DigiPoints

The Digital Knowledge Handbook

Volume III, Issue 4 July 1999

Inside

A Brief History of the Remote Control

Why Infrared?

Encoding Schemes

Universal Remotes

Anatomy of a Remote: Hardware

Points to Remember during Customer Education

Tips, Techniques, and Trivia

Future of IR Signaling

Future of the Remote Control

Further References

Learning Just Enough to Be Dangerous: Glossary

Testing Your Knowledge

Acknowledgments

Answers to Issue 3-3 Questions

©1999 by the Society of Cable Telecommunications Engineers Inc. All rights reserved.

DigiPoints is published monthly by the Society of Cable Telecommunications Engineers, 140 Philips Road, Exton, PA 19341-1318, 610-363-6888, July 1999, Volume III, Issue 4. Office of publication is 140 Philips Road, Exton, PA 19341-1318. Developed by KnowledgeLink Inc.

Editor: Marvin Nelson, BCE, SCTE

Remote Controls

This issue of *DigiPoints* will discuss a very important customer link to the cable network — the remote control. Although one doesn't usually think of it as an interface device, the remote fulfills that function by providing the ultimate connection between the subscriber and our network. This issue will provide background on how remotes function, how they are built, and what we can expect from remote control technology in the future. Perhaps most importantly to cable technical personnel, we will also give you some helpful hints to aid you in educating your customers. In particular, this issue should help you teach customers how to use their remotes to take advantage of new features and functions available through the cable system.

A Brief History of the Remote Control

In the beginning, there was "Lazy Bones." That was what Zenith Electronics Corp. named their pioneering TV remote control back in 1950. Lazy Bones attached to the TV via a long cable and could change channels by activating a motor

attached to the rotary tuner, as well as turn the set on and off.

While users loved the concept, dangling a wire across the living room clearly left something to be desired. Soon work started on eliminating the cable. In 1955, this resulted in the introduction of the ultrasonic remote, once again pioneered by Zenith. The early devices used a mechanical trigger-like arrangement to generate a distinctive high frequency sound for each key. A special receiver in the TV interpreted the sounds and performed the appropriate function. As an aside, according to company history, the reason Zenith originally went with a purely mechanical implementation was that the marketing people of that era did not believe the consumer would accept the need to replace batteries in a remote control. Be that as it may, by the 1960s battery-powered ultrasonic remotes using electronic rather than mechanical sound generation had become the technology of choice — and would remain so for almost 20 years.

Ultrasonic technology was not without limitations. As the number of functions on the remote grew, more complex receivers were required to reliably distinguish between an ever-increasing number of tones. With the advent of the infrared-emitting LED (Light Emitting Diode) in the 70s, some short-term improvements were gained by modulating the ultrasonic frequencies onto an infrared light signal and using a photodetector for the receiver. However, the number of different commands that could be encoded as discrete frequencies remained a problem, and by the late 1970s, Matsushita, Philips, Zenith and others were all working on signaling schemes that encoded each key as a sequence of infrared light pulses rather than as individual tones. These new techniques supported a much wider range of key functions with the additional benefit of improved reliability, and by the early 1980s, the industry had largely switched to infrared (IR) pulse-encoded transmission formats.

Today, the familiar IR remote control, using some variation of this technique, is ubiquitous in the living rooms of the world.

© SCTE

Why Infrared?

As a cost-effective signaling medium, infrared offers several advantages:

- Low-power infrared is not subject to the same regulatory issues and restrictions as other control technologies such as radio frequency devices.
- Infrared radiation propagates strictly as line of sight. It thus generally stays within the room in which it is used (unless supplemented by a separate wireless RF link). The wireless link allows operation of identical systems in adjacent rooms or dwellings without any special provisions to prevent interaction between units.[1]
- A relatively broad subcarrier modulation frequency range is available using inexpensive transmitter and receiver hardware components — from baseband up to around 500 KHz.
- Infrared is insensitive to interference from external electrical or magnetic fields. (One exception is when the interference is within the same electromagnetic spectrum as the infrared source, such as interference generated by certain types of fluorescent lights. The effects of this type of interference can be countered by various design modifications, which will be discussed later.)
- Energy efficiency is relatively high, which is important for good battery life. (Although remote control infrared transmitters use high pulse currents through the IR-emitting LED — 600 mA or more — overall power consumption is low because the encoding schemes used typically ensure that, on the average, the diodes are not "on" very much of the time.)
- Infrared components, both emitters and detectors, are readily available and inexpensive.

Infrared is not without its limitations, though:

- Line of sight propagation, while an advantage in many cases, becomes a limitation when the user needs to control his device from another room, such as when a single set-top box or satellite receiver is used as the source for multiple TV sets in different rooms.
- Infrared is subject to mutual interference from multiple sources in same room. For practical purposes most consumer IR transmissions use a wavelength of either 880 or 940 nm, corresponding to the two commonly available IR-emitting LED types. While the use of different subcarrier frequencies to modulate the light signals and different encoding schemes to carry the data content will usually prevent false operation, "jamming" of one remote by another is possible when an attempt is made to use two remotes simultaneously. This is usually not an issue when individual remotes are in use. (After all, how many people hold a remote in each hand and try to do two things at once?) But it can be a

1 Some vendors of infrared control devices refer to a separate wireless unit that extends control beyond line of sight as an "IR Blaster." This term is also used to describe wired devices that extend an IR link between a set-top converter and another piece of equipment, such as a VCR. These IR Blasters plug into an IR Blaster port on the source device and contain an LED at the end of a flexible cord. The IR from the LED is used to control the other device. Line of sight between devices is, therefore, not required. On application is for the control needed to start and stop VCR recording based on a subscriber's program choices made through an interactive TV program guide provided via the set-top converter.

© SCTE

problem in applications where a remote is used to send a command to a device that retransmits the signal in a different format to another device. Examples of this situation can be found in some high-end audio systems, or in electronic program guide applications that are not integrated in the same box as the channel tuner.

- Similarly, although largely immune to electrical or magnetic interference, the infrared signal is somewhat subject to interference from sunlight and other infrared sources. The main culprit is often fluorescent lights. All fluorescent lights emit a spectral line of modulated infrared radiation at 1,014 nm, close to the wavelengths used for infrared remote control. In conventional fluorescent lights, the modulation is mostly at line frequency and its harmonics, well below the subcarrier frequencies used by remote controls. However, some of the new high-efficiency compact fluorescent lights emit modulated IR radiation up to 40 KHz and beyond and can be a problem.

Encoding Schemes

Basic Principles

Almost all modern remote control encoding schemes use an infrared signal modulated by a combination of a subcarrier frequency and a control pulse train. The subcarrier frequency is usually in the range of 30 to 60 KHz (although some go as high as 400 KHz or even 1 MHz). Modulating the IR with a subcarrier allows the use of a bandpass filter in the receiver to help eliminate interference from other environmental infrared sources, such as the sun, lighting, etc. The infrared wavelength used is either 940 nm or 880 nm, corresponding to the two commonly available IR LED types. In practice, almost all consumer electronics remote controls use the 940 nm wavelength, while devices conforming to the Infrared Data Association (IrDA) standards for wireless PC-to-PC communication (and some automobile door locks) use 880 nm.

The modulated signal is transmitted as a series of pulses, with the data encoded into either the pulse width or the pulse position, or both. Figure 1 shows a typical transmitter/receiver system in block diagram form. Although Figure 1 shows individual elements for each function in the encoding/decoding process, in modern practice these are usually combined into single components. Thus, transmitter ICs combine a subcarrier generator, pulse encoder, and modulator into a single chip. Integrated receiver modules combine the photodiode, bandpass filter, and detector (and frequently some degree of optical filtration). If a separate optical filter is used at the receiver, this is usually incorporated into the decorative plastic bezel by selecting a bezel material that has maximum translucency at around 940 nm. It is also worth noting that in many formats, the subcarrier is not a 50 percent duty cycle signal[2]. Frequently, a value of around 33 percent or even less is used to extend battery life.

2 Duty cycle is the percentage of cycle time that a voltage is present.

© SCTE

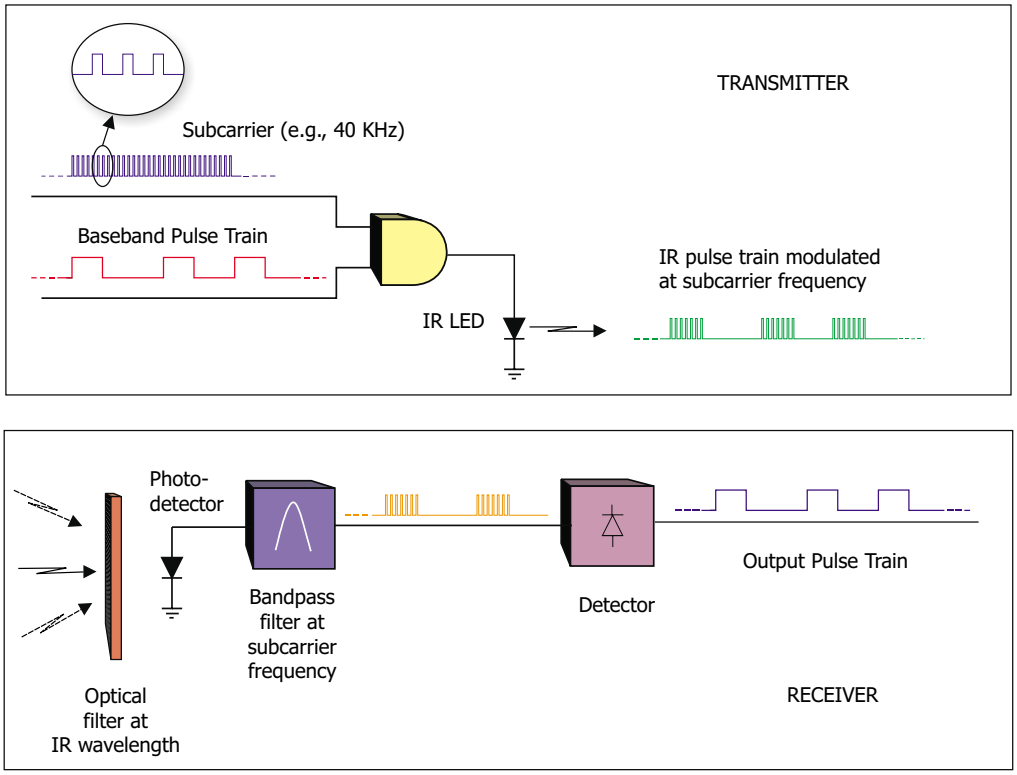


Figure 1: Basic Infrared Transmission

From that simple basis, a multitude of different methods for encoding the actual key data has evolved. In fact, more than 200 different formats are in use today. “Why so many?” you may ask. The reasons are partly historical, and partly that each approach offers its own set of advantages and disadvantages that have been traded off differently by various developers of IR codes.

Historically, much of the original development work in this area was conducted independently by various companies, resulting in multiple solutions to the same problem from the very beginning. Coupled with the fact that each manufacturer generally wants his code to be different from everyone else’s (after all, when you use your remote to turn on your “Brand A” TV set, you don’t want your “Brand B” CD player to respond as well), this has resulted in a wide variety of encoding formats, each with its own set of claimed advantages. These can be broken out into two very broad categories: manufacturer-specific or component-specific. Manufacturer-specific coding is defined by an individual consumer electronics manufacturer to be used across all their product lines (e.g., Sony). Component-specific coding originates with suppliers of IR encoder/decoder ICs and components who offer a range of parts using a predefined encoding scheme (e.g., NEC Semiconductor).

In the sections that follow, we will examine some of these encoding schemes in more detail.

© SCTE

Bit Encoding Format

Figures 2(a) through (d) show a few of the many ways in which binary information can be encoded into infrared pulses. Each figure presents one possible way of representing binary 1s and 0s together with what the baseband bit string "1101" would look like in that format. All of these are representative of popular IR codes in use today. In 2(a), the data is encoded in the spacing of the pulses. In 2(b), the data is encoded in the width of the pulses. Figure 2(c) is an example of Manchester encoding, in which the data value is determined by the direction of the transition (on → off versus off → on) that occurs in the middle of each bit period. The last example shows a slightly more complicated scheme in which a 1 is represented as a double pulse followed by a single pulse, while a 0 is the opposite.

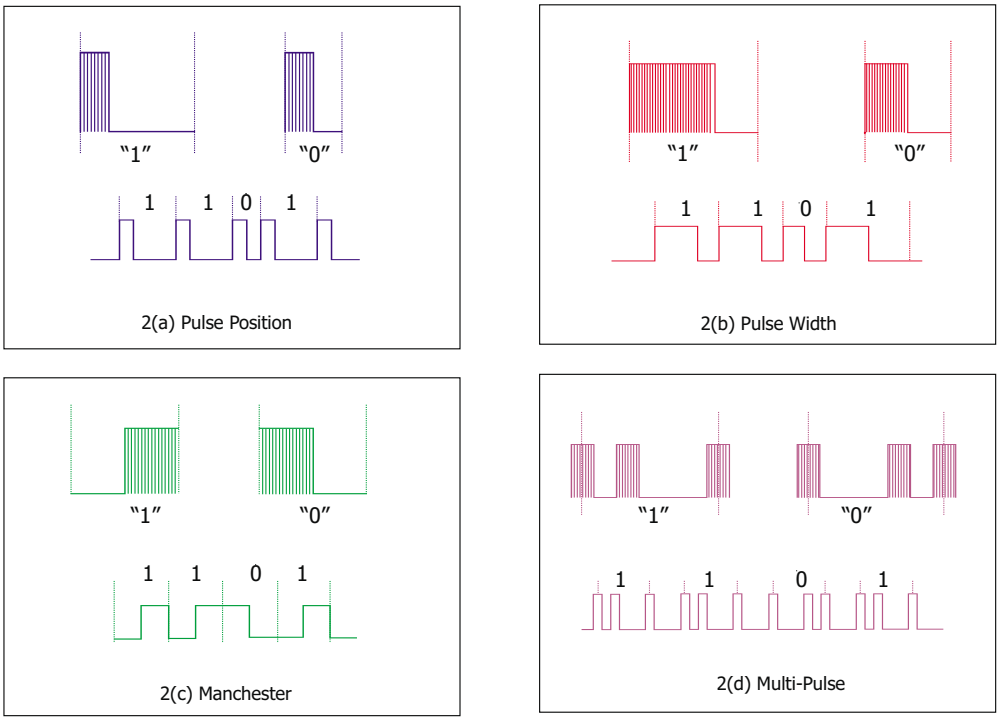


Figure 2: Typical Bit-Encoding Schemes

Just to give a brief idea of the tradeoffs faced by IR code designers even at this level, consider the following differences: Scheme (a) is usually the most energy-efficient. Scheme (b) tends to give better range. Scheme (c) has the advantage of a fixed bit period that makes receiver timing easier and also guarantees that all transmissions are the same duration regardless of the mixture of 1s and 0s in the message. (This is not true for (a) and (b).) Finally, scheme (d) has built-in redundancy that facilitates error detection by the receiver. We will present more on this subject later.

© SCTE

IR Command Structure

Early IR remote control codes sent just five or six bits of information, enough to encode a set of 32 (2^5) or 64 (2^6) key functions as simple binary values. However, as remote control applications proliferated, adding an address field or a "system code" allowed multiple devices using the same basic bit encoding to coexist in the same room. For manufacturer-specific encoding schemes, these fields usually indicate device type (TV, VCR, CD, etc.), giving each class of device a unique system code so that all are able to use the same bit encoding without any possibility of false responses to one another's remote signals. (Some manufacturers may also allocate more than one system code to certain classes of device – VCRs, for example – where there is a possibility that a user may want to use multiple units simultaneously.) In the case of component-specific encoding schemes, these fields usually contain a unique bit pattern allocated by the component manufacturer to each individual customer (frequently referred to as a "customer code"), which prevents interference between different products that both use the same remote control IC.

The data content of most IR codes today consists of a "frame" comprising several fields. A system address field may be anywhere from four to 16 bits long. A key function (command) data field is typically five to eight bits in length. Additional fields have variable lengths as needed for framing and error detection as described in the next section.

Error Detection

As far as error detection is concerned, the remote control is able to take advantage of the ultimate feedback and retry system – the human brain. If the user presses "Channel Up" and the TV doesn't respond, he will usually just press the button again. As long as this doesn't become a routine occurrence, the user is unlikely to become dissatisfied. On the other hand, if the user presses "Channel Up" and the TV turns off, or if it randomly changes channels by itself without being requested, we probably have an irate customer. Accordingly, most remote control data transmission schemes are more concerned with preventing false operation than on attempting to correct errors.

The most basic way to ensure accuracy is to require that the transmitter repeat each IR data frame a minimum number of times. The receiver compares the decoded data from two or more consecutive frames and ignores the signal if they are not identical. Sony, for example, uses this approach in its standard IR protocol (known as "SIRCS" – Sony Infrared Control System). This technique is simple to implement and quite reliable, keeping in mind once again that the objective is to prevent false operation, not to correct errors. All it requires in the remote control design is that the transmitter always send the predetermined minimum number of frames for each keypress, even if the key is released very quickly.

Another popular solution is to send both the command value and its inverse. The receiver then compares the two halves of the received data and ignores the command if they are not exact opposites of one another.

© SCTE

A third method is the use of a checksum[3]. Since the number of check bits needed is fewer than the number of data bits in the command, this generally results in a more compact frame size when compared to sending the data and its inverse. The tradeoff is a somewhat more complex receiver-decoding algorithm. General Instrument, for example, uses this approach. It is also useful in applications where a small frame size is needed to meet critical response time requirements, such as in the special protocol that was designed by Universal Electronics, Inc. for the Full Service Network trials some years back.

Finally, there are those encoding schemes that include built-in redundancy to facilitate error checking without the use of any added data fields. The scheme shown in Figure 2(d), for example, requires that each bit consist of exactly one double pulse and one single pulse within a certain timeframe. The receiver algorithm can use this characteristic to detect and discard any corrupted transmissions.

Format for Repeats

The majority of IR coding schemes will continue to repeat the data frame for as long as a key is held down. (A few do this only for keys that control "ramping" functions such as volume, but they are the exception.) However, implementation of this technique is not as simple as it may seem at first glance. This is because another important aspect of remote control command transmission is the ability of the target device to distinguish between a button that is steadily held down versus a button that is being rapidly pressed and released. This is very significant from a user interface point-of-view. If a user presses, say, the digit "2" twice in rapid succession, he reasonably expects to get "22." But if he slowly and deliberately presses the same key only once, he should get just "2," not "22" or "222," even if his finger lingers on the button a little longer than usual. Of course, with many devices, if you press a key and keep holding it, the receiver after some predetermined delay will begin to interpret your action as a repeated keystroke. An example of this operation is the way the keys on a PC keyboard function. But it is the receiving device, and not the remote, that makes this decision. So the receiver must be able to reliably tell the difference between a key that is pressed and held and a key that is being tapped rapidly.

One simple method used to differentiate between these two occurrences involves frame repetition rate. Transmitted IR data frames are sent at a specific repetition rate for as long as the key is held down. When the key is released and another is pressed, the remote makes sure that the gap between the last frame of the old function and the first frame of the new function is *longer* than the standard repetition rate. All the receiver has to do is monitor the gaps between frames. A longer gap means a new keypress.

Many encoding schemes combine the above approach with a unique start burst sent only at the beginning of the very first frame of data. Often, this burst is of longer duration than the pulses

3 Checksums are discussed in detail in *DigiPoints Volume One*, Chapter 6, "Error Detection and Correction."

© SCTE

used to encode the data and provides a reference level to stabilize the AGC circuit in the receiver, as well as providing a unique prefix to the first data frame.

Another widely used approach is the one implemented in the NEC format. The actual data frame is sent only once in this format, followed by a series of abbreviated repeat signals for as long as the key remains pressed. The advantage of this approach is conservation of battery power. Since the repeat signal consists only of a short "blip" instead of the whole data frame, a key held down for a long period uses less energy than in many other encoding schemes. It does, however, have the disadvantage that if the receiver misses the first frame for any reason, no recovery is possible as the subsequent signal pulses are devoid of any data. The key has to be released and pressed again before anything will happen. Also, error detection when using this approach requires use of either data/inverse or a checksum. Comparison of consecutive frames will not work since by definition the second and subsequent frames contain no data.

Yet another approach involves the use of a so-called "toggle bit." This is a bit at a fixed location in the data frame that toggles back and forth between a 0 and a 1 as each *new* key is pressed. In other words, as long as you press and hold down a single key, the toggle bit remains at the same value in all data frames. But as soon as you release the key and press another (even if it is the same key again), the toggle bit flips. By monitoring this bit, the receiver can detect when one key is released and another pressed. This approach was popular in some older European encoding schemes, but is no longer widely used. It suffers from the disadvantage that, if the toggle bit is strictly interpreted by the receiver, odd behavior can result if the user happens to press a button while the remote is not pointed at the target. In this situation, the next time he uses the remote, the toggle bit will be out of sync. A toggle bit is still present in some modern codes (Philips RC5, for example), but is no longer consistently used to detect key repeats since most of these codes can also be handled by monitoring frame repetition rate as described earlier.

Finally, there are codes that actually use different data values to indicate the start, continuation and end of a keypress. These are not widely used for remote control because they are fairly cumbersome for an application that is simply "press a key, send a code." However, variations of this approach are common in protocols for wireless keyboards that must support more complex functions such as key rollover and multiple key chording (CTRL/ALT/DEL, for example).

Frame Structure

Figure 3 shows a typical infrared remote control data frame embodying the concepts described above. In this example each frame of data comprises 32 bits of information: a 16-bit customer code followed by 8 bits of command data and an 8-bit inverse. Every frame is preceded by a start burst or preamble that uniquely identifies the beginning of a new frame as well as providing an AGC settling signal for the receiver. Following the preamble, bits are encoded using a pulse position scheme similar to that shown in Figure 2(a). Frames repeat at a predetermined rate for as long as a key is held down. (Note that the repetition rate must be measured from start of frame to start of frame. Measuring from the end of one frame to the beginning of the next does not result in a fixed value because the Pulse Position Modulation (PPM) encoding used generates different length frames depending on the mixture of 1s and 0s in the data fields.)

© SCTE

The above example, although not intended to be an exact replica of any specific format, is representative of many of the more popular codes in use on today's equipment. Space does not permit exploration of the more esoteric encoding schemes. Suffice to say that there are as many formats as there are manufacturers, and then some. Well over 200 are in use worldwide.

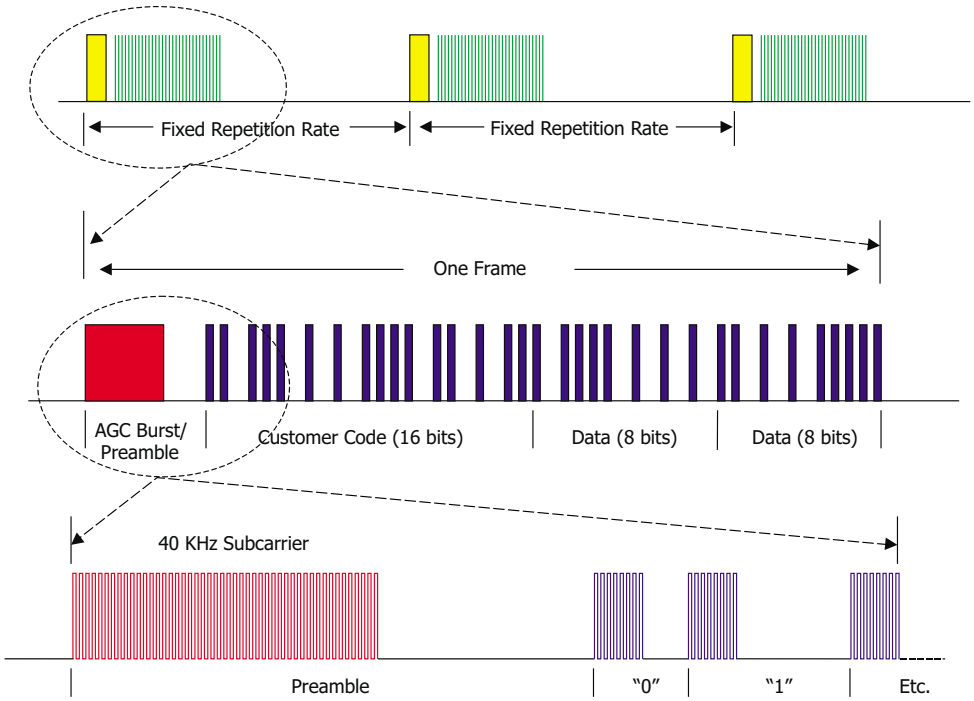


Figure 3: An Example IR Frame Format

Universal Remotes

With so many IR formats in use, it becomes apparent why "universal" remotes have become so popular in recent years. Whether used for consolidation (eliminating "remote clutter" on the coffee table), lost or broken replacement, or simply to group commonly used functions into one easy package, the universal remote has become an expected feature of higher-end consumer electronics offerings.

The first attempts at universality were "learning" remote controls intended to consolidate all of a user's multiple remotes into a single unit. Using a built-in IR receiver, these were designed to learn and store the signals from a user's original remote. Expensive and tedious to program, they never really caught on with the average consumer. And, of course, they were of no use as replacements for lost or broken remotes since the unit needed a functioning original remote to learn from.

Widespread acceptance of the universal remote finally began with the introduction of "pre-programmed" units using a ROM memory preloaded with a complete library of infrared functions. With these units, the user is only required to enter a short "set-up" code to indicate which brand of equipment he wants to control. The first successful products of this type were

© SCTE

introduced by Universal Electronics, Inc. starting in 1988, under the brand names “One For All” at retail and “Uniwand” for the cable industry.

Since that time, consumer awareness of the universal remote has grown steadily. Today, most high-end and many mid-range consumer entertainment devices ship with some form of universal remote included as standard. These run the gamut from highly flexible devices intended to control all facets of a home theater system, all the way down to very simple remotes that offer nothing more than universal volume control for whatever brand of TV the user owns.

Fundamentally, the objective of most universal remotes delivered with equipment (as opposed to purchased as a lost or broken replacement) is to facilitate the user’s everyday viewing experience by consolidating all the frequently used functions of his devices into a single, convenient-to-use remote. Because of the requirement that these units be able to accurately emulate all of the multitude of different IR formats, the microcontrollers and firmware systems used within universal remotes have become quite specialized over the years.

Anatomy of a Remote: Hardware

Basic Configuration

From a practical point of view, there is little observable difference in today’s hardware designs between fixed, single-device remotes and basic, universal remotes. Specialized microcontrollers are available from NEC, Motorola, Samsung, Zilog and others. The extra functionality of the universal units is accomplished largely by built-in features (usually including a more powerful microcontroller CPU core), a sophisticated counter/timer allowing generation of many different subcarrier frequencies, and additional memory space for more firmware and data.

While fixed remotes tend to use 4-bit CPUs with small memories (even by microcontroller standards!), almost all universals use 8-bit microcontrollers similar to those found in pagers, cellular phones, and even some automobile applications. The Motorola 68HC05, Samsung KS86/KS88, and Zilog’s Z86 families are all popular 8-bit choices for universal remotes. All of these manufacturers offer specially configured versions of their basic chips, optimized for universal remote control applications. These typically include all or most of the following:

- 8-bit CPU
- ROM (4 to 32 Kbytes)
- RAM (a few hundred bytes)

© SCTE

- A dedicated key matrix input port with built-in pull-ups[4] and interrupt generation
- Built-in low voltage detection and shutdown (essential in battery-operated devices!)
- A sophisticated counter/timer system with a hardware carrier generator

With this much inside the chip, the external circuitry is fairly simple. Referring to Figure 4, the external circuitry typically consists of little more than a key matrix, IR and visible LEDs, a crystal or ceramic resonator to provide a clock signal for the processor, and batteries. Worth noting is the isolation of the microcontroller input power from the IR LED driver supply (via a diode and a capacitor) so that the high current pulses through the IR LED don't inadvertently trigger the low voltage detector and/or reset the processor. Also shown in Figure 4 is a non-volatile memory (frequently an EEPROM) that is often present on better quality universal units to retain settings when the batteries are changed. The EEPROM may also, in some cases, allow the remote to be upgraded or reconfigured in the field or at the factory.

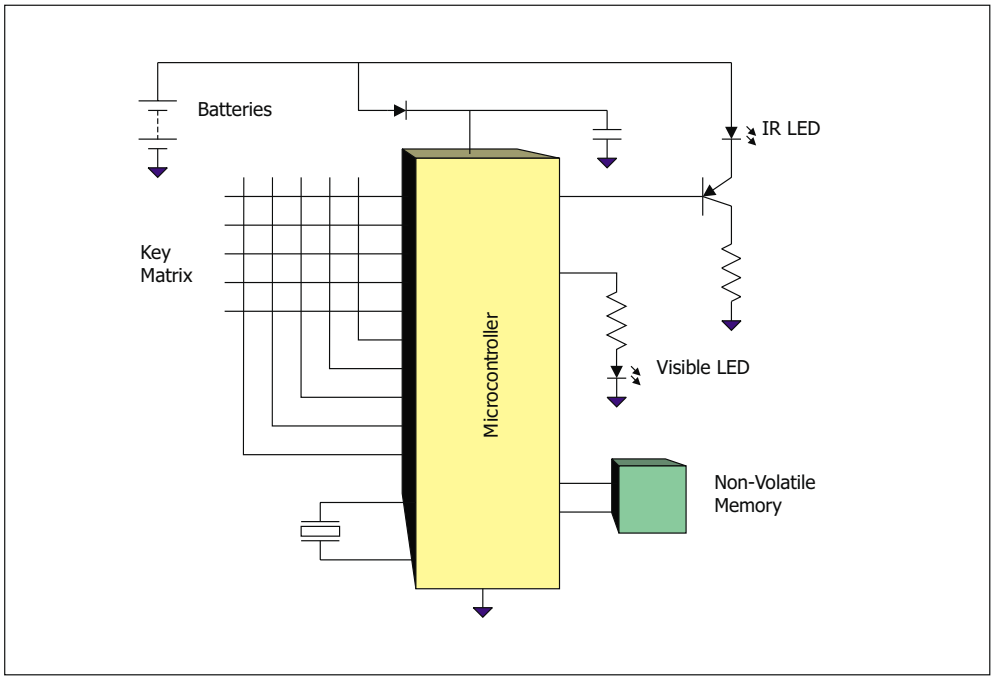


Figure 4: Typical Remote Control Hardware

4 A "pull-up" is a resistor, usually of a relatively high value, connected between a signal input line and the supply voltage. The purpose is to ensure that when no external signal is present, the input remains at a known state rather than "floating" at some indeterminate value.

Other Factors in the Hardware Design

In addition to the components we have just discussed, a remote control needs to have battery power. The ergonomics of most standard size remote controls usually dictate either four AAA or two AA batteries as the power source. At first glance, AAs appear to be the obvious preference with about two-and-a-half times the nominal capacity of AAAs. However, in practical applications, the choice is not always that clear.

A dry cell battery is fully discharged when it reaches about 0.75 V. With two batteries in series, this means an end-of-life voltage of about 1.5 volts. Unfortunately, few, if any, microcontrollers operate at voltages that low. Cut-off at 2.0 or even 2.2 volts is typical, so when using only two batteries, their full capacity is not available to the remote.

In the case of four-battery configurations, end-of-life voltage is around 3.0 volts and the problem does not exist. Thus, in many applications, while two AAs may be adequate, they do not allow the remote to operate as long as a similar remote with a four-battery configuration.

The LED configuration also impacts remote control performance; in fact, it is related to the battery design issue. A remote can have one or two IR LEDs. Many well-performing remotes will have two. This is not just a case of "more is better" – in a good design, these may actually be different components, one "wide angle," usually around 30 degrees, and one "narrow angle," usually around 15 degrees. (The radiation pattern is set by the plastic lens molded into the front part of the diode encapsulation.) This makes for a more "friendly" unit: a user who sits up close to his TV doesn't have to point the remote very accurately, while a user who likes to shoot from across the room still has good range. The concept is like operating the high/low beams on an automobile, except both LEDs are on at the same time. A similar effect can also be obtained by using two narrow angle LEDs physically angled so that their beams either diverge or converge slightly.

In a two-LED remote, the battery configuration affects the choice of how to wire the LEDs. A four-battery remote has sufficient supply voltage headroom to allow the LEDs to be connected in series. In a two-battery remote, on the other hand, the LEDs must be connected in parallel. This parallel connection doubles battery current consumption and is another factor that reduces battery life.

Internal Operation

Microcontrollers suitable for remote control applications use what is known as a "fully static" design. This means that the CPU clock can be completely stopped, and as long as the power to the chip is maintained, the processor will remain "frozen in time," indefinitely holding the last state it was in. All it takes to have the microcontroller pick up again from exactly where it left off is to restart the clock. Importantly, while in this static state, power consumption is almost negligible – on the order of a few microamps. The microcontroller can literally go into hibernation between tasks. If it weren't able to do that, the batteries would last only a few weeks at most.

© SCTE

Accordingly, the remote control design must include a way to make the CPU "wake up" and pay attention when a key is pressed. In most modern ICs, the circuitry to do this is built in. (Some older designs may still use external components, but operation is exactly the same.) Referring to Figure 5, prior to shutting down the microcontroller firmware will configure one side of the key matrix as inputs, and the other side as outputs set to a logic "low" state. These are labeled 11, 12 and 21, 22 in the figure (which, for clarity, shows only a four-key matrix – in actual practice this would be more like 6x6 or 8x8). Built-in pull-ups hold these floating input ports in a logic "high" state. In addition, all the inputs are internally diode "ORed"[5] to an interrupt line that triggers the CPU clock to restart whenever it transitions from high to low. The system remains in this mode, shut down until triggered awake by any key activity as described below.

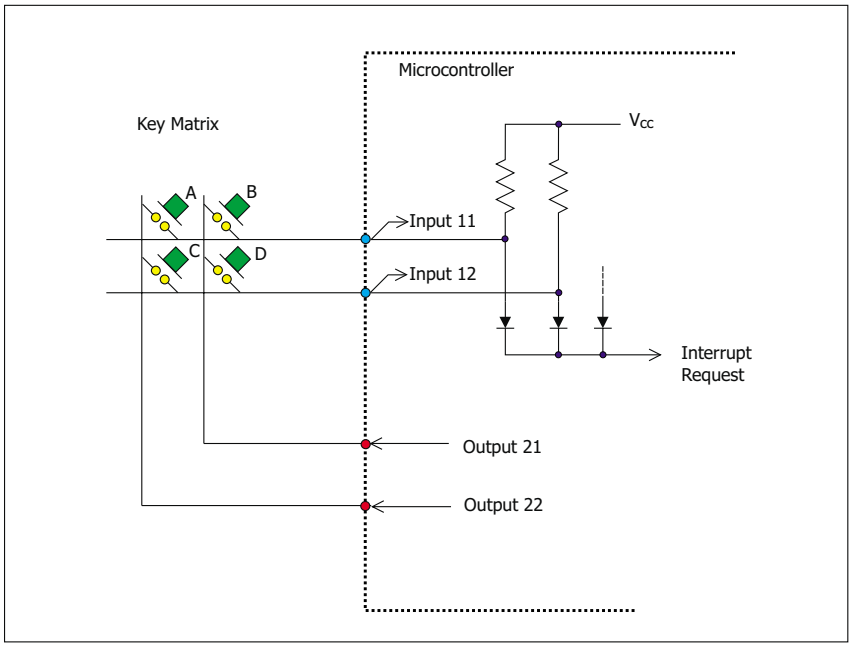


Figure 5: Key Matrix Decoding

When a key is pressed — we will use key "B" in this example — this completes a connection between input pin 11 and output pin 21. Since the output pin is set "low," this causes input 11 to transition from high to low, together with the interrupt request line to which it is connected. This restarts the CPU clock and activates the processor.

At this point, the processor knows that a key has been pressed, but not which one. Looking at Figure 5, if input 11 is low, this could be the result of either key "A" or key "B" being pressed.

5 "OR" is the logic function in which any *one* (or more) of multiple inputs being "true" (usually a binary 1 represented by a certain voltage level) causes the output to be "true." Only if *all* inputs are "false" (binary 0) is the output "false."

The processor now performs a “key scan,” which consists of individually toggling each of the output lines while monitoring the input line, until it finds which output causes the input to toggle with it. With key “B” pressed, this will be output 21.

Once the key has been uniquely identified, the firmware in the microcontroller will now look up the function assigned to that key (and the device code to which it is currently set, if a universal remote) and generate the appropriate IR signal. After the minimum number of data frames has been sent, the key scan will be repeated between successive frames in order to determine whether to continue transmitting or not. Once the keyscan detects that all keys have been released, the system reconfigures itself to the initial state described above and shuts down once again.

Extended Features

Many of today’s remotes go well beyond the basic functionality described above. Space does not permit examination of every possible feature in detail. The following is a brief description of the more significant ones.

Set-Up Retention

Particularly in cable system applications, it is important that universal remotes maintain their set-up parameters when batteries are changed by the user. The most popular method to achieve this is by providing a small non-volatile memory chip outboard of the main processor (usually a serial EEPROM) into which the firmware stores its initial configuration and set-up parameters. Whenever the processor senses that power has been cycled, it reloads its parameters from the non-volatile memory. Another, less reliable (but also less expensive) method sometimes found in retail aftermarket remotes is to install a large capacitor behind the blocking diode (see Figure 4) at the processor V_{dd} input. Since the power consumption of the microcontroller when shut down is extremely low, this capacitor is able to “hold up” the processor RAM for a period of time after the batteries are removed, often several tens of minutes. As long as the user is instructed to always replace batteries promptly, set-up will be retained in the RAM.

Macros

Many remotes include the capability to send a sequence of IR codes in response to a single keystroke. These may be permanently fixed, or factory, installer, or user programmable. Examples range from simple “power on” macros that might perform a sequence such as “Set-top box on, TV on, TV to channel 3.” More complex functions, such as direct tune-in keys for premium channels, are also possible. An example would be to rotate the channel number sent with each successive keypress — to step through HBO1, 2, and 3 for example.

Configurability

Along with the macro capability, the ability to configure specific key layouts to match custom labels for a particular cable operator makes for a more flexible remote. Usually this takes the form of a base remote that can be customized via parameters loaded into its non-volatile memory

© SCTE

at the factory. The remote manufacturer can build a single model and customize it at shipment time by updating the non-volatile memory and top case silk-screening or Mylar overlay – thus being able to offer rapid delivery to meet customer needs.

Upgradeability

Another feature available in remotes equipped with non-volatile memory is the ability to add an entirely new device to the universal code library to supplement those already stored in ROM. For example, adding a new device like a DVD player that has just come on the market. For the cable industry this makes a very flexible remote that can easily be reconfigured for use with any type of set-top box, even if it is a brand new model.

Points to Remember during Customer Education

Since practically all installations of advanced analog or digital set-top boxes now include a universal remote control, we will concentrate on the major issues encountered with this type of unit.

Universality

The first thing to remember is that universal remotes designed for delivery with cable set-top boxes are meant to consolidate common functions needed by the user when viewing cable. They are somewhat more specialized than the ones sold at retail stores as general-purpose replacement units. Retail remotes are usually configured to provide adequate (but not necessarily exhaustive) control of a range of equipment in the consumer's home. The more expensive the unit, the wider the range of equipment types and functions offered, but they all still offer general-purpose coverage.

Cable system universal remotes, on the other hand, are intended to optimize the consumer's cable viewing experience. They must support a complete range of set-top box controls and cable-centric convenience features (such as direct tune-in keys for premium channels) precisely matched to the box and system that they are to be used with. The functions for other devices are usually a basic set intended to facilitate using that device in conjunction with the set-top box, and are not intended to completely replace the device's original remote. Sometimes this subtle difference needs to be explained to the user — especially those customers who already have one or more high-priced aftermarket remotes on their coffee tables.

Mode Switching

Universal remotes come in two forms: "Moded" and "Modeless." A moded remote requires that the user select which device is currently being controlled via a "device" or "mode" key such as "TV," "VCR," "Cable," etc. Modeless remotes, on the other hand, assign predetermined groups of keys permanently to specific devices (e.g., CH+/– and digits to the set-top box; Volume and

© SCTE

Mute to the TV; and Play/Pause, Stop, FF, and Rewind to the VCR). This eliminates the requirement for device keys. Fundamentally, the modeless approach sacrifices flexibility (e.g., you can't change channels on your VCR even if you want to) in return for simplicity of use. Moded remotes are also usually capable of supporting more devices (e.g., audio equipment) in addition to TVs and VCRs. There are arguments both pro and con for moded versus modeless remotes, and in fact, some manufacturers offer both.

Some customers, especially those who have not used a universal remote before, may have difficulty in grasping the concept of "modes" (i.e., that they have to press the "TV" button to control the TV, the "VCR" button to control the VCR and so on). Not to mention the fact that when they pick up the remote after not using it for a while, it will still be set to control whatever they were using last. So, if the remote you are leaving with the customer is a "moded" universal, never assume that he/she has used a universal before and/or will understand this concept without being educated about it.

Customizing

Some higher-end remotes offer features that allow a certain degree of customization in the field. One example is the ability to program the unit to "lock in" the volume control or channel changing to a specific device so that those keys always send that device's code regardless of the current mode. Another example is the ability to program a "power on" sequence so that a single keypress will turn on the entire system. Using these features where appropriate can be a powerful tool in making a customer feel more at ease with the unit.

Set-Up Code Shock

Some customers may balk when confronted with a set-up code listing that includes eight or 10 different possibilities for their brand. "Do you mean I have to try ALL those!?" Not usually. A well-designed set-up code listing will have the entries ordered not in numerical sequence, but by probability. In most cases, anything built within the last 10 years will be found within the first three, or so. And if the remote you are supplying does not have set-up retention, once you find the correct code(s), write them on a scrap of paper and stick it in the battery compartment. It may save a call when the batteries eventually die and the user has lost the manual.

Multiple VCR Codes

When setting up VCR codes on a universal remote (or the set-top, if it has a VCR control feature to go with its onscreen program guide), remember that some manufacturers offer multiple system addresses for their VCRs. If the VCR doesn't want to respond, be sure to try all listed set-up codes. The VCR in question, even though identical to one previously encountered at another customer site, may be configured for a different system address. Furthermore, when setting up the VCR controller feature that comes with some set-tops, resist the temptation to compare set-up codes with those in the accompanying universal remote. This is not necessarily an apples-to-apples comparison. Often the remote will include codes for devices such as camcorders or playback-only decks, which are of no use to the set-top application.

© SCTE

Battery Life

Occasionally, customers may complain that their remotes need new batteries even though the old batteries are still good. This is usually because there will be some charge left after the remote stops working. (Remember the two-battery vs. four-battery operating voltage tradeoff.) A set of batteries that are no longer strong enough to operate a microcontroller may still work for a while in a penlight. The customer should be assured that this is normal, and the remote is functioning as designed.

Two-battery remotes tend to "die" more abruptly than their four-battery counterparts. End of battery life in two-battery remotes usually occurs relatively suddenly with complete loss of function when the low-voltage detect trips. In four-battery remotes, end of battery life is often first seen as a gradual loss of range as the batteries weaken.

Tips, Techniques, and Trivia

Infrared is light in a part of the spectrum not visible to the human eye. In many ways, the output of a remote control behaves much like a beam of visible light. It can be reflected off a mirror, or other shiny surface, and it will pass through clear glass, although glass tends to attenuate IR more than it does visible light. For example, a user is still able to control stereo volume from outside the house on the patio by shooting IR through the family room window. Also, if you need to control something that is just around a corner, try a strategically placed mirror before spending money on a repeater for your remote. If you can see the device to be controlled in the mirror, so can the remote.

If you suspect a remote control is not working, you can use a video camera to observe the IR output of the remote. Since most CCD detectors used in video cameras can "see" into the infrared end of the spectrum, the flashes of IR light should be clearly visible every time a button is pressed by directly observing the light source through the video camera viewer. Each flash is one frame of data. (Individual bit pulses are too rapid for the human eye.)

If an optical filter is used at the receiver, this is sometimes incorporated into the decorative plastic bezel. Some devices will operate poorly, or not at all, if the cover is off or the bezel has been removed. This is especially true if other sources of broad spectrum infrared (e.g., sunlight) are present.

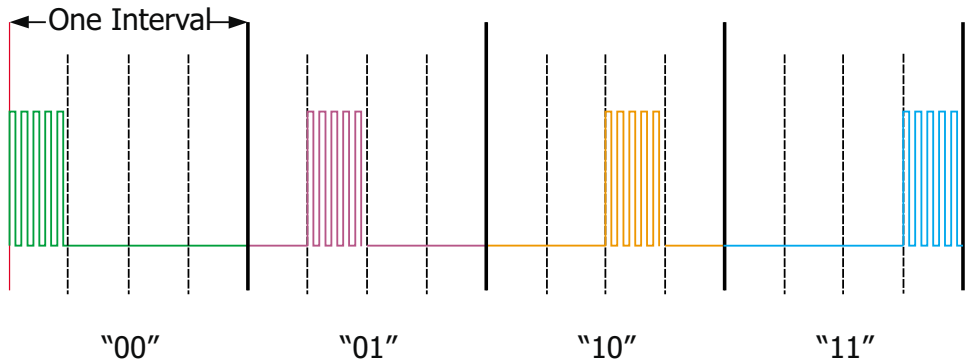
Future of IR Signaling

Regardless of what new applications and features develop over the coming years, the basic set of functions offered by the remote control such as channel switching and volume control will remain central to the consumer entertainment experience. Around this, however, will evolve a whole new layer of required functionality as interactive applications move from the personal computer into the living room.

© SCTE

Much of this additional functionality can initially be accommodated within the framework of current infrared signaling protocols. However, as more complex uses for the remote control develop, new protocols will emerge that support higher data transfer rates as well as two-way capability. This process is already beginning with both proprietary protocols (such as TWIRP – Two-Way IR Protocol) and open standards (such as IrDA Control[6]) starting to be adopted. A common thread through most of these is the use of "multi-PPM" – using a single IR pulse with multiple possible positions to encode more than one bit at a time. The objective of this technique is to reduce energy consumption while increasing bit rate. Applications that require larger amounts of data (program guide information, for example) or longer sessions of user activity (keyboard entry or game playing) thus become possible. An example of this type of encoding is shown in Figure 6.

It is probable that infrared will continue to be able to satisfy the functional requirements of upcoming systems and applications. Given the other cost and regulatory advantages it enjoys, IR is here to stay, at least in the near term.



4-PPM encoding. In each time interval, there are four possible positions in which the pulse can occur. Because there are four possibilities, two bits worth of data can be encoded using a single pulse.

8-PMM (3 bits/pulse) and even 16-PPM (4 bits/pulse) schemes also exist.

Figure 6: Example of Multi-Bit Encoding

6 IrDA – the Infrared Data Association – is a non-profit industry organization that creates and promotes infrared communication standards. The first standard published in 1994 addressed the need for high speed limited distance (3 feet) wireless data interchange between computers (e.g., synchronizing files on a laptop or PDA with a desktop PC, or sending print data from a laptop to a shared spooler). This standard, known as IrDA 1.0 (or 1.1 in a later revision), became loosely referred to as "IrDA." However, IrDA is working on several other standards, including one for control of consumer electronic devices from up to 25 feet away. Recently issued as the "IrDA Control" standard, it is distinct from IrDA 1.0.

© SCTE

Future of the Remote Control

Even though the communication technology may remain infrared, it is reasonable to expect that the remote control itself will become considerably more sophisticated as new applications proliferate. One of the biggest problems facing today's remote control designers is how to add new functionality without increasing the perceived complexity of the remote – it is not acceptable to just keep adding more buttons and labels every time a new application or feature comes along. In the short term some of this is being addressed through the use of colors, "hot keys," on-screen prompts, sliding covers or lids to conceal seldom-used key groupings on the remote. Longer term, there will probably be a move toward different input mechanisms such as touch screens, navigation devices, and the like, as adjuncts to the traditional keys… and, ultimately, perhaps even voice activation.

Increasingly sophisticated remotes cannot, however, become more complex to set up and use. In fact, they will need to become simpler. Many customers already find set-up and operation of universal remotes difficult. Tomorrow's remotes will need to be easier to use and set up if the consumer is to get the full benefit of the capabilities and services available. Leading suppliers of multiple remote control devices have several research initiatives in this area.

Alphanumeric input and PC style pointing/navigation devices will become increasingly important as part of the so-called "convergence" of home applications. Some functions that today are the domain of the PC will likely migrate to the TV – personal e-mail, for example. Other functions that today are the domain of the TV may similarly migrate to the PC, such as acquisition of weather information. Keyboard and pointer inputs will become part of the overall viewing experience, and remote control technology will have to be there to support it.

Further References

The following websites are good sources of more information on remotes and IR protocols in general:

www.chambord-tech.com

www.irda.org

www.twirp.com

www.ueic.com

© SCTE

Learning Just Enough to Be Dangerous: Glossary

AGC – Automatic Gain Control.

AGC Burst – A burst of IR carrier sent at the start of a remote control transmission to alert the receiver and allow it to adjust amplifier gain to match the current distance between the device and the remote.

EEPROM – Electrically Erasable and Programmable Read Only Memory.

Infrared (IR) – Section of the spectrum immediately below lowest visible light (red), generally regarded as including all wavelengths from 760 nm down to about 30 µm.

IrDA – Infrared Data Association. A standards-setting organization concentrating on infrared transmission.

LED – Light Emitting Diode. Note that "light" in this context includes wavelengths both within and outside of the visible spectrum.

Manchester Encoding – A method of encoding data bits in which the value of the bit is indicated by the transition of the carrier (on→off or off→on) that occurs in the middle of each bit time.

Microcontroller – A "single chip" computer in which everything needed (e.g., CPU, RAM, ROM, clock, and input/output) is included in one integrated circuit. This is in contrast to a "microcomputer" which requires external memory chips, input/output, etc. Because they are completely self-contained, microcontrollers are popular in applications where cost, space, or power consumption is a major factor.

nm – Nanometer. One billionth of a meter.

PPM – Pulse Position Modulation. Method of encoding data bits to be transmitted using fixed-length pulses with variable spacing between them. When used alone, "PPM" implies one bit of data is encoded in each pulse. When used with a numeric prefix (e.g., 4-PPM), the number indicates how many different combinations are possible for each pulse/space sequence. Thus 4-PPM can encode two bits in each pulse, 8-PPM can encode three bits, etc.

PWM – Pulse Width Modulation. In contrast to PPM, PWM encodes data bits to be transmitted by varying the pulse length.

Unified Remote – A remote control that can control multiple types of devices, but only of the *same* brand (e.g., a Sony TV and a Sony VCR).

Universal Remote – A remote control that is capable of controlling multiple brands and types of equipment. The key attribute is the ability to control different brands (i.e., to transmit multiple IR formats).

© SCTE

Testing Your Knowledge

1. What type of lighting is most likely to interfere with remote controls? Why?

2. Name three popular bit-encoding schemes currently used for infrared remote controls.

3. What is the difference between a “universal” and a “unified” remote control?

4. What advantage is gained by having non-volatile memory, such as an EEPROM, on a universal remote?

5. What advantage is gained by equipping a remote with two IR LEDs?

6. In the context of remote controls, what is a “macro”?

7. What is a “fully static” microcontroller design? Why is this important for remote controls?

8. What is the advantage of the multi-PPM (4-PPM, 8-PPM) encoding schemes that are being proposed for future applications?

Acknowledgments

SCTE and KnowledgeLink Inc. thank Patrick H. Hayes and Universal Electronics Inc. for their extensive contribution to the content of this issue.

© SCTE

Answers to Issue 3-3 Questions

1. What were the first circuits in televisions to use digital signal processing?

   The first circuits in televisions to use digital signal processing were comb filters and PIP processors.

2. What is a comb filter used for, and what did it replace?

   A comb filter is used to separate luminance (Y) and chrominance (C) components of a composite video signal. It replaced low-pass/bandpass filters.

3. In real-world digital signal processing, what type of representation is normally used for negative (and positive) numbers?

   In real-world digital signal processing, numbers are normally represented using two's complement.

4. In digital video processing, virtually all filters are of what type?

   In digital video processing, virtually all filters are of the Finite Impulse Response (FIR) type.

5. What is the standard measure of circuit complexity in digital signal processing?

   The standard measure of circuit complexity in digital signal processing is gate count.

6. Name three ways to reduce the number of bits of a signal.

   The most common ways to reduce the number of bits of a signal are truncation, symmetric rounding and limiting. Truncation and symmetric rounding are used to eliminate one or more least significant bits (LSBs), while limiting is used to eliminate one or more most significant bits (MSBs).

7. What functional element do digital designers try to eliminate wherever possible?

   Digital designers try to eliminate multipliers wherever possible.

8. What is converting a data stream from one effective sampling frequency to a different effective sampling frequency called?

   Converting a data stream from one effective sampling frequency to a different effective sampling frequency is called sample rate conversion.

© SCTE