UNIVERSAL REMOTE FOR NEXT GENERATION

[1]PALLAVI PATIL, [2]M. A. DIXIT

[1,2]Cummins College of Engineering for Women, Pune
Email: [1]pap2357@gmail.com, [2] mrudul.dixit@cumminscollege.in

Abstract - A universal remote is a remote control that can be programmed to operate various brands of one or more types of consumer electronics devices. For this Philips, Samsung and Sony brands (Protocol) will be implementing to operate devices like TV, DVD etc. The remote stores the code for different appliances in its memory and when a key is pressed retrieves the code from memory and processes it, where normal remotes generate a code, every time a key is pressed.

This paper deals with the hardware and software details for the implementation of Universal Remote Control (URC). Software design of URC is based on Programmable System on Chip (PSoC). The design is implemented using CY8C55 family of PSoC 5 architecture having ARM cortex M3 processor. The code is written in Embedded C, compiled with PSoC creator 2. Device selected for this particular project is CY8C5568AXI-060.

Keywords – ARM cortex M3 processor, Protocol, PSoC #5, URC

I. INTRODUCTION

Universal remote control simplifies our life because it controls many devices regardless of type and brand. It is a device to replace all other remotes used in households and perform all their functions. Normally in homes, remotes are used for appliances like TV, DVD Player, Air conditioner and Music System. Simple universal remote can only control a set number of devices determined by their manufacturer, while extra features added universal remotes allow the user to program in new control codes to the remote. Programming a universal remote can be a fairly complex procedure; it is most often performed by technically minded individuals, although non-technical users can often operate the remote after it has been programmed.

Though the technology is synchronized for all remotes (Infrared Transmission and ON/OFF modulation in the range of 32-36 KHz), there is no agreed convention on code format for data transmission. Communication between remote and appliances is established by predefined code. The problem faced by designers of Universal Remote is the non-standardization of remote control codes [3]. IR based remote control don't have same characteristics like carrier frequencies, existence of lead code, signal definition for logic '1' or '0', signal duration. In the absence of standards, each brand uses its own code. Fortunately, some slack standards exist and the codes of most of the brands loosely conform to these standards.

The Universal Remote proposed in this paper suggests a non-conventional solution to the conventional problem of non-standardization of codes in remotes. Its strength lies in its method of operation.

II. FEATURES AND FUNCTIONS OF URC

- A power button and series of buttons to select which device the remote is controlling at the moment. A typical selection includes TV, VCR, DVD and cable/satellite along with other devices that sometimes include audio equipment or home automation devices.
- Channel and volume up/down selectors (with sign +/-). A numeric keypad for entering channel numbers and for time, date entry.
- User definable (Programmable) buttons are simply blank buttons that can be programmed to control any aspect of the device.
- LCD display for status information. Touch screen technology, allowing navigation through touching the screen rather than using buttons.
- Aliases which allow multiple devices to be accessed without changing device modes (for example, using the TV's volume control while the remote is still in DVD-player mode.)
- Infrared learning or head-to-head learning, enables a user to program certain buttons to perform specific actions. To set up a new command. One can activate infrared learning, hold the remotes head-to-head, and press the button on the old remote to 'teach' the command.

Fig.1 Universal remote showing various devices modes on LCD

III. WORKING PRINCIPLE

The block diagram of the whole system is shown in Fig.2 It consists of a ARM cortex M3 processor, IR Tx and Rx, Capsense, LCD Display with touch screen control and Programmable buttons.

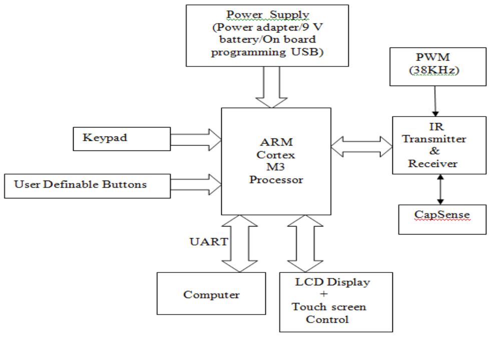

Fig. 2 Block Diagram

A.IR Transmitter and IR receiver:

Almost all audio and video equipment can be controlled using an infrared remote control. At the receiving end, a receiver detects the light pulses, which are processed to retrieve/decode the information they contain.

When an interface used between transmitter and receiver in infrared (IR) then it is called IR remote control. In IR remotes IR LED is used in transmitters which emits IR light. In receivers any IR light detector can be used like IR Sensor, Photo diode, Photo transistor etc.

TV room may have hundreds of tinny IR sources, the lamps around, even the hot cup of tea. A way to avoid all those other sources remote control use to pulsate its infrared in a certain frequency. The IR receiver module at the TV, VCR or stereo "tunes" to this certain frequency and ignores all other IR received. The best frequency is between 30 and 60KHz, the most used is around 36KHz. So, remote controls use the 36KHz (or around) to transmit information. Infrared light emitted by IR Diodes is pulsated at 36 thousand times per second, when transmitting logic level "1" and silence for "0".
To generate a 36KHz pulsating infrared is quite easy, more difficult is to receive and identify this frequency. This is why some companies produce infrared receives, that contains the filters, decoding circuits and the output shaper, that delivers a square wave, meaning the existence or not of the 36KHz incoming pulsating infrared. To avoid a Philips remote control to change channels in a Panasonic TV, it uses different codification at the infrared, even that all use basically the same transmitted frequency, from 36 to 50KHz. So, all use a different combination of bits or how to code the transmitted data to avoid interference.

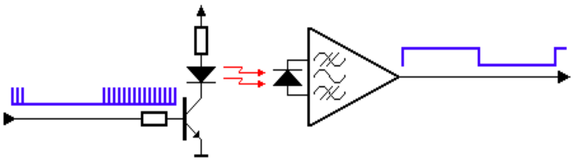

Fig. 3 Simple IR Tx and Rx

In the picture above Fig. 3 shows a modulated signal driving the IR LED of the transmitter on the left side. The detected signal is coming out of the receiver at the other side [4].

B.IR Protocol (Philips RC5):

The protocol uses bi-phase modulation (Manchester coding) of a 36KHz IR carrier frequency. All bits are of equal length of 1.778ms in this protocol, with half of the bit time filled with a burst of the 36KHz carrier and the other half being idle. In the Fig.4 a logical zero is represented by a burst in the first half of the bit time. A logical one is represented by a burst in the second half of the bit time. The pulse/pause ratio of the 36KHz carrier frequency is 1/3 or 1/4 which reduces power consumption.

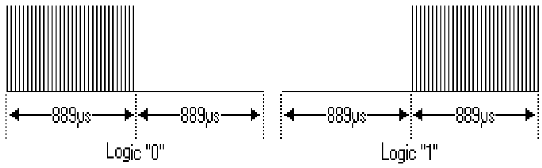

Fig.4 Representation of Logical 0 and logical 1

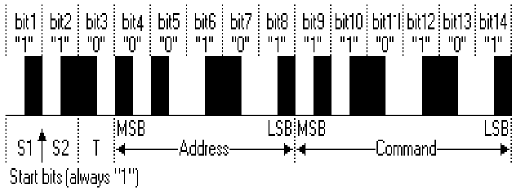

Fig.5 Pulse train of an RC-5 message

As shown in the Fig.5 the first two pulses are the start pulses, and are both logical "1". Half a bit time is elapsed before the receiver will notice the real start of the message. Extended RC-5 uses only one start bit. Bit S2 is transformed to command bit 6, providing for a total of 7 command bits. The value of S2 must be inverted to get the 7th command bit though. The 3rd bit is a toggle bit. This bit is inverted every time a key is released and pressed again. This way the receiver can distinguish between a key that remains down, or is pressed repeatedly. The next 5 bits represent the IR device address, which is sent with MSB first. The address is followed by a 6 bit command, again sent with MSB first. A message consists of a total of 14 bits, which adds up to a total duration of 25 ms. Sometimes a message may appear to be shorter because the first half of the start bit S1 remains idle.

And if the last bit of the message is a logic "0" the last half bit of the message is idle too.

As long as a key remains down the message will be repeated every 114ms. The toggle bit will retain the same logical level during all of these repeated messages. It is up to the receiver software to interpret this auto repeat feature. Philips has created a beautiful list of "standardized" commands. This ensures the compatibility between devices from the same brand [4].

TABLE I
RC5 Address

RC5 Address	Device
$00-0	TV1
$01-1	TV2
$02-2	Teletext
$03-3	Video
$04-4	LV1
$05-5	VCR1
$06-6	VCR2
$07-7	Experimental
$08-8	Sat1
$09-9	Camera
$0A-10	Sat2
$0C-12	CDV
$0D-13	Camcorder
$10-16	Pre-amp
$11-17	Tuner
$12-18	Recorder1
$13-19	Pre-amp
$14-20	CD Player

TABLE II
RC5 Command

RC5 Command	TV Command	VCR Command
$00-0 to $09-9	0 to 9	0 to 9
$0A-10	-/--	-/--
$0C-12	Standby	Standby
$0D-13	Mute	
$10-16	Volume +	
$11-17	Volume -	
$12-18	Brightness +	
$13-19	Brightness-	
$20-32	Program +	Program +
$21-33	Program -	Program -
$32-50		Fast RW
$34-52		Fast FW
$35-53		Play
$36-54		Stop
$37-55		Recording

In this way other protocols for example Samsung, SIRC (Sony), Panasonic etc. can be implemented.

C. Capsense:

The Capsense system provides a versatile and efficient means for measuring capacitance in applications such as touch sense buttons, sliders, proximity detection, etc. It provides capacitance sensing using a switched capacitor technique with a delta-sigma modulator to convert the sensing current to a digital code [2], [5].

D.LCD Display with Touch screen control:

There are universal remote controls in market have too many keys and are equally complicated to use. It becomes very cumbersome to use these remote controls, as user gets confused while operating these controls.

Popularity of iPOD, iPhone, Android phones is due to their rich graphical user interface using very few keys and touch pad. Taking this as clue if one can design a universal remote control which has graphical LCD with touchpad and few keys with the help of which we can make this remote control very intuitive shall be a great solution to offer.

A touch screen is an electronic visual display that can detect the presence and location of a touch within the display area. it enables one to interact directly with what is displayed, rather than indirectly with a pointer controlled by a mouse or touchpad. Also it lets one do so without requiring any intermediate device that would need to be held in the hand (other than a stylus, which is optional for most modern touch screens) [2], [5].

IV. SOFTWARE IMPLEMENTATION DETAILS

Development is done using PSoC #5 development kit shown in Fig.6.With its unique array of configurable blocks, PSoC 5 is a true system-level solution providing microcontroller unit (MCU), memory, analog, and digital peripheral functions in a single chip. The CY8C55 family offers a modern method of signal acquisition, signal processing, and control with high accuracy, high bandwidth, and high flexibility [5].

In addition to communication interfaces, the CY8C55 family has an easy to configure logic array, flexible routing to all I/O pins, and a high-performance 32-bit ARM Cortex -M3 microprocessor core. Designers can easily create system-level designs using a rich library of prebuilt components and boolean primitives using PSoC Creator, a hierarchical schematic design entry tool [5].

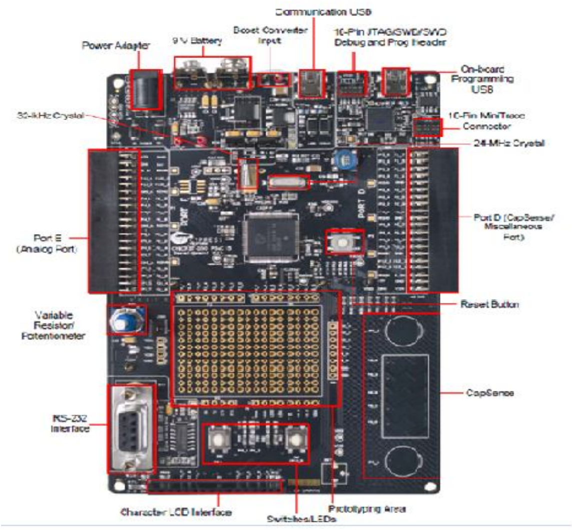

Fig.6 PSoC #5 Development Kit

V. IMPLEMENTATION RESULTS

Fig.7 shows IR sensor (TSOP) detects code at the receiver sent by remote (transmitter). It is indicated by glowing LED. As shown in fig.7, Arduino module is used.

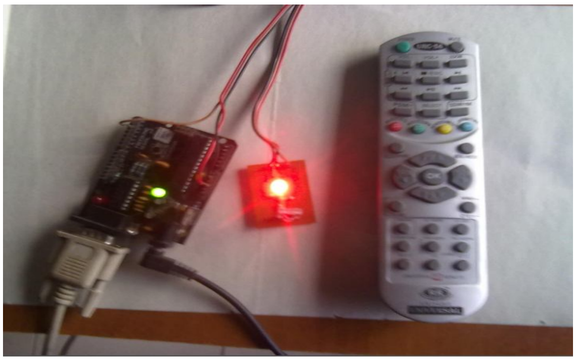

Fig.7 Detection of code using IR sensor

Fig.8 shows pulses at the receiver in digital form on HyperTerminal. These pulses can be seen on digital oscilloscope also. HyperTerminal is used for serial communication via RS232 (DB9) connector.

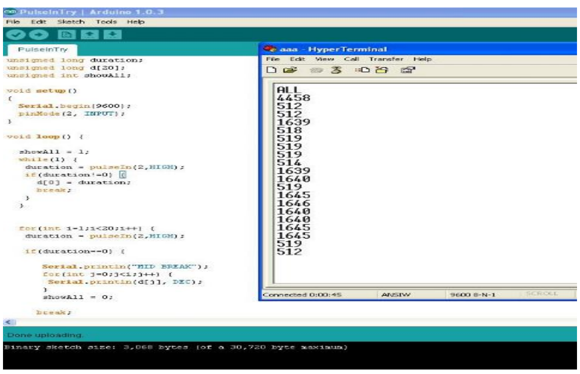

Fig.8 Pulses at receiver on HyperTerminal

Fig.9 shows Capsense system. Capsense is user friendly, smooth and has long life instead of normal buttons. As shown in fig.9 button 0 and button 1 is used to manage switch ON/OFF of device and slider showing the volume up/down in terms of percentage.

Fig.9 Capsense system showing button (for ON/OFF) And Slider (for channel up/down or volume up/down

CONCLUSION

This paper presented a implementation of a Universal Remote Control using PSoC #5. It also includes the result of individual blocks for IR sensor and Capsense. IR sensor decode the Philips and Samsung protocols separately. The advantage of URC is one can get relieve from the stress of handling too many remote control units. Mainly the concept leads to construct user friendly and intuitive interfaced URC. There is no more regular purchase of various sized-batteries for each and every remote. System-on-chip (SoC) solutions greatly simplify the process of designing a remote control and reduce system bill of materials (BOM) cost by eliminating the need for numerous discrete components.

REFERENCES

[1] C.S.Choy, "Infra-red Remote Control System Designed for Universal Control", IEEE Transactions on Consumer Electronics, Vol. 41, No. 4, November 1995.

[2] Taewan Kim,Hakjoon Lee and Yunmo Chung, "Advanced Universal Remote Controller For Home Automation and Security", IEEE 2010.

[3] J.Sathyan and A.R.Ramakrishnan, "A Unique Self Contained Universal Remote Control", IEEE 2004.

[4] www.sbprojects.com

[5] www.cypress.com