do the product searches at Newegg you'll find that PS/2-capable (including both PS/2 only and USB+PS/2) keyboards and mice are in the distinct minority. For keyboards, only 177 products out of 810 total are listed as PS/2-capable; for mice, only 27 out of 634. However, even though that result supports the claim, it's not cite-able, any more than your survey is. A CN tag is therefore still needed. But the claim is now supported at least as well as the "most motherboards" claim, so can I expect that you will not delete it again? If not you'll be quite inconsistent. Jeh (talk) 08:36, 6 February 2013 (UTC)Reply "Since CN..." It would be amusing to see how long an editor including {{citation needed}} with every single addition to any article would go on unimpaired. :p "Regarding the claim..." Then by all means reinsert that claim. See how easy it is to accept a believable claim? "However, even..." Well then by your own protocols you shouldn't have inserted it, and therefore wouldn't have cared that I removed it. ¦ Reisio (talk) 01:25, 7 February 2013 (UTC)Reply As I have explained, my own protocols do allow for adding such material with a CN tag. My OR can establish believability, plausibility, likelihood, etc. But not verifiability. Jeh (talk) 20:31, 7 February 2013 (UTC)Reply Latest comment: 14 years ago2 comments2 people in discussion IMHO this wire color table should be removed completely. There are no standards or conventions for these things in the kinds of places where cheap commodity consumer goods (like computer keyboards) are made. This is just confusing to people who come here and need that kind of information most. The very presence of multiple combinations should be a clue that this isn't really any kind of reliable information. At best, it is a collection of random anecdotes. If people want to know which color is which signal, they should be instructed to verify the pin assignments on the connector. That is the ONLY place that is guaranteed to be correct. Even the silk screen nomenclature on the PC board internal to the keyboard has been known to be incorrect. Rcrowley7 (talk) 20:07, 21 September 2010 (UTC)Reply I agree. Be bold. ⇌Elektron 23:55, 30 January 2011 (UTC)Reply Latest comment: 13 years ago4 comments4 people in discussion Both use 5V DC, how can there be different voltage through an adapter? Zac67 (talk) 19:35, 26 May 2011 (UTC)Reply On a motherboard I have next to me the 5v of both connect to the same point. The ground is the main system ground.--Tagno25 (talk) 04:41, 1 October 2011 (UTC)Reply Both use 5V for power supply (although the original IBM PC apparently used 6V instead for its DIN keyboard interface), but the voltage levels for the data lines are different. The PS/2 interface uses 5V CMOS logic levels, whereas the USB data lines work at a much lower voltage level (however, they are 5V safe as well). PS/2-USB-adapters do not contain any electronics, they are passive adapters using a "tricky" wiring allowing the keyboard to detect if it is connected to a PS/2 port or not. If not, the controller in the keyboard will switch protocols (if the keyboard's firmware has been programmed to support both protocols), thereby giving the illusion of the adapter translating between PS/2 and USB protocols. Of course, this does not work, if the keyboard supports USB only. The PS/2 data lines are pulled up to 5V on the mainboard, but I don't see how this could become a problem because the USB data lines are 5V-safe as per the USB specification, and the keyboard must be designed to withstand this voltage on the data lines. --Matthiaspaul (talk) 00:32, 23 January 2012 (UTC)Reply Thanks for clarifying, I was 'huh'-ing over this one. It could be made clearer in the article - I'll go do this in a min :) ≈ cmn ≈ ("" /✍) 15:43, 8 April 2012 (UTC)Reply Latest comment: 13 years ago6 comments5 people in discussion While hot-plugging is not supported and the system may not detect the new (or even old) device, it can usually be accommodated by pressing the reset button, causing the motherboard to POST and pick up the device on hard-reboot. Synetech (talk) 19:03, 20 June 2011 (UTC)Reply Actually, no. (And what would be the advantage of your method, anyway, if you still have to shut down the operating system?) The problem with the PS/2 port is that it is not hot-plug safe by design. Removing or inserting devices while the system is powered up (no matter if the operating system has been booted or not) can cause physical damage to the keyboard controller or the chipset of the motherboard. Originally, there was no protection for this, you just were not allowed to do it (the same applies to the parallel port, BTW). This should no longer be a real problem in recent motherboards with PS/2 ports, but originally it was - and quite a few people lost their mainboards this way. Hardware manufacturers have tried to make the PS/2 port much more fail-safe by changing the circuitry over the years, but it is still not a garanteed property and the plug itself is electro-mechanically not designed for hot-plugging. --Matthiaspaul (talk) 00:09, 23 January 2012 (UTC)Reply It's not so much the circuitry, but the fact modern PS/2 connectors have a 'make-first break-last' ground shell, which alleviates ungrounded current spikes, which is what destroyed many a mobo in the 90s. With that said, 99.99% of BIOSes check PS/2 status on input, and set registers appropriately (both host and device side), not touching them again until reboot. It's a holdover from the olden days. Modern PS/2 plugs can be hotplugged perfectly safely (from an electromechanical standpoint), just many manafacturers choose not to implement it because it's a legacy port and not worth the cost/effort tradeoff. ≈ cmn ≈ ("" /✍) 15:53, 8 April 2012 (UTC)Reply Hmmm? Make-first break-last ground connections are completely standard and even expected in physical interfaces designed for hot-plugging. Look at the Serial ATA article, for example. Anyway, my understanding is that it is (or rather was) the circuitry. In the original PS/2 and clones that used the same or similar chips, damage would most commonly happen not because the ground shell was guaranteed to contact when nothing else did, but because there was no guarantee on the "make" sequence of the pins. The lengths of the pins weren't staggered, so which "made" first on connection was a matter of happenstance. There was about a 50% chance that data pins would be connected before power, with the unhappy result that the keyboard would, for a moment, draw power through the data pins. The chip on the mobo couldn't handle that... poof. Jeh (talk) 17:43, 8 April 2012 (UTC)Reply The problem with hot-plugging isn't about power being drawn through data lines - as long as these have short-circuit protection nothing much can go wrong there (presuming the connector isn't plugged shifted or twisted). The point is, if the ground connection has not been made yet while you connect data lines you'll have the potential equalization voltage and current present on the data pins which can easily kill circuitry. Connecting power before everything else is not essential but a good idea (so you can have hardware initialization before mating and avoid garbage on the signals). Furthermore, the line protocol should be designed so that garbage data is detected and discarded while mating. On the PS/2 connector, prior ground contact isn't granted but looking at the construction it is highly likely. Early power contact is nowhere near granted and the protocol isn't designed for hot-plugging either (which you could possibly neglect for keyboard or mouse data). Zac67 (talk) 09:03, 9 April 2012 (UTC)Reply Sorry but the problem was of power drawn through data lines, because the chips didn't have short-circuit protection. And connecting power after ground, but before data lines, would have addressed that issue, so it was a lot more important than simply avoiding data glitches. Jeh (talk) 06:20, 14 April 2012 (UTC)Reply Latest comment: 13 years ago2 comments2 people in discussion I've noticed that newer motherboards tend to have a single combined (half-green, half-purple) PS/2 port instead of separate ones. I've read conflicting information on what exactly they are for. Some people say that they make you choose either a PS/2 keyboard or a PS/2 mouse and have to use a USB one of the other, while (more) people say that you can use a PS/2 splitter (like the kind that laptop users sometimes use) to plug both devices into the dual-port. Motherboard manuals seem to vague or even lacking in their discussion of the combo-port. It would be good if someone with a definitive answer could add it to the article. Synetech (talk) 19:03, 20 June 2011 (UTC)Reply Splitters work by connecting one device as normal and connecting the other to the two spare pins (not sure which is which offhand). It would be down to the individual motherboard manufacturers whether they actually implement those two extra pins. So you aren't going to get a definitive answer only a "it works/doesn't work with board x" answerPlugwash (talk) 04:15, 16 January 2012 (UTC)Reply Latest comment: 13 years ago1 comment1 person in discussion There's note, that windows input driver v.8 no longer supports PS/2, but this topic Input and HID - Architecture and Driver Support shows the opposite. — Preceding unsigned comment added by Bigbes (talk • contribs) 10:56, 31 December 2011 (UTC)Reply Latest comment: 12 years ago2 comments2 people in discussion I seem to remember that the legacy I/O devices take up valuable IRQs. A floppy, a COM port, a Keyboard socket and a mouse socket will take one interrupt each. Several devices can be attached to one USB bus and only take up one interrupt, although they may not perform as fast eg when using a keyboard for gaming. — Preceding unsigned comment added by 92.26.151.41 (talk) 23:23, 22 January 2012 (UTC)Reply Since the full support of PCIbus, APICs, etc., IRQs are no longer "valuable" anyway. Shared interrupts are expected and fully supported on PCI and PCI Express. Jeh (talk) 02:47, 30 January 2013 (UTC)Reply Latest comment: 11 years ago4 comments2 people in discussion I have a Deltaco external hard disk case, and its DC connector is just similar to PS/2, I tested it and a mouse connector fits perfectly. This article has nothing to say about power cable usage. 85.217.42.90 (talk) 17:09, 8 September 2013 (UTC)Reply And it should not. What you're talking about is the physical connector type, which is a six-pin Mini-DIN connector. This article describes one use of such a connector. Similarly we have an article on D-subminiature connectors, and we have articles on Serial port and RS-232, but the latter two do not mention the many other ways 9- and 25-pin D-sub connectors are used. (Not to my recollection - if they do, that should be fixed too.) Jeh (talk) 17:44, 8 September 2013 (UTC)Reply Sorry, I thought this was the main article. Didn't realize PS/2 is one of many. 85.217.42.90 (talk) 00:16, 9 September 2013 (UTC)Reply No problem at all, you pointed out something that IMO should be fixed by renaming this article. Btw I too have a few external HD enclosures with the six-pin mini-DIN power connectors... And I have a few others with a connector that looks identical... but look close and you see there are four thick pins instead of six slender ones. Far better for a power connector, and much more rugged, but nonstandard. Jeh (talk) 17:04, 9 September 2013 (UTC)Reply Latest comment: 10 years ago6 comments5 people in discussion Anent the above, this article is not so much about the physical connector (which, as the IP above pointed out, is used for other things, even in this exact configuration, the standard 6-pin type). It is about the signals and protocols and pinout that appear on the connector in this particular usage. Note that the articles on Serial port and RS-232 are not called "Serial connector" or "RS-232 connector". Accordingly, I think this article would be better named "PS/2 port". I so propose. Please discuss. Jeh (talk) 17:44, 8 September 2013 (UTC)Reply You might be right on this. 85.217.42.90 (talk) 00:17, 9 September 2013 (UTC)Reply I call one of these things a "PS/2 port", and so am in favor of such a change. TOOGAM (talk) 17:56, 16 October 2013 (UTC)Reply Agree. Rename and add either a disambig or a redirect -- Akb4 (talk) 04:35, 5 December 2013 (UTC)Reply The Mini-DIN connector is the physical connector. A PS/2 connector is simply a Mini-DIN connector using specific signals and protocols. So I think the terms PS/2 connector and PS/2 port are just synonyms that mean the same thing, as are Serial connector and Serial port. – Wbm1058 (talk) 14:13, 15 September 2014 (UTC)Reply Well then, you should be happy, as there is already a redirect from PS/2 connector to PS/2 port, just as there is for Serial connector to Serial port. Jeh (talk) 22:59, 15 September 2014 (UTC)Reply Latest comment: 11 years ago1 comment1 person in discussion The article notes that a number of peripherals were shipped with a chipset smart enough to detect whether it was plugged into a ps/2 port or a usb port and could auto-switch between the two, and that such devices generally had usb connectors on them, plus a passive usb to ps/2 adaptor. Is the reverse ever true? Are there devices that have an auto-detecting chipset and a ps/2 plug on them and a passive connector with a ps/2 socket and a usb plug? And are these passive connectors, either usb-plug-becomes-ps/2 plug or ps/2-plug-becomes-usb-plug (if those exist) interchangeable, or is there more than one possible pinout? -- Akb4 (talk) 04:35, 5 December 2013 (UTC)Reply Latest comment: 11 years ago1 comment1 person in discussion I read this as hearsay all the time over the internet and I am perplexed because it simply can't be a hard rule. The USB keyboard I use myself, Microsoft Sidewinder X4, manages to get around this mythical limit somehow, and it isn't simply due to their special keyboard software because it retains its ability to recognize like 15 keys at once in Linux as well. Furthermore the "source" used to back this statement up is a link to another wiki.174.45.212.205 (talk) 01:36, 1 February 2014 (UTC)Reply Latest comment: 11 years ago2 comments2 people in discussion In the "port availability" section: > PS/2 mouse and keyboard connectors have also been used in non-IBM PC-compatible computer systems, This reads as (non-IBM) (PC-compatible computer systems), when in fact it is trying to say (non-(IBM PC)-compatible) (computer systems). 96.50.85.160 (talk) 20:18, 8 April 2014 (UTC)Reply WP:SOFIXIT. Since you spotted the problem, you can probably come up with a wording that fixes it. Jeh (talk) 20:38, 8 April 2014 (UTC)Reply Latest comment: 7 years ago1 comment1 person in discussion Hello fellow Wikipedians, I have just modified one external link on PS/2 port. Please take a moment to review my edit. If you have any questions, or need the bot to ignore the links, or the page altogether, please visit this simple FaQ for additional information. I made the following changes: Added archive //www.usb.org/developers/devclass_docs/HID1_11.pdf to When you have finished reviewing my changes, you may follow the instructions on the template below to fix any issues with the URLs. Y An editor has reviewed this edit and fixed any errors that were found. If you have discovered URLs which were erroneously considered dead by the bot, you can report them with this tool. If you found an error with any archives or the URLs themselves, you can fix them with this tool. Cheers.—InternetArchiveBot (Report bug) 11:32, 5 December 2017 (UTC)Reply Latest comment: 4 years ago2 comments2 people in discussion In the Communication Protocol section, it says "To send a byte of data back to the keyboard, the computer pulls Clock low, waits briefly, then toggles it with a clock signal generated by the computer, while outputting a frame of bits on the Data line, one bit per Clock pulse, (...). The computer releases the Clock line when it is done.". Does this imply that the host generates the clock while in host-to-device transmission? Other sources [1][2][3][4] say that the device is always responsible for generating the clock cycle. When the host wants to transmit, it first pulls the CLOCK low to inhibit the transmission from the device, then it pulls DATA low, releases CLOCK and waits for the device to start generating the clock pulse. [1] 20Keyboard.pdf [2] [3] [4] //retired.beyondlogic.org/keyboard/keybrd.htm Obs.: References [3] and [4] are included as references for this article. I don't know if it is the case, but maybe the host is responsible for generating the clock on an older PS/2 protocol that I am not aware of? If not then the text should be corrected. — Preceding unsigned comment added by 2A00:1398:4:3601:D37B:883D:9C3E:8857 (talk) 21:38, 28 December 2020 (UTC)Reply I completely agree, I've rewritten the text. AFAIK, both AT and PS/2 have always had the device generate the clock. In any case, this is the usual mode, and if there are exceptions they should be called out explicitly. By the way, while I love Beyond Logic, that source ([4]) seems to have a mistake as it says This bit will be read into the keyboard on the next falling edge, after which you can place the next bit of data. whereas the other three sources all agree that on host-to-device transmission, data is clocked in on rising edges instead. Digital Brains (talk) 11:31, 17 January 2021 (UTC)Reply Latest comment: 2 years ago1 comment1 person in discussion When I came across this article I was surprised by the statement that motherboards still commonly include a PS2 port. I checked and as of 2022 this is correct, however as this is date relevant and the ports will most likely disappear or become less common at some point in the future then the statement will then no longer be true. Therefore I have added a date when this is known to be true. This should be updated in the future if the interface continues to be fitted to motherboards. Lkingscott (talk) 07:32, 29 September 2022 (UTC)Reply Add topic Retrieved from " 4The editor will now load. If you still see this message after a few seconds, please reload the page. Retrieved from "