March 2018
Volume 125
Issue 1980
£5.90

www.electronicsworld.co.uk

# Electronics WORLD

THE ESSENTIAL ELECTRONICS ENGINEERING MAGAZINE


## Synchronous, low-EMI LED driver from Analog Devices

for automotive applications

## INSIDE THIS ISSUE

**LEDs and lighting:**
- automotive
- consumer
- industrial

**Trend**
- Science fiction that led to the smart home

**Regular column**
- Embedded vision, Industry 4.0 and IIoT

*Also inside:*
* TECHNOLOGY
* DIGITIZERS

# Your Home of Engineering

**Helping you take on the world's toughest challenges**

Be prepared for anything with our broad range of products

www.rs-online.com

# CONTENTS

## REGULARS


**04 >** **Trend**
Science fiction that predicted the smart home

**05 >** **Technology**

**42 >** **Products**



## COLUMNS


**08 >** **PCB cleaning**
By Mike Jones, MicroCare

**10 >** **Digitizers**
By Oliver Rovini and Greg Tate, Spectrum Instrumentation

**13 >** **MCUs**
By Lucio di Jacio, Microchip Technology

**18 >** **Embedded design**
By Dr Dogan Ibrahim, Near East University, Cyprus

**22 >** **Vision-based system design**
By Giles Peckham and Adam Taylor, Xilinx



## FEATURES


**24 >** **LED innovation drives next generation of intelligent vehicles**
Stefan Grötsch and Thomas Christl from Osram Opto Semiconductors, consider how lighting technology has enabled the development of intelligent vehicles

**27 >** **The time is now for lighting manufacturers to grow their business**
IoT and smart lighting control technologies can help lighting equipment OEMs improve their profits, TAM, CAGR and wallet share, says Erik Davidson from Cortet by CEL

**30 >** **Mix and math**
Swathi Sridhar and Namrata Dalvi from Microchip Technology describe a method for RGBA colour mixing using Bluetooth low-energy communications

**33 >** **Driving an LED beam with a single MCU port line**
By Huijie Li, Junting Li And Zhiqi Lin, Changchun University of Technology, China

**36 >** **Electronic phosphor offers many new capabilities**
By Marián Štofka, Slovak University of Technology, Bratislava, Slovakia

**39 >** **Introduction to LEDs and their lighting applications**
By Stojce Ilcev Dimov, Durban University of Technology (DUT), South Africa


Cover supplied by ANALOG DEVICES See p6-7

# SCIENCE FICTION THAT PREDICTED THE SMART HOME

The best science fiction has an uncanny knack for eerily foreshadowing future technology. Throughout history, outlandish concepts that seemed wildly fanciful at the time have become everyday occurrences today. For example, did you know that George Orwell thought of voice recognition as early as 1948? Or that the cult TV show Star Trek might just have kicked off the modern obsession with wearable technology?

## Connected Home

Fiction: In his book *'The Machine Stops'* (1904), EM Forster described a world where knowledge and information are shared by a system linking every home, and people communicate with each other via screens.

Reality: In the 1980s, Tim Berners-Lee's research at CERN resulted in the World Wide Web, making information accessible between devices on a network. By 2016 there were approximately 22.9 billion connected devices worldwide.

## 3D Printing

Fiction: Years before it happened, Neal Stephenson gave a detailed and scientifically accurate description of the exact method we use today for 3D printing in his book *'The Diamond Age'* (1995).

Reality: The first commercially-available 3D printer was launched in 2009; it created three-dimensional objects from layers of material, just like the "nano-assemblers" in Stephenson's book.

## Video Calling

Fiction: When mission commander Frank Bowman arrives at the space station in Stanley Kubrick's 1968 film *'2001: A Space Odyssey'*, he places a videophone call to his daughter from a payphone booth.

Reality: Inventions like Skype in 2003 and Facetime in 2010 have now brought Bowman's videophone to our pockets!

## Voice Recognition

Fiction: In George Orwell's novel *'1984'*, written in 1948, the characters used a device called a speakwrite, a Dictaphone that understands and records the spoken word.

Reality: In 1952, Bell Laboratories produced 'Audrey', which could understand spoken digits. Ten years later, IBM demonstrated 'Shoebox', which could handle 16 words. Today, with the likes of Siri and Alexa, we are witnessing the dawn of the voice-search age.

## Vacuuming And Automation

Fiction: The 1962 TV show *'The Jetsons'* told the story of a family in a futuristic utopia set in 2062, where household chores like vacuuming were completed by robots.

Reality: Today, vacuuming robots are increasingly popular and feature Wi-Fi connectivity, app controls and intelligent visual navigation.

Fiction: In his novel *'Demon Seed'* (1973), Dean Koontz imagined an entire home operated by a computer program, called Alfred. It controlled everything, from lighting to temperature, security, even cooking!

**Several technologies that are scientific realities today trace their origins back to influential works of fiction and the minds of sci-fi visionaries**

Reality: Today's connected devices help with chores, from slow cookers which allow remote-controlled cooking, to smart fridges and barcode readers that automatically order food.

## Flat Screen TVs

Fiction: Back in 1953, in his novel *'Fahrenheit 451'*, Ray Bradbury imagined that homes of the future would have "parlour walls" – flat TV screens as large as a living-room wall.

Reality: The first flat plasma display panel was invented in 1964 and flat screen technology has gone on to produce screens of enormous size.

## Wearable Technology

Fiction: Back in 1962, the Starfleet in TV show *'Star Trek: The Next Generation'* personnel wore communicator badges on their uniforms, allowing them to speak to each other remotely, as well as access the on-board computer.

Reality: In 2002, a California-based start-up created a connected pin badge. Wearable technology continues to advance with the latest Apple Watch allowing users to communicate via text and voice, keep track of their health, control their smart home, pay for goods and services and, of course, tell the time! ●

*By Chris King, Head of Home at comparethemarket.com*

**EDITOR:**
**Svetlana Josifovska**
*Tel: +44 (0)1732 883392*
*Email: svetlanaj@electronicsworld.co.uk*

**SALES:**
**Suzie Pipe**
*Tel: +44 (0)20 8306 0564 | Mobile: +44 (0)7799 063311*
*Email: suziep@electronicsworld.co.uk*

**DESIGN: Tania King**

**PUBLISHER: Wayne Darroch**

*ISSN: 1365-4675*

*PRINTER: Buxton Press Ltd*

SJP business media

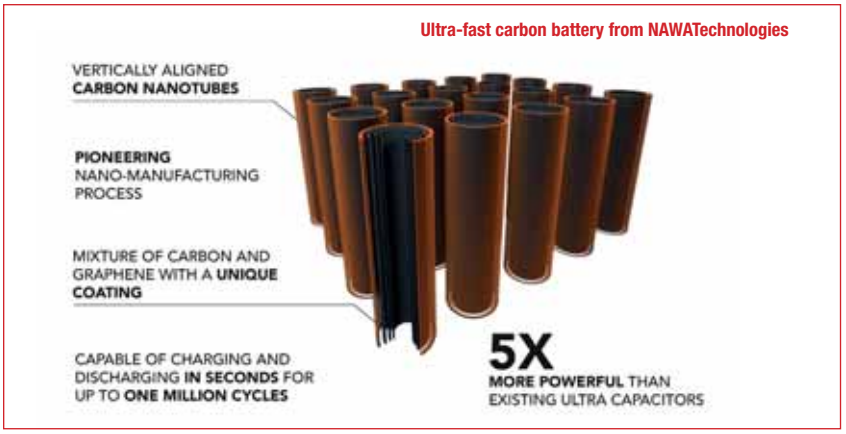
*2nd Floor,*
*52-54 Gracechurch Street,*
*London, EC3V 0EH*

**SUBSCRIPTIONS:**
*Subscription rates:*
*UK - 1 year digital only £53.00+VAT*
*UK - 1 year print and digital sub £68.00*
*UK - 2 year print and digital sub £109.00*
*UK - 3 year print and digital sub £143.00*
*International - 1 year digital only £53.00*
*International - 1 year print and digital sub £164.00*
*International - 2 year print and digital sub £290.00*
*International - 3 year print and digital sub £409.00*
*Tel/Fax +44 (0)1635 879361/868594*
*Email: electronicsworld@circdata.com*
*www.electronicsworld.co.uk/subscribe*

*Follow us on Twitter @electrowo*

*Join us on LinkedIn*

# NEW ULTRA-FAST CARBON BATTERY COULD REVOLUTIONISE ENERGY STORAGE



Ultra-fast carbon battery from NAWATechnologies

South-of-France-based NAWATechnologies has developed a carbon-type energy storage system on the principle of the ultra-capacitor, which can be charged and discharged in seconds for over a million cycles without loss in performance. This compares to conventional lithium-ion batteries that often take hours to fully charge and have a limit of about 5,000 cycles.

NAWATechnologies's new Ultra-Fast Carbon Battery is far more advanced than conventional lithium-ion batteries, in a market where the ultra-capacitor segment is expected to reach $2bn by 2025, with applications in hybrid cars, electric bicycles and power tools, among others.

NAWATechnologies has also pioneered a unique coating for the nanotubes used in its batteries, which help them have five times higher power density than existing ultra capacitors. The company has now entered low-volume pilot production.

NAWATechnologies chose carbon for its batteries because the material is abundant, accessible and natural, and is re-usable and recyclable. Being carbon-based, the Ultra-Fast Carbon Batteries do not suffer the same thermal issues as conventional storage systems.

There is a significant difference in the way the batteries store electricity, too. In a regular ultra-capacitor, there is a purely electrostatic reaction, while with a lithium-ion battery there is a purely chemical reaction. NAWATechnologies's combination of vertically-aligned carbon nanotubes, a unique coating and a chemical electrolyte, puts its batteries in a product space between regular ultra-capacitors and lithium-ion batteries, offering tremendous potential.

The company's so-called NAWASHELL is a highly innovative mechanical composite structure capable of storing energy. Applied to a smartphone case, for example, it could revolutionise this market in the way energy is stored. In automotive applications, these batteries could rapidly draw and store energy from regenerative braking systems. And in urban mobility, NAWATechnologies's batteries could be fitted into autonomous vehicles, including buses, trucks and other vehicles, to be recharged within seconds when at rest.

"What is really exciting is the sheer potential of ultra-capacitors. We have a ground-breaking battery that stores more electricity more rapidly – and it's safer, more reliable and kinder to the environment than current storage systems," said Pascal Boulanger, COO of NAWATechnologies.

The company was formed in April 2013 with staff with over 15-year experience in nanomaterials R&D.

# PORTABLE SUPERCONDUCTING SINGLE-PHOTON DETECTORS COULD FIGHT CANCER AND DRIVE CARS

Researchers from the University of Glasgow and the STFC Rutherford Appleton Laboratory have developed a new, compact cooling technology that could pave the way for advanced superconducting detectors in driverless cars, practical quantum communications and medicine.

Although the technology was initially developed for space astronomy, the new supercooled detector platform, capable of detecting single photons, is robust enough and has low enough power consumption to be used outside a laboratory environment.

The team's research builds on existing developments in the extremely sensitive light sensors known as superconducting nanowire single photon detectors (SNSPDs), which can detect individual photons even at infrared wavelengths. While SNSPDs have facilitated numerous significant advances in quantum science over the last decade, they need to be cooled to just a few degrees above absolute zero (-273.15°C) to work effectively – a process that requires expensive and hazardous liquid helium and a lot of power.

The newly-developed portable, less-power-hungry platform for SNSPDs opens a wide range of new applications. For example, in driverless cars it could do distance measurement, whereas in medicine it could fight cancer.

"In the cancer treatment called photodynamic therapy (PDT), the treatment drug exchanges energy with surrounding oxygen molecules on optical excitation, creating singlet oxygen radicals which kill tumour cells. A miniaturised cooling platform like ours would make SNSPD use in clinical PDT much more practical, potentially making cancer treatments more effective," said Professor Robert Hadfield from the University of Glasgow's School of Engineering.

The technology was initially developed for the European Space Agency's Planck mission, which launched in 2009 and successfully surveyed cosmic background radiation in the microwave and infrared spectra over four and a half years in space. The team then adapted it by using a fibre-optic-coupled superconducting detector supplied by the Dutch start-up Single Quantum, and housed it in a miniaturised cooler capable of reaching temperatures of 4.2 Kelvin, or -268.95°C, which runs from standard mains power.

# SYNCHRONOUS, LOW-EMI LED DRIVER FEATURES INTEGRATED SWITCHES AND INTERNAL PWM DIMMING

By **Keith Szolusha**, Applications Engineering Section Leader, and **Kyle Lawrence**, Associate Applications Engineer, Power Products Group, Analog Devices

The breadth of LED applications has grown to encompass everything from general lighting to automotive, industrial and test equipment, sign boards and safety equipment, expanding the design requirements, too. The latest LED solutions require compact, efficient, low noise drivers, with high dimming ratios and advanced fault protection. The LT3922 easily meets these demands.

The device is a synchronous LED driver with integrated 2A switches that can be configured as a boost, buck or boost-buck LED driver. Its high efficiency synchronous and integrated power switches fit into a tiny 4mm × 5mm QFN package. The integrated synchronous switches are run with controlled switching edges that do not ring, offering just the right balance of high efficiency and low noise, and can be run at up to 2.5MHz, resulting in compact solutions.

## CHOICE OF TOPOLOGIES

LED strings are driven by controlled current that does not need to be returned directly to ground. Both LED+ and LED–, or either one, can be attached to non-ground potentials, opening up options including floating output DC/DC LED driver topologies, buck mode (step-down) and boost-buck (step-up and step-down). The LT3922's high-side PWMTG driver and synchronous switches can be configured as a boost, buck mode or boost-buck LED driver, while retaining use of all of the ICs features, namely internal PWM dimming, SSFM, low EMI, ISMON output current monitor, and output fault protection carry over from the standard boost topology to the buck mode and boost-buck.

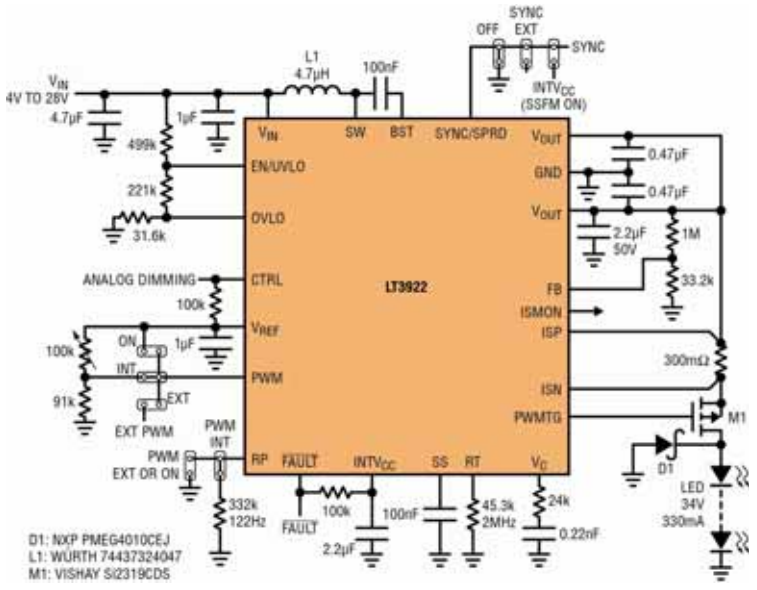



**Figure 1: 2MHz regular boost schematic with 2000:1 PWM dimming at 120Hz**

1. **Boost:** The LT3922 can power LEDs up to 34V when operated as a boost converter, leaving some headroom, below 40V, for open LED overshoot. The 2MHz, 4V to 28V boost LED driver shown in Figure 1 powers a 330mA string of LEDs at up to 34V. It can be externally PWM dimmed at 120Hz to a 2000:1 ratio or it can be internally dimmed to 128:1 ratio with an analog input voltage on the PWM pin.

   It survives open LED and LED+-to-ground short-circuits and reports these faults by asserting a FAULT pin. The output current can be monitored via the ISMON pin, even during PWM dimming. At 2MHz switching frequency, its fundamental EMI harmonic resides above the AM band, but its EMI is still low. Spread spectrum frequency modulation (SSFM) can be added to spread the switching frequency between 2MHz and 2.5MHz and reduce the EMI at the fundamental and its many harmonics. The efficiency of the 2MHz boost converter remains as high as 91% at $12V_{IN}$ due to the integrated synchronous switches. At lower $V_{IN}$, when the peak inductor current hits its limit, the output current reduces without flicker while the LEDs remain on.
2. **Buck:** The input voltage can be as high as 36V and a string of LEDs can be driven at up to 1.5A when the LT3922 is used in a buck-mode topology. The high-side ISP and ISN current sense input and PWMTG PMOS driver are easily moved to the high side of the LEDs, which is connected to the input in buck mode. LED– is connected directly to the inductor and not ground. When driving two 1A LEDs at 6.5V, the synchronous buck mode efficiency is as high as 94% at 12V $V_{IN}$ and remains as high as 89% at 36V $V_{IN}$. The high bandwidth of the buck mode converter allows it to work with a 1000:1 PWM dimming ratio at 100Hz.
3. **Boost-Buck:** The LT3922 boost-buck topology in Figure 2 supports an input voltage range extending above and below the LED string voltage. The sum of the LED string voltage and input voltage must remain below 35V to keep the ISP and ISN voltage below the 40V absolute maximum.

   This patented low-EMI topology features a boost-type low ripple input inductor and a buck mode-type low ripple output-facing inductor. A 4-18V automotive input or multiple battery chemistry input (5V, 12V and 19V) boost-buck converter can drive an LED string voltage anywhere between 3V and 16V.

## AUTOMOTIVE LIGHTING

LEDs are ideal for use in automotive lighting. Efficient LED headlights are robust, with lifetimes orders of magnitude longer than their relatively burnout-prone filament-based predecessors. Drivers are small and efficient with wide input and output voltage ranges, and should feature low EMI.

The tiny LT3922 LED driver features low EMI, high efficiency and fault protection required in automotive environments. It can be powered from the automotive 9-16V input range and operates to 36V in the face of transients and down to 3V (cold-crank conditions). Its low EMI Silent Switcher architecture, SSFM and controlled switching edges make it ideal for powering LEDs with low EMI. Its versatility makes it useful in boost, buck and boost-buck applications for exterior daytime running lights, signal lights, tail lights and headlight segments, as well as interior dashboard and heads-up displays with high dimming ratio. Built-in flexibility and fault protection reduce the number of required components to protect against short and open LED strings.

The 400kHz automotive boost LED driver in Figure 3 passes CISPR 25 Class 5 EMI tests, as shown in Figure 4, which shows conducted and radiated EMI test results of the LT3922 along with class 5 EMI limits. This is a result of a combination of LT3922 low-EMI features, including controlled switching edges and SSFM. Of course, proper layout and a small amount of ferrite bead filtering (FB1 and FB2) should be used for best EMI results.

## BUILT-IN FEATURES FOR LOW EMI

LT3922 incorporates Power by Linear's patented Silent Switcher architecture, where internal synchronous switches minimise hot-switching-loop size and controlled switching edges do not ring.

The switching edge rate is controlled by the LT3922, eliminating high frequency ringing found in switching converters without this feature. The LT3922's controlled switching edges reduce power switch high-frequency EMI without degrading efficiency and power capability.

SSFM in the LT3922 spreads the resistor-set switching frequency up and down from 100% to 125% of its value at a rate of 1.6kHz for the 400kHz converter. This decreases both the peak and average EMI in the converter at low and high frequencies. The feature is easy to turn on and off by connecting the SYNC/SPRD pin to INTVCC or GND, respectively.

## LED CURRENT ACCURACY

LED current accuracy is a necessity for vehicles with redundant light clusters. The brightness of both sides must match for obvious reasons. Identically manufactured LEDs can produce different brightnesses at the same drive current. The internal dimming feature of the LT3922 can be used for brightness trimming near or just below 100% duty cycle and then set to accurate 10:1 or 100:1 ratios. This can save the light cluster manufacturer from paying extra for specially binned LEDs. ●

**Analog Devices (UK) Ltd • Tel: 01628 477066**
**Email: uksales@linear.com • www.linear.com**

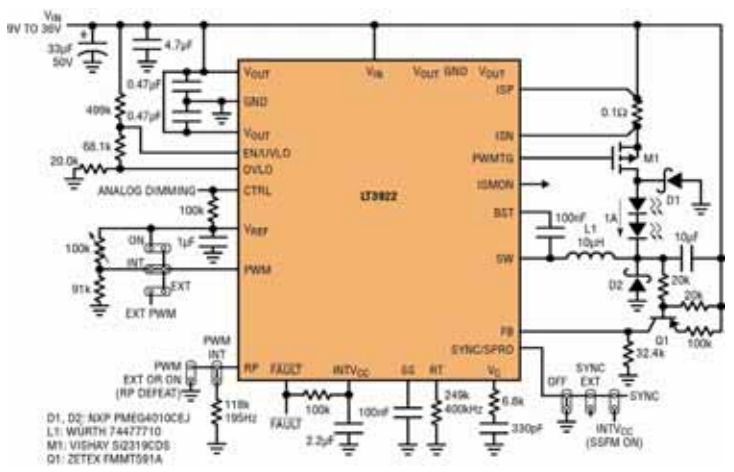



**Figure 2: 2MHz boost-buck LED driver with low input and output ripple, a solution that passes CISPR 25 Class 5.**

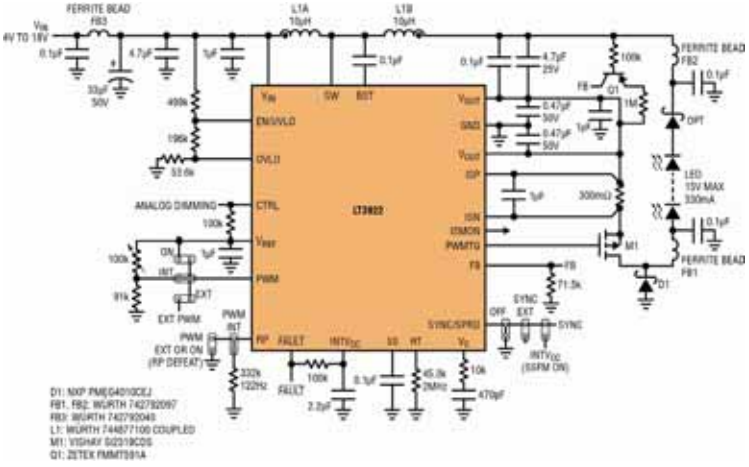



**Figure 3: 400kHz automotive boost LED driver with filters for low EMI and option for 100%, 10% or 1% internally-generated PWM dimming**

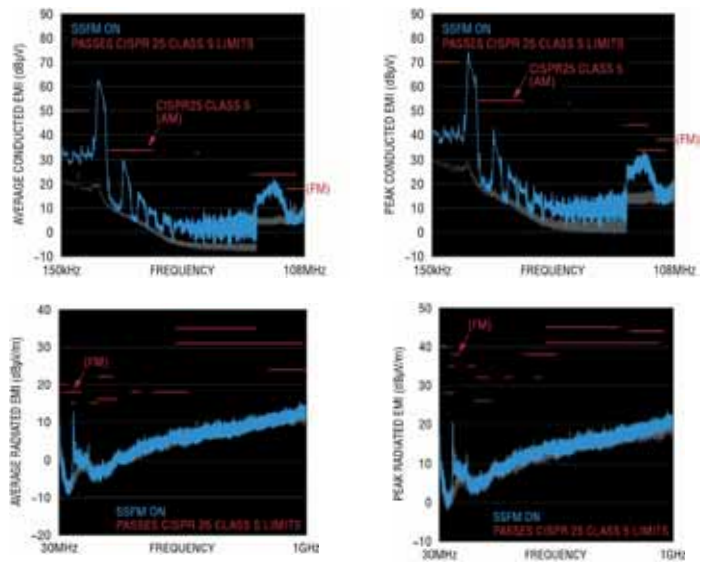



**Figure 4: EMI profile of the 400kHz LED driver shown in Figure 3, which passes CISPR 25 Class 5 with minimal EMI filters**

# The problem with the way we've always done it

BY **MIKE JONES**, VICE PRESIDENT, MICROCARE

For decades, dip-and-brush cleaning has been the benchmark for cleaning printed circuit boards (PCBs) during rework and repair. The tools are common, cheap and widely used, consisting of IPA alcohol, a pump bottle and a brush. Sadly, despite its convenience, dip-and-brush PCB cleaning can result in dirty boards, making them unreliable.

## Process Control and Reliablity

Process control remains a major deficiency in PCB cleaning, whether it be on a benchtop or in an automated cleaning system. Removing contamination quickly, consistently and affordably remains a huge reliability issue for PCB manufacturers.

The biggest weakness of dip-and-brush cleaning is that it offers little in the way of process control. Some technicians scrub better than others, some use too much fluid, while others barely wet the board. But, most importantly, dip-and-brush cleaning does not allow to rinse the contamination off the board. The PCB often remains sticky with fluxes even after cleaning, which leaves behind highly-corrosive halites, causing faults and white residue. It is important to remember: "If not properly rinsed, it's not clean".

Dip-and-brush cleaning might have been good enough some years ago, when PCBs were simple, with larger, more rugged components. Today, with new fluxes and pastes, and new PCB designs and delicate component packages, the method's efficiency and cost-effectiveness are questionable. In today's world of demanding electronics, a better process is needed to produce reliably clean PCBs.

## Successful Cleaning

Cheryl Tulkoff, a Texas-based reliability engineer for over twenty years, has taken a stand to eliminate the dip-and-brush process before it destroys more circuit boards.

"Contamination is believed to be one of the primary drivers of field issues in electronics today. It induces corrosion and electrochemical migration or ECM," says Tulkoff. "Contamination-related failures are frequently intermittent in nature, often manifesting themselves as no-fault-found (NFF) returns and, therefore, very difficult to diagnose."

In the paper *'To Kill A Circuit Board: Perils In Manual Soldering and Cleaning Processes'* presented by Tulkoff at an SMTAi Expo in Chicago, originally in 2014, she noted that "manual soldering and cleaning processes are among the least controlled processes in printed circuit board assembly, creating special challenges to both quality and long-term reliability". This still stands – even more so, with new PCB technologies, packages, pastes and fluxes.

**The best way to clean any board is an automated cleaning process, such as a vapour degreaser**

There are two important changes required for better cleaning at the workbench. Any cleaning process should be defined by (a) the specific quantity of cleaning material for each cleaning task, and (b) the quality of those materials must be managed. Cleaning with a dirty solvent and a worn-out brush is not going to make the grade.

The best way to clean any board is an automated cleaning process, such as a vapour degreaser, that will produce consistently-clean boards. However, automated cleaning is often impractical during rework and repair or when a manufacturer is operating at capacity. The benchtop technician needs a faster, cheaper and more convenient answer. The best option is to use a controlled dispensing system.


**Dip and brush cleaning is an uncontrolled cleaning process that leads to erratic results**

"If flux residues must be removed manually, a four-step process of wet, scrub, rinse and dry is recommended," says Tulkoff. "Use some form of dispensing system for the solvent to control the flow and volume."

### Faster and Better

The use of a controlled dispensing system that attaches to the aerosol can is key to control. This method delivers faster and better cleaning, with less waste than dip-and-brush. This type of tool creates a true benchtop cleaning process and can easily be documented and standardised, even to ISO standards, which will improve quality across the board.

In addition, it has been documented that technicians use less solvent when they use controlled dispensing technology, because the solvent is fresh and pure, and it is sprayed onto the exact location. There is no over-spray from traditional, high-pressure aerosols, and no dirty solvent wasted or thrown away.

It is also easy to quantify the amount of solvent for each cleaning job. This will allow a process to be written for benchtop cleaning to prevent overuse of a chemistry, as well as flux migration across the circuit board.

Another benefit of dispensing tools is that there is less waste, since the systems are designed to use the last drop of solvent within the aerosol can, making it very cost-effective. In addition, it is far easier to dispose of the cans because they are not partially filled with residual solvent.

The message is simple: to clean PCBs effectively, safely and reliably, employ the four-step process of 'wet, scrub, rinse and dry' using an appropriate solvent and a controlled dispensing system. Don't let old-fashioned dip-and-brush cleaning be the standard on your modern PCBs.


**Proper cleaning comes from using the four steps of cleaning: wet, scrub, rinse and dry**

**Vapour degreasing is a cost-effective way to clean PCBs using modern, planet-friendly solvents**

# Advantages of high resolution in high bandwidth Digitizers

BY **OLIVER ROVINI** AND **GREG TATE**,
SPECTRUM INSTRUMENTATION

Two key specifications of digitizers are bandwidth and amplitude resolution. Increasing resolution results in decreasing bandwidth, so users must make a tradeoff in selecting a digitizer to meet their measurement needs. Digitizers convert samples of an analogue signal into digital values using analogue-to-digital converters (ADCs). The resolution of an ADC is the number of bits it uses to digitize the input samples. For an n-bit ADC, the number of discrete digital levels it can produce is 2n, thus a 12-bit digitizer can resolve $2^{12}$ or 4096 levels.

The least significant bit (LSB) represents the smallest interval that can be detected, which in the case of a 12-bit digitizer is 1/4096 or $2.4 \times 10^{-4}$. To convert the LSB into voltage, take the input range of the digitizer and divide it by two, raised to the resolution of the digitizer.

Resolution determines the precision of a measurement. The greater the digitizer resolution, the more precise the measurement values. A digitizer with an 8-bit ADC divides the vertical range of the input amplifier into 256 discrete levels. With a vertical range of 1V, the 8-bit ADC cannot ideally resolve voltage differences smaller than 3.92mV; while a 16-bit ADC, with 65,536 discrete levels, can ideally resolve voltage differences as small as 15µV.

## Measuring Small Signals

One reason to use a high-resolution digitizer is to measure small signals. Based on the way the minimum voltage level was computed, a lower-resolution instrument and a smaller full-scale range could be used to measure smaller voltages. However, many signals contain both small and large signal components, requiring a high-resolution instrument with a large dynamic range that can measure small and large signals simultaneously.

Figure 1 compares 12-, 14- and 16-bit ideal digitizer responses to a segment of a ±200mV damped sine waveform. The segment selected is near the end of the waveform and has small amplitude. The 14- and 16-bit digitizers still have sufficient resolution to render the signal accurately but the 12-bit digitizer with 100µV resolution (based on a full-scale level of ±200mV) is unable to resolve levels smaller than 100µV. The error in reading – for any resolution – will increase with decreasing signal amplitude.

## Limitations in Achieving Maximum Resolution

There are several sources of error in digitizer systems, but they most fit into the categories of noise and distortion.

Distortion is an error in the acquired waveform that is highly correlated

with the signal being measured. Distortion is not random; it's dependent on the input signal. Most common is harmonic distortion, where distortion components appear in the frequency domain at integer multiples of the input frequency. Typical sources of harmonic distortion are non-linearity of the transfer function of the digitizer system including saturation, clipping, slew-rate limiting, and so on. Digitizer configurations that interleave multiple ADCs to achieve higher sampling rates add significant distortion on the sampling frequency due to mismatches in the gain and offset of each ADC – this is interleaving distortion.

### Noise

Noise, as opposed to distortion, is assumed to be uncorrelated with the input. Noise can be defined as any portion of the error signal whose frequency locations are not functions of the input frequency.

Noise itself is commonly broken into categories depending on its distribution (i.e. the shape of the histogram of the error) or the shape of the noise spectrum. When categorized based on frequency domain shape, noise that's evenly spread across all frequencies is called "white" noise. Noise with constant power per octave is called "pink" noise. There are many other noise shapes.

Noise is also categorized by the distribution of its histogram. Noise with a normal distribution is called Gaussian noise, of which there are many sources. Noise is also created through quantization, or the rounding error in converting analogue voltages to digital numbers. The simplest quantization methods produce a uniform error distribution that is white.

Noise arises in all electronic devices and designers make every effort to decrease internal noise. Gain stages are particularly prone to generating and increasing the level of noise in a digitizer.

Both distortion and noise limit the resolution that can be achieved by a digitizer. Noise, by adding a random component to each sample value, limits the digitizer's ability to resolve small amplitude values. This can be seen in Figure 2, which is the same damped sine signal waveform of Figure 1, but, here, the signal (with and without amplitude noise) is compared in the time domain.

Similarly, the frequency domain spectrum of the damped sine signal, with and without additive white noise, is shown in Figure 3. Note that the addition of the spectrally flat noise raises the baseline of the spectrum. Any signal with amplitude lower than the noise floor will be obscured, effectively limiting the digitizer's dynamic range.

### Harmonic Distortion

Normally, harmonic distortion occurs with much lower levels and is not visible on the time-domain waveform. Harmonics are generally evaluated in the frequency domain using a Fast Fourier Transform (FFT) spectrum analyzer, as shown in Figure 4. In the frequency domain, the third harmonic can be seen clearly. The presence of harmonic and other distortions can hide smaller spectral features that limit the dynamic range of the digitizer.

One measure of the spectral purity of a digitizer's output is the spurious free dynamic range (SFDR). SFDR is defined as the ratio of the RMS signal frequency component to the RMS value of the next larger spectral component (often referred to as a "spurious" or a "spur") at its output. The ideal spectrum in Figure 4 has a SFDR of about 81dB.

### Designer-Dependent

Minimizing the effects of distortion is primarily the job of the designer of the digitizer. Non-linearity, harmonic distortion and other sources of distortion must be reduced in the design. The user has little control over distortion other than not overdriving the digitizer, although there is some control to a degree:

1. Maximize the signal being analyzed in the input range of the digitizer to maximize the signal-to-noise ratio. Digitizers with multiple ranges make this easier, but make sure the noise level doesn't scale with input attenuation.
2. Use the minimum measurement bandwidth consistent with the application, since noise is proportional to bandwidth. This can be implemented using input bandwidth limiting or digital filtering.
3. Use signal processing such as averaging to reduce noise level

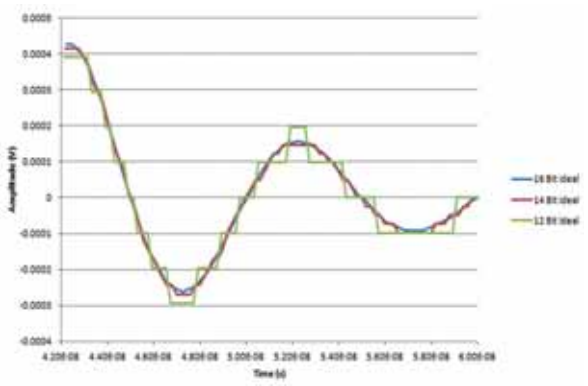


Figure 1: A comparison of digitizer resolution on measurement precision

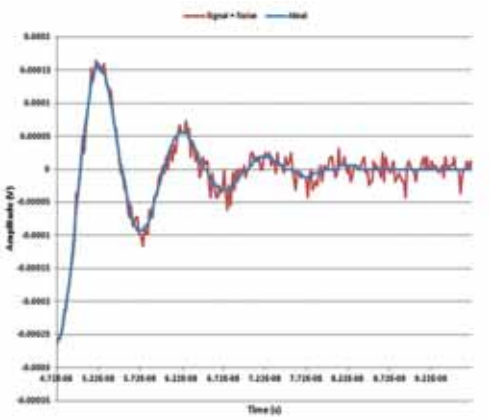


Figure 2: The damped sine waveform with and without amplitude noise

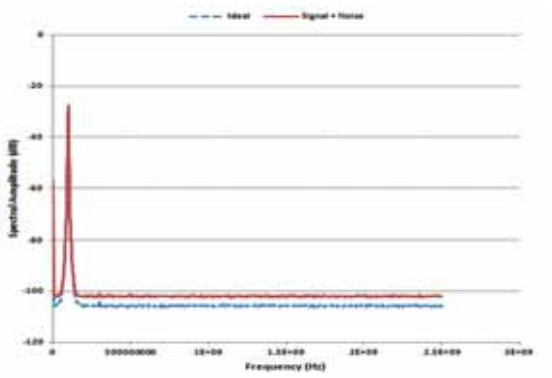


Figure 3: The frequency domain view of the damped sine with and without noise. Note that the addition of white Gaussian noise raises the baseline proportional to the noise amplitude

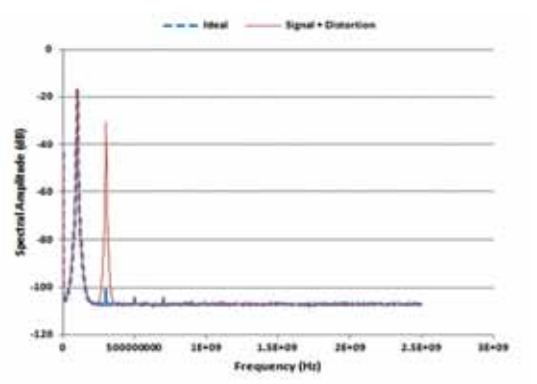


Figure 4: Comparison of the spectra of an undistorted waveform and one with harmonic distortion

proportional to the number of measurements averaged. Keep in mind this requires a repeatable signal and multiple acquisitions.

4. For low-level signals, use external low-noise amplifiers to boost signal levels and maximize signal-to-noise ratio.
5. Use the proper termination in the complete signal path. For high bandwidth, a desirable choice is a consistent 50Ω termination for signal source, cabling and digitizer termination.

'Figures of merit' are measurements for which there are common definitions such that numerical values instantly convey the quality of the measurement system being described.

## Applications Requiring Digitizers

Applications that require digitizers with a large dynamic range and hence greater resolution are those where the signals encountered include both high and low amplitude components:

- Echo-ranging measurements including radar, sonar, LIDAR, ultrasound and medical imaging. In these applications a large transmitted pulse is followed by a much weaker return echo, and the digitizer must accurately handle both signals.
- Ripple measurements of signals with high offset values and small variations riding on top of the offset. Both components must be characterized.
- Modulation analysis; amplitude modulated signals (AM, SSB, QAM, etc) exhibit wide variations in signal amplitude.
- Mass spectrometry; particles with significantly different mass/charge ratio need to be detected or the sensitivity of the mass spectrometer improved.
- Phase measurements require measurement of very small differences in amplitude to distinguish small phase differences.
- Propagation studies measuring signal path attenuation over various paths and through different media often result in a wide range of amplitude values.
- Component testing, where large voltage or current drops need to be characterized. ●

**FREE BOOKS**

**This article is adapted from *"The Digitizer Handbook – Precision and Performance in PC Instrumentation"* by Spectrum Instrumentation.**

**To receive a free copy, go to https://spectrum-instrumentation.com/en/contact-us and tick the box marked "Please send me a copy of the Digitizer Handbook" , adding "EW" in the Comment section.**

# Down Memory Lane

BY **LUCIO DI JASIO**, MICROCHIP TECHNOLOGY

Recently I have been testing a new Mikromedia (TFT) display board. This brought back memories from several years ago, when I was involved in the PIC32 launch and the very first PIC32 Mikromedia board, whose name at the time was "Multimedia".

My first personal/home computer – back when the dinosaurs still roamed the Earth – was a Sinclair ZX Spectrum. It was based on an 8-bit Zilog Z80 microprocessor running close to 4MHz, for an equivalent performance of less than 1MIPS. It used a small programmable logic device (PLD) to assist the CPU and generate a composite video output that could be fed to a normal TV receiver acting as a colour monitor. A rubber keyboard completed the "workstation", with a cassette tape recorder acting as a "mass storage" device.

Even the very first PIC32MX models, running at 80MHz, would execute instructions at approximately 100x the speed of the good old Z80; so back in 2008, I thought it would be fun to develop an emulator of the ZX-Spectrum on a Mikromedia board.

I wrote the emulator in C code, and the result was so effective I ended up playing some of the original video games on the Mikromedia. A few examples are still available today as videos on YouTube, such as "Jet Set Willy" and "Space Invaders".

In the years since the Mikromedia family has grown to a large selection of screen sizes and microcontrollers, including the PIC18F67K40, the first PIC18 family to be "upgraded" with a full set of Core Independent Peripherals.

The PIC18 is still an 8-bit microcontroller, but it runs some 16x faster than the original Z80. It also integrates enough peripherals to emulate a large part of the Spectrum hardware. With the video output taken care of by the Chip-on-Glass controller of the Mikromedia, I started wondering, could a PIC18 Mikromedia emulate my old ZX-Spectrum?

## PIC18 Graphic Libraries

I've had plenty of experience working with the Microchip Library for Applications (MLA) over the years; I even wrote a book about it. What I'd lost track of was how much the MLA had grown from the 8-bit controllers, evolving over the years whilst developers where focusing on newer, 16- and 32-bit devices. None of the demos would compile out of the box anymore with the new XC8 compiler, but it didn't take me long to port them back, either. After all, the MLA code is well structured and, aside from some work-around resource limitations (the PIC18'K40 RAM tops at 4K), I spent most of the time speed-optimising the display (parallel) interface. Font management was also too convoluted for my needs, so I ended up chopping away most of it.

## PIC18 Z80 Emulation

Next, I was ready to dive into the Z80 emulator itself.

The Z80 has an instruction set that expands on the original Intel 8080 ISA, almost tripling it; by definition, it is a CISC. There are almost 700 unique instructions, although an emulator can reproduce most of them by working on groups that share similar logic but operate on different registers.

The first implementation for the PIC32 was approximately 1,200 lines long, mostly composed on nested switch() statements. Recompiling for the PIC18 turned out to be very straightforward. In fact, the only real obstacle turned out to be the recursive technique I'd used to simplify the decoding of two large groups of instructions with IX and IY pointers. Those Z80 instructions are obtained by applying a prefix (0xDD or 0xFD) to any instruction that uses the HL register pair as a pointer.

Normally, the XC8 compiler won't allow recursive calls, since by default it applies an optimisation technique known as Compiled Stack that is particularly efficient for the PIC architecture. But in the case of the PIC18, an alternate instruction set (more efficient when implementing a more traditional C stack) can be enabled via a simple project option, setting the Stack type to "Re-entrant mode"; see Figure 2.

## Benchmarking

Skipping a few minor issues that related to the different sizes of the integer types on the 8-bit compiler (where int defaults to a 16-bit

word), I was ready to run some Z80 code and start benchmarking.

I hand-coded a few instructions and initialised the ROM array with them (see Listing 1), then ran the emulator inside the MPLAB simulator (this is recursive!) and watched the (emulated) Z80 registers animate.

Figure 1: Mikromedia PIC18FK

```
/*
 * test rom
 */

const byte ROM[] = {
0x3E, 0xFF, // LD A,#FF
 0x47, // LD B,A
 0x48, // LD C,B
 0x51, // LD D,C
 0x5A, // LD E,D
 Z80_RST0
 };
```

Listing 1: First Z80 test code

The Z80 was alive, but how fast was it running? Activating the simulator stopwatch function provided some clues.

As you can see from Figure 3, when I placed a breakpoint at the beginning of the Z80 instruction decoding loop, it quickly became obvious that the PIC18 (@64MHz) would emulate each instruction at approximately a tenth of the speed of the original microprocessor. In other words, my dream of running the ZX Spectrum on the Mikromedia PIC18FK had zero probability of becoming reality any time soon.

Eventually, it was comforting to notice that the emulator did not use that much memory. As can be seen from Figure 4, only 9% of the (128kbyte) program memory of the PIC18 was being used, so there was plenty of room for the emulation of the ZX Spectrum ROM (16kbytes) and other code necessary to provide the emulation of the keyboard, and perhaps an SD card to replace the original cassette tape.

## In Search of Performance

I could easily imagine myself spending more time to complete the ZX Spectrum emulation, regardless of the reduced speed. After all, that would get me a nice basic interpreter to play with, and perhaps I could even get some of the original (BASIC) games going. Instead, I got engrossed in a new quest for performance.

As it turned out, the original emulator code, written for the PIC32, had never required any optimisation. You see, at the time it just worked, so I spent all my efforts ensuring correctness, rather than speed. Now that was a challenge I wanted to take on. I started with an analysis of the XC8 compiler efficacy, especially when implementing some of the C language constructs that I had used the most in the emulator. As mentioned before, the actual Z80 instruction decoding was implemented by nesting two (and occasionally three) levels of switch statements. The specific order of the case(s), for example, was a known factor from my previous experiences with compilers.

You can get a glimpse of the nested nature of the code by looking at a snippet of the main decoding function in Listing 2.

```
ir.opcode = fetch();
switch( ir.x)
{
//-------------------------------------------------
case 0:
switch( ir.z)
{
case 0: // z=0 relative jumps and assorted ops
switch( ir.y)
{
case 0: // NOP
DIS("NOP");
break;
case 1: // EX AF AF'
DIS("EX AF,AF'");
t = reg[A]; reg[A]=reg2[A]; reg2[A]=t;
tf=flags; flags=flags2; flags2=tf;
break;
case 2: // DJNZ
DIS("DJNZ");
ofs = fetch();
r = --reg[B];
if (r > 0) // dec B
 pc += ofs; // jump if NZ
GETS(r);
GETZ(r);
GETH(r);
flags.n=1;
break;
```

Listing 2: A snippet of the main Z80 instruction decoding mechanism

Unfortunately, it did not take me long to discover that the gains to be had by rearranging the "cases" were minuscule.

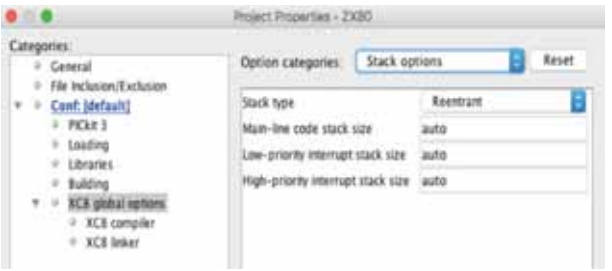



Figure 2: Changing the stack type to Reentrant

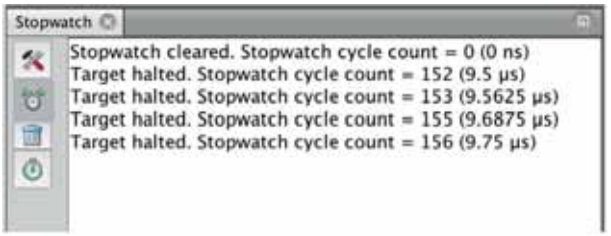



**Figure 3: Benchmarking the Z80 Emulator with the MPLAB simulator stopwatch**

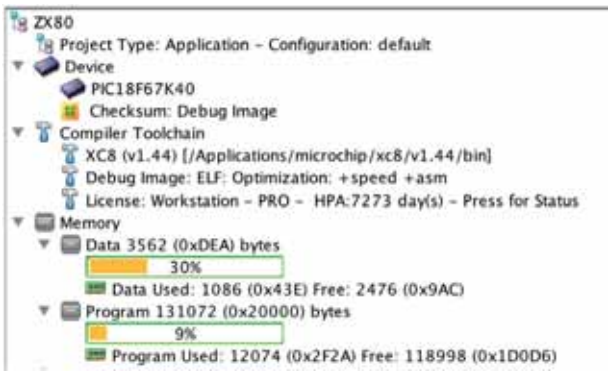



**Figure 4: Z80 Emulator Code Size and RAM usage**

The XC8 compiler (in PRO mode) does already a pretty good job optimising that type of code. This is especially true when the number of options is regular, a power of two and complete, as is almost always the case when decoding instructions.

Even from a quick look at the snippet in Listing 2 alone, it is also obvious that there is a significant amount of code required for the emulation of the Z80 ALU flags – notice the use of the macros called GETS(), GETC() etc.

The Z80 arithmetic logic unit (ALU) has six such flags, and mapping them on top of the PIC18 own flags requires a significant amount of work.

## Re-Compiler

Eventually I stopped fiddling, and for a moment got to thinking about the deeper nature of the problem. I was trying to emulate an 8-bit (CISC) processor with another 8-bit (RISC) processor, running at barely 16x the speed. So, it was obvious that any single-byte Z80 instruction had to be covered in a maximum of 16 PIC instructions. Longer instructions (prefixed, including literals, or including offsets and 16-bit addresses) would double and occasionally triple that budget, but I could not deviate too much from it. Yet, getting a single opcode from memory was already costing the PIC18 almost half of that budget even when the compiler was in-lining the fetch() macro:

```
#define fetch() ROM[pc++]
```

It was clear now that the only option was to work in assembly, which, in fact, it would require avoiding runtime decoding of the Z80 instructions altogether. What I needed was a (static) recompiler – a tool capable of performing the following steps:

1. Decoding of the emulated processor (binary source) opcodes;
2. Identifying branches and loops, eventually re-generating "labels" for all branch targets;
3. Generating assembly code for the target processor, replacing each emulated instruction with an appropriate inline (short) sequence or a call to a convenient function with the desired effect.

## Python to the Rescue

My usual go-to language when I need a quick-and-dirty proof of concept, a few-lines-long script, is Python. My first attempt at implementing (proof of concept) the decoding step, for example, turned out to be a couple of dozen lines long:

```
class Code(object):
 add = 0 # the current opcode address

 def __init__(self, opcode:int)->None:
 self.opcode = opcode
 self.decode()

 def decode_NA(self:type)->None:
 pass

 def decode_alu(self:type)->None:
 self.operand = self.z
 self.dest = 7
 self.mnemo, self.change_flags, self.use_flags = {
 0:('ADDA', {S,H,C,V,N,Z}, {} ),
 1:('ADCA', {S,H,C,V,N,Z}, {C}),
 2:('SUBA', {S,H,C,V,N,Z}, {} ),
 3:('SBCA', {S,H,C,V,N,Z}, {C}),
 4:('ANDA', {S,H,C,V,N,Z}, {} ),
 5:('XORA', {S,H,C,V,N,Z}, {} ),
 6:('ORA', {S,H,C,V,N,Z}, {} ),
 7:('CPA', {S,H,C,V,N,Z}, {} ),
 }[self.y]
 def decode(self:type)->None:
 self.x, self.y, self.z = split2reg(self.opcode)
 self.size = 1
 self.add = Code.add
 {
 0: self.decode_NA,
 1: self.decode_NA,
 2: self.decode_alu,
 3: self.decode_NA,
 4: self.decode_NA,
 5: self.decode_NA,
```

```
6: self.decode_NA,
7: self.decode_NA,
} [self.x]()
Code.add += self.size

# step 1. Decoding
codeArray = [Code(x) for x in rom]
```

**Listing 3: Snippet of the Python script (partially) implementing the decoding step**

Note how the snippet in Listing 3 represents nothing more than a translation in Python of the first "switch" layer in the equivalent C implementation in Listing 2. The second step requires fixing the actual target address of all jumps and calls once their actual value is known after the first pass.

```
# step 2. Generating a dictionary of labels (targets)
for index, code in enumerate(codeArray):
 if code.add in labels:
 labels[code.add] = index
```

**Listing 4: Labelling step**

The third and last step is like the first, except this time we select from the identified/decoded instructions and replace them with in-line PIC code or pre-templated calls to functions.

```
def generate_adda(self:type)->None:
print(f'\tmovf \t{reg_dict[self.operand]},W\n'
f'\taddwf \tA,F' )

def generate(self:type)->None:
{
'ADDA': self.generate_adda,
# . . .
} [self.mnemo]()
```

**Listing 5: Generation step**

Listing 5 shows only an example for the 'ADD A, reg' type of Z80 instruction that translates to this PIC assembly pair:

```
movf reg,W
addwf regA,F
```

## Optimising Flags Generation

Things get hard when handling the ALU flags. In the example here I have all but ignored the effects of the 'ADD A, reg' instruction on all six flags of the Z80 ALU. If we had to reproduce their exact behaviour when re-compiling each arithmetic instruction, the resulting code would be quite large. Yet, most frequently the effects on the flags produced by each individual instruction are ignored or immediately overridden by the following instructions during program execution.

One possibility is to enhance the recompiler to recognise which flags are needed and which can be safely ignored.

```
def track_flag(flag:int, line:int, mark:set)->bool:
 if line >= len(codeArray): # reached the end
 return False
 if line in mark: # looping
 return False
 code = codeArray[line]
 mark.add(line)
 if flag in code.use_flags: # code uses flag
 return True
 if flag in code.change_flags: # code changes flag
 return False
 else:
 if track_flag(flag, line+1, mark): # check if its follower
uses
 return True
 elsif code.branch:
 return track_flag(flag, labels[code.branch], mark)
 else:
 return False

# step 2bis analyze Z flag usage
for line in range(len(codeArray)):
 if codeArray[line].change_flags:
 if track_flag(Z, line+1, set()):
 print("Must generate Z!")
```

**Listing 6: Tracking flags actual usage**

Adding an intermediate step (before the re-generation) by calling the track_flag() function for each opcode that does modify flags allows us to determine at compile time where and when flag emulation needs to be implemented or skipped altogether to save time and space.

Notice that the track_flag() function as I implemented it in Listing 6 is recursive and, although smart enough to identify loops (see use of the mark set), can be quite computationally-intensive.

## In the End

I took quite a tangent this time, starting off with some graphics library porting and playing with a new TFT display board and, before realising it, I was down Memory Lane, building an emulator and then a recompiler tool in Python.

On the positive side, I have a way to run (albeit slowly) some legacy Z80 code if I need to, and could even run some Spectrum basic on a PIC18 Mikromedia.

The recompiler tool was also a perfect excuse to practice some Python 3 coding (did you notice I used the new type hinting feature and even some f' strings formatting?). I learn something new every day! ●

# Pulse width modulation techniques in embedded power control

By **Dr Dogan Ibrahim**, Professor at Near East University, Cyprus

Pulse width modulation (PWM) is a commonly used technique for controlling the power delivered to analogue loads using digital waveforms. Although analogue voltages and currents can be used to control the delivered power, they have several drawbacks.

Controlling large analogue loads requires large voltages and currents that can't easily be obtained with standard analogue circuits and DACs. Precision analogue circuits can be heavy, large and expensive, and they are also sensitive to noise. By using PWM, the average value of voltage and current fed to a load is controlled by switching the supply voltage on and off at a fast rate. The longer the power-on time, the higher the voltage supplied to the load.

## PWM Wavewforms

Figure 1 shows a typical PWM waveform where the signal is basically a repetitive positive pulse with period T, on-time $T_{ON}$ and off-time of $T_{OFF} = T - T_{ON}$ seconds. The minimum and maximum values of the voltage supplied to the load are 0 and $V_p$, respectively.

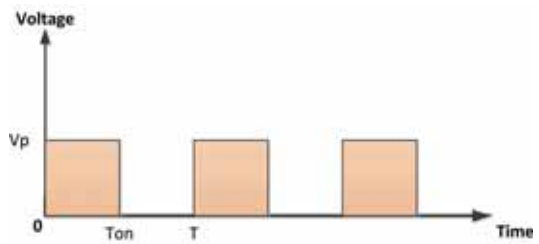



**Figure 1: A typical PWM waveform**

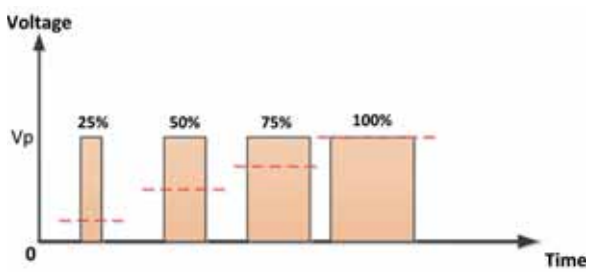



**Figure 2: Average voltage (dashed lines) supplied to a load**

The PWM switching frequency is usually set very high (several kHz) so it does not affect the load. PWM allows for efficient operation of the load, since the power loss in the switching device is very low. When the switch is on, there is practically no voltage drop across it, and when off, no current is supplied to the load.

On the other hand, resistive-method power delivery incurs high losses.

The duty cycle (D) of a PWM waveform is defined as the ratio of on-time to its period:

$$\text{Duty Cycle (D)} = T_{ON}/T$$

The duty cycle is usually expressed as a percentage:

$$D = (T_{ON}/T_{OFF}) \times 100\ \%$$

By varying the duty cycle between 0% and 100% we can effectively control the average voltage supplied to the load between 0 and $V_p$, calculated by considering a general PWM waveform as shown in Figure 1. The average value A of waveform f(t) with period T and peak value $y_{max}$ and minimum value $y_{min}$ is:

$$A = \frac{1}{T}\int_0^T f(t)\,dt$$

or,

$$A = \frac{1}{T}\left(\int_0^{T_{ON}} y_{max}\,dt + \int_{T_{ON}}^{T} y_{min}\,dt\right)$$

In a PWM waveform $y_{min} = 0$ renders the above equation as:

$$A = \frac{1}{T}(T_{ON}\, y_{max})$$

or,

$$A = D\, y_{max}$$

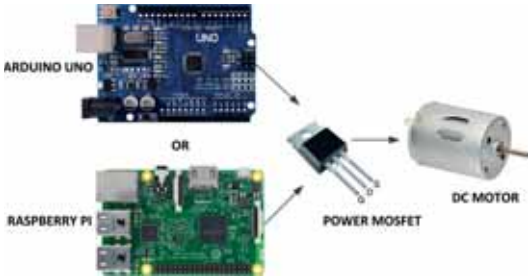



**Figure 3: Application setup**

The equation shows that the average value of voltage supplied to the load is directly proportional to the duty cycle of the PWM waveform and, by varying the duty cycle, the average load voltage can be controlled.

Figure 2 shows the average voltage for different duty-cycle values.

The RMS value of a waveform is its effective value, and can be calculated as follows.

The RMS value of a repetitive waveform is given by:

$$V_{RMS} = \sqrt{\frac{1}{T}\int_0^T v(t)^2 dt}$$

where $v(t) = V_P$ for $0 \le t < T_{ON}$

Therefore: $$V_{RMS}^2 = \frac{1}{T}\int_0^{T_{ON}} V_P^2 dt$$

giving: $$V_{RMS}^2 = \frac{V_P^2}{T} T_{ON}$$

or, $$V_{RMS} = V_P\sqrt{\frac{T_{ON}}{T}}$$

or simply, $$V_{RMS} = V_P\sqrt{D}$$

The equation shows that the RMS value of the voltage supplied to the load is proportional to the square root of the PWM waveform's duty cycle.

Most inductive loads, such as electric motors, actuators and others, respond to the average value of the PWM waveform. In some applications it may be necessary to smooth out the waveform to minimise any ripples, to look more like a DC waveform. This is usually done by increasing the PWM frequency with a low-pass filter at the output of the waveform. The filter's cutoff frequency must be much lower than the PWM frequency.

## PWM with Microcontrollers

Generating a PWM waveform with a microcontroller is a simple task, which can be done using standard delay functions as described in the following code:

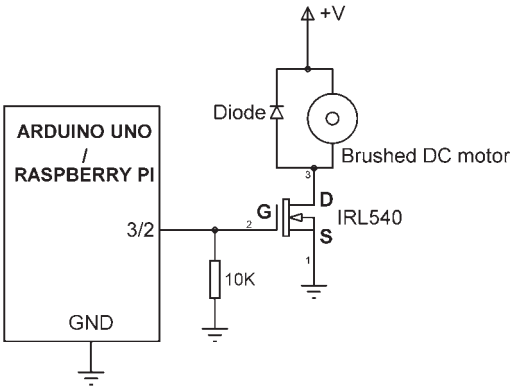



**Figure 4: Circuit diagram**

**Set PWM pin to logic 1**
**ON delay**
**Set PWM pin to logic 0**
**OFF delay**

The PWM frequency and duty cycle depend on the on and off delays. Assuming they are in milliseconds, the PWM frequency in kHz and the duty cycle are approximated by:

**Let T = ON delay + OFF delay**

Then,

**PWM frequency = 1/T**
**Duty cycle = ON delay/T**

There are two main drawbacks in using delay functions to generate PWM waveforms. First, the generated waveform is not very accurate because the delay functions are not normally accurate. In addition, the time taken to set and clear the PWM pins are not taken into account in the above simple code. Second and, more importantly, the above code ties up the CPU, preventing it from processing any functions other than generating the PWM waveform.

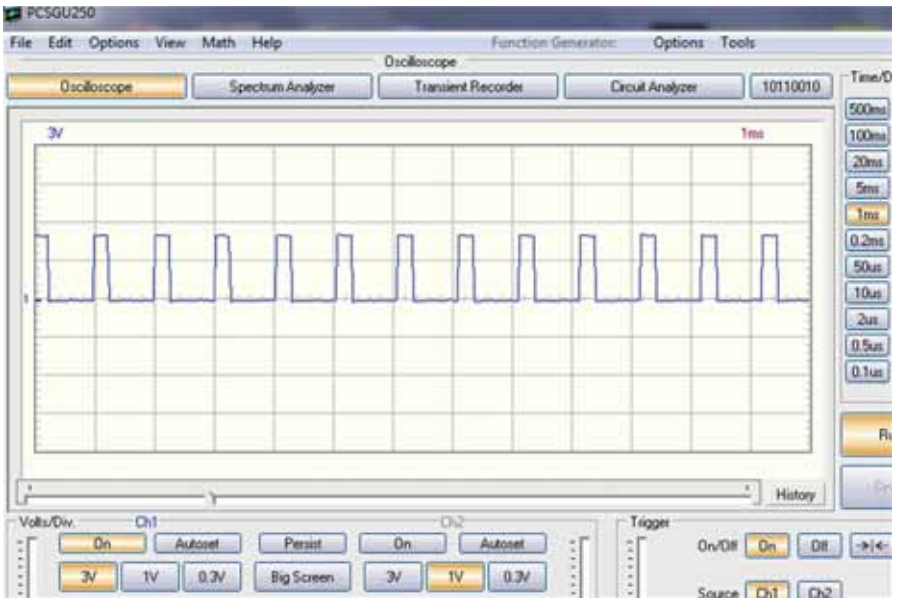
Fortunately, most modern microcontrollers include an on-chip multi-channel PWM module that, once configured, runs independent of the CPU.

## Example Applications

Here we discuss using PWM to control the speed of a small brushed DC motor. Two programs are presented based on the current most-popular microcontrollers, Raspberry Pi and Arduino Uno. Any model Raspberry Pi can be used in this application; the PWM duty cycle and waveform frequency are selected to be 25% and 1kHz respectively for demonstration purposes.

Figure 3 shows the setup. The processor generates PWM waveforms and drives a power MOSFET transistor as a

**Figure 5: Generated PWM waveform**




switch, which in turn supplies voltage to the motor. The motor speed changes as the duty cycle of the waveform varies.

MOSFET power transistors can handle large currents and are commonly used as switches in power control applications.

Figure 4 shows the application circuit diagram. We chose an IRL540 power MOSFET transistor, because its gate is logic-level-compatible and it can handle currents to 40A.

The gate of the MOSFET is connected to pin 3 of the Arduino Uno, and to pin 2 of the Raspberry Pi. Power to the motor is supplied from an external source since the Arduino Uno or the Raspberry Pi GPIO ports can only supply very limited power.

### Raspberry Pi Software

The Raspberry Pi program is written in Python; see Listing 1. At the beginning of the program, the GPIO module RPi. GPIO is imported and Variable Motor assigned to pin 2 of the Raspberry Pi GPIO port – a pin configured as output. The PWM waveform is configured using the 'GPIO. PWM' statement, where the motor pin number and PWM frequency are specified. The PWM waveform is then started with a 25% duty cycle.

The Python PWM library supports the following functions:

**Listing 1: Raspberry Pi program listing**

```
#---------------------------------------------------------------------------------
#                        PWM MOTOR SPEED CONTROL
#                        =======================
#
#  This is a very simple motor speed control. Port 2 of the raspberry Pi is
#  connected to a small brushed DC motor through a power MOSFET switch.
#  A PWM waveform is generated with frequency and duty cycle of 1kHz
#  and 25% respectively. The motor rotates at quarter of its full speed.
#
# File    : motor.py
# Date    : January 2018
# Author: Dogan Ibrahim
#---------------------------------------------------------------------------------
import RPi.GPIO as GPIO

GPIO.setwarnings(False)
GPIO.setmode(GPIO.BCM)
Motor = 2
GPIO.setup(Motor, GPIO.OUT)

#
# Generate PWM waveform at motor pin with 25% duty cycle and frequency
# of 1000Hz
#
p = GPIO.PWM(Motor, 1000)
p.start(25)
```

Figure 5 shows the PWM waveform sent to the motor when the program is live. The waveform was captured on a PC using the Velleman PCSGU250 oscilloscope; the horizontal axis is 1ms/division and the vertical axis 3V/division.

### Arduino Software

Arduino Uno supports six PWM channels, connected to three timers, with registers TCCR0B, TCCR1B and TCCR2B:

| | |
|---|---|
| **Pin 5 and 6** | **Connected to Timer0** |
| **Pin 9 and 10** | **Connected to Timer1** |
| **Pin 11 and 3** | **Connected to Timer2** |

The PWM frequency depends on these timer settings, the clock frequency and the pre-scaler value. The pre-scaler is 3-bit wide, stored in the least three bits of the three timer registers CS02, CS01 and CS00. The pairs of PWM pins have the same PWM frequencies; for example, PWM on pins 11 and 3 are controlled by timer TCCR2B and have the same frequencies.

The default PWM frequencies are:

| | |
|---|---|
| **Pin 5 and 6** | **976.56Hz** |
| **Pin 9 and 10** | **400.20Hz** |
| **Pin 11 and 3** | **490.20Hz** |

The PWM frequency can be changed by re-loading the three least significant bits of the timer registers. For example, the timer values and corresponding PWM frequencies for PWM pins 11 and 3 are:

**TCCR2B = TCCR2B & B11111000 | B00000001 frequency = 31372.55Hz**
**TCCR2B = TCCR2B & B11111000 | B00000010 frequency = 3921.16Hz**
**TCCR2B = TCCR2B & B11111000 | B00000011 frequency = 980.39Hz**
**TCCR2B = TCCR2B & B11111000 | B00000100 frequency = 490.20Hz (DEFAULT)**
**TCCR2B = TCCR2B & B11111000 | B00000101 frequency = 245.10Hz**
**TCCR2B = TCCR2B & B11111000 | B00000110 frequency = 122.55Hz**
**TCCR2B = TCCR2B & B11111000 | B00000111 frequency = 30.64Hz**

In this program, the PWM frequency is set to 980.39Hz for simplicity, by using the following statement in the program setup routine:

**TCCR2B = TCCR2B & B11111000 | B00000011**

The duty cycle of the generated PWM signal is controlled by the statement 'analogWrite', which accepts the PWM pin number and duty cycle as its parameters. The duty cycle ranges from 0 to 255, where 0 corresponds to 0% and 255 corresponds to 100%. In this example the required duty cycle is 25%, corresponding to 64.

Listing 2 shows the Arduino Uno program. At its beginning, the motor is defined as pin 3, also configured as output. The PWM frequency is then set to approximately 1kHz.

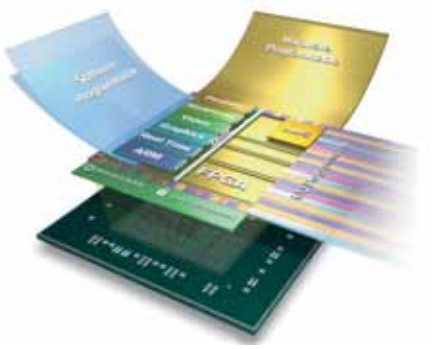
Inside the main program, the PWM duty cycle is set to 25% by loading 64 to the analogWrite function. ●

**Listing 2: Arduino Uno program listing**

```
/*************************************************************
*                PWM MOTOR SPEED CONTROL
*                ========================
*
*  This is a very simple motor speed control. Pin 3 of the Arduino is
*  connected to a small brushed DC motor through a power MOSFET
*  switch. A PWM waveform is generated with a duty cycle of 25% and
*  frequency of approximately 1kHz. The motor spins at a quarter of its
*  full speed.
*
* File    : Motor
* Date    : January 2018
* Author: Dogan Ibrahim
*************************************************************/
#define Motor 3

//
// Configure pin 3 as output and set PWM frequency
//
void setup()
{
   pinMode(Motor, OUTPUT);
   TCCR2B = TCCR2B & B11111000 | B00000011;
}

//
```



# Embedded vision in Industry 4.0 and the IIoT

BY **GILES PECKHAM** AND **ADAM TAYLOR**, XILINX

A key enabling technology of Industry 4.0 is embedded vision. Industry 4.0 introduces automation, data collection and sharing to the manufacturing flow, where embedded vision systems are instrumental in the positioning and guidance of parts and their identification. Embedded vision is not limited to the elements visible to the naked eye; it is applicable to x-rays and the infra-red part of the spectrum, adding further useful information to the manufacturing processes.

Many embedded vision systems use machine learning inference at the edge, which allows accurate detection and identification of not only parts' quality but their correct assembly and positioning on the production line. Such approaches enable better yields as non-conformance is identified earlier in the manufacturing flow.

## Industry 4.0 Applications

Designing a solution for the large volumes typically found in Industry 4.0 requires close consideration of size, weight, power and cost, often called SWaP-C, as well as processing capacity and connectivity.

Within Industry 4.0 applications, embedded vision systems are typically applied at the network edge and connect to other edge devices and the Cloud via the Industrial Internet of Things (IIoT), but also to sensors, cameras and other modules, via both industrial standard interfaces and proprietary/bespoke ones.

For Industry 4.0 applications, the IIoT converges both the information technology (IT) and operational technology (OT) networks. A vertical integration across the layers of the automation pyramid allows scalability and enlargement, as well as end-to-end communication. However, many current industry applications use separate networks, requiring gateways and bridges to connect them.

Separation exists because OT networks need to support real-time deterministic communication and demonstrate a low packet-delay variation, while IT networks are optimised for high bandwidth, flexible topologies and automated configuration. Unfortunately, separate networks are difficult to scale, require multiple protocols, need network engineering before installation and often have bandwidth restrictions in OT segments.

Converging IT and OT networks addresses these issues, providing a network that is no longer very hierarchical and limited in scalability or performance, along with achieving the desired vertical integration.

## Time-Sensitive Networking

One converged network solution used in Industry 4.0 applications is Time-Sensitive Networking (TSN), a set of IEEE 802 sub-standards that enable deterministic communication over Ethernet networks. Embedded vision systems supporting Industry 4.0 and IIoT applications should support TSN for ease of integration.

TSN enables different classes of network traffic to share the same link, providing both network management and a reserved path for scheduled traffic, ensuring deterministic communication. Ultimately, TSN enables the implementation of one common network that supports

| Standard / IEEE Draft | Title | User's Advantage |
|---|---|---|
| IEEE 802.1AS (evolving to P802.1ASrev) | Network Time Synchronization | All nodes share the same time |
| IEEE 802.1Qbv | Scheduled Traffic | Scheduled Ethernet frames never collide |
| IEEE 802.1Qci | Filtering & Policing | Removes babblers from the network (Security) |
| P802.1CB | Seamless Redundancy | Zero Loss switch-over |
| P802.1Qcc | Stream Reservation | Path provisioning according to IEEE |
| IEEE 802.1Qbu and IEEE 802.3br | Frame Pre-emption | Maximum bandwidth without compromising real-time behavior |

**Table 1: IEEE TSN standards**

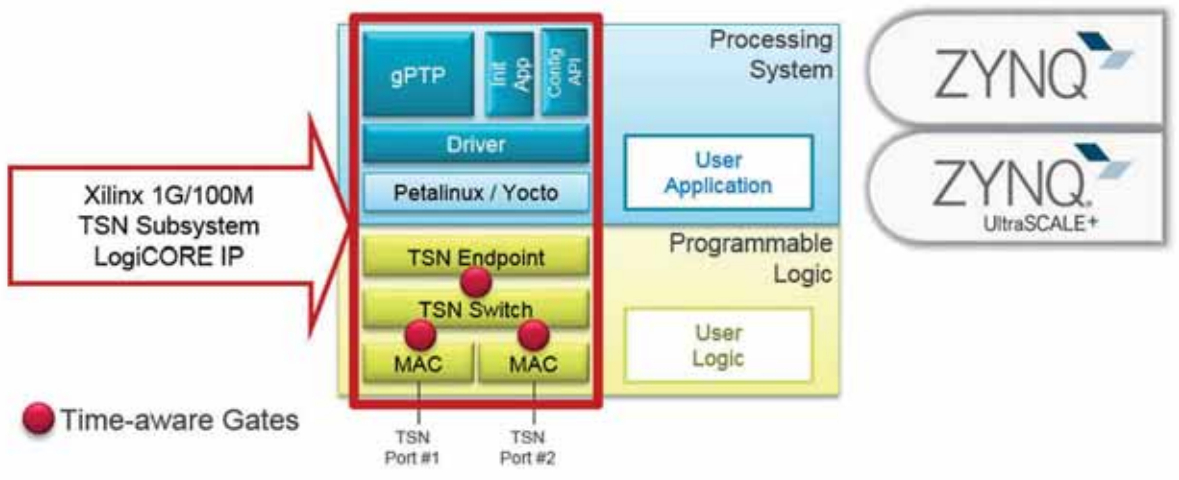



**Figure 1: TSN implementation within a Zynq SoC or Zynq UltraScale+ MPSoC**

multiple communication standards; see Table 1.

When TSN operates over an Ethernet link, there are several improvements to standard Ethernet. Standard Ethernet communications are not time-aware; they distribute data over the entire bandwidth of the link, with packets queued for transmission. TSN implements a time awareness across the network, with scheduled traffic in time-defined slots, and supports cyclic data transmission while pre-empting higher priority packets. TSN also better manages the very high frame-rates in optical monitoring processes due to its use of cyclic behaviour with high throughput.

A correct implementation of TSN must provide low latency and deterministic response at the TSN end-points and switches. Accurate timing for the transmission of scheduled Ethernet frames requires dedicated circuitry such as the programmable logic found in FPGAs. The combination of a processor system and programmable logic also enables the implementation of TSN within a single device, helping to achieve SWaP-C, which is critical for Industry 4.0 embedded vision applications.

### Example Solution

An example of TSN implementation is to use a Zynq-7000 or Zynq UltraScale+ MPSoC device, utilising both the processing system (PS) and the programmable logic (PL); see Figure 1. Xilinx's TSN Endpoint Ethernet MAC LogiCORE IP supports Ethernet with 100Mbit/s (Fast Ethernet) and 1000Mbit/s (Gigabit Ethernet). It comprises programmable logic for MAC, TSN Bridge and TSN Endpoint, along with software components for network synchronisation, initialisation and interfacing with network configuration controllers for stream reservation. The software is designed to run on PetaLinux and will be published as Yocto patches.

The LogiCORE IP, built with dedicated resources from the device's PL, provides deterministic behaviour for synchronisation (IEEE 802.1AS), scheduled traffic (IEEE 802.1Qbv) and seamless redundancy (P802.1CB), while also offloading from the processing unit.

The IP core also comes with an optional integrated time-aware L2 switch that creates the chain or tree topology required by many industrial applications, without allocating another port of an external TSN switch. Seamless redundancy (P802.1CB) is then feasible as it requires the additional port. The user can freely configure before synthesis to integrate the switch or not.

**Unfortunately, separate networks are difficult to scale, require multiple protocols, need network engineering before installation and often have bandwidth restrictions in OT segments**

The TSN IP core provides individual interfaces for each traffic class, used in conjunction with the processors, DDR memory and interconnect within the PS. These AXI Stream interfaces support scheduled, reserved and best-effort/legacy-traffic over the network.

The AXI infrastructure is ideal for high-bandwidth communication to the processing system. Direct Memory Access (DMA) for each traffic class manages the data stream. AXI Lite is used to perform initialisation and configuration of the TSN subsystem.

Not all TSN sub-standards are fully adopted today, but the reconfigurability of programmable logic provides the benefit of enabling updates to the TSN subsystem LogiCORE IP core as the TSN standards are finalised. ●

# LED innovation drives next generation of intelligent vehicles

**Stefan Grötsch**, Principal Key Expert, Applications, and **Thomas Christl**, Marketing Manager, both from Osram Opto Semiconductors, consider how lighting technology has enabled the development of intelligent vehicles

The developments of LED technology and optoelectronics have played a significant role in the automotive industry. LEDs and optoelectronic devices are widely used in vehicles, including displays, interior and exterior lighting, but also as infrared light sources and sensors for safety and driver-assistance systems.

## Significant Developments

We are increasingly seeing intelligent and autonomously-operated lighting systems create a safer and more comfortable driving environment.

Statistics have shown that insufficient lighting is responsible for many accidents, so countries like Sweden, Italy and Hungary have introduced corresponding laws, and many others require drivers keep their lights permanently on when driving. New vehicle-lighting standards often mandate that lights switch on automatically, to avoid driver error or lapse. Also, autonomy has been introduced, since many motorists rarely use their high-beam lights, worried they'll dazzle other road users.

One of the most significant developments of recent years in automotive lighting technology is the development of high-beam lights directed away from oncoming traffic yet still provide illumination of the road. Another significant development is adaptive headlights, where headlamps adjust to the curve profile of the road, allowing for a precisely controlled light distribution without dazzling other road users.

The implementation of multichip LEDs is one of the contributing factors in the evolution of these intelligent lighting systems, which continues thanks to advances in multi-pixel LEDs.

## Adaptive Headlamps

Multichip LEDs help set new standards for adaptive functions in headlamp systems (adaptive front-lighting systems – AFS) and glare-free high-beam headlights.

The first iterations of AFS used traditional lighting technology that required mechanical systems to adjust headlight aim.

However, with LED technology, auto manufacturers can implement these intelligent systems more easily than before and without mechanical systems for pivoting.

In the latter, intelligent controllers turn on/off the chips needed in a specific driving situation. The controllers use several sensors to ensure the right chips are operating and that light is distributed properly to maximise the safety of both driver and other road users. The sensors can automatically and flexibly turn off regions of the headlamp's beam according to the driving situation, and dynamically adjust to account for the position of other users on the road. In addition, dynamic cornering and turning or marker lights can illuminate potential obstructions on the side of the road.

## Modern LEDs

The latest generations of automotive LED modules are even smaller, allowing greater flexibility when developing front-lighting systems. LED modules are also cheaper and easier to install, reducing time to market and cost of fitment, bringing them to a wider selection of vehicle models, not just the premium class.

To date, adaptive LED headlamps have operated with individually-controlled chips for each illuminated area, but the trend is toward LED chips with micro-structured pixels that can be controlled individually.

Under the micro AFS project, a group of German companies worked for three years on a new class of energy-efficient AFS LED headlamps. Osram Opto Semiconductors was the project coordinator and, together with partners Daimler, Hella, Infineon and the Fraunhofer institutes IZM and IAF, announced the first milestone of 256 controllable pixels. This was followed by a pixel-light source with 1024 individually-controllable light points, providing about 3 lumens at only 11mA for an individual pixel surface of 0.115mm x 0.115mm from a closed emission surface of 4mm x 4mm with a grid size of 0.125mm. Here, the multi-pixel LED chip was mounted directly onto the smart Si-LED driver chip. A new conversion technology enabled high contrast ratios and efficient white-light generation. The individually-controllable light points help more clearly define the light images to be produced.

**The trend is toward LED chips with micro-structured pixels that can be controlled individually**

The key here is the integration of microelectronics and optoelectronics. Experts from Osram Opto Semiconductors developed the innovative LED pixel chip that emits flexibly-controllable light patterns in blue. The challenge was to define light points during chip processing and enable them to be connected directly to the control. Three of the 1024-pixel LEDs have already been used in headlamps of a Daimler test vehicle.

## Autonomy Through Light

The last year has seen unprecedented development in autonomous driving technologies. Traditional car manufacturers are competing with Silicon Valley innovators in a race to roll out the first truly self-driving car. Recently, Apple announced its plans to develop autonomous vehicle technology and, earlier this year, Ford predicted that it would produce its first car with Level-4 autonomy in 2021.

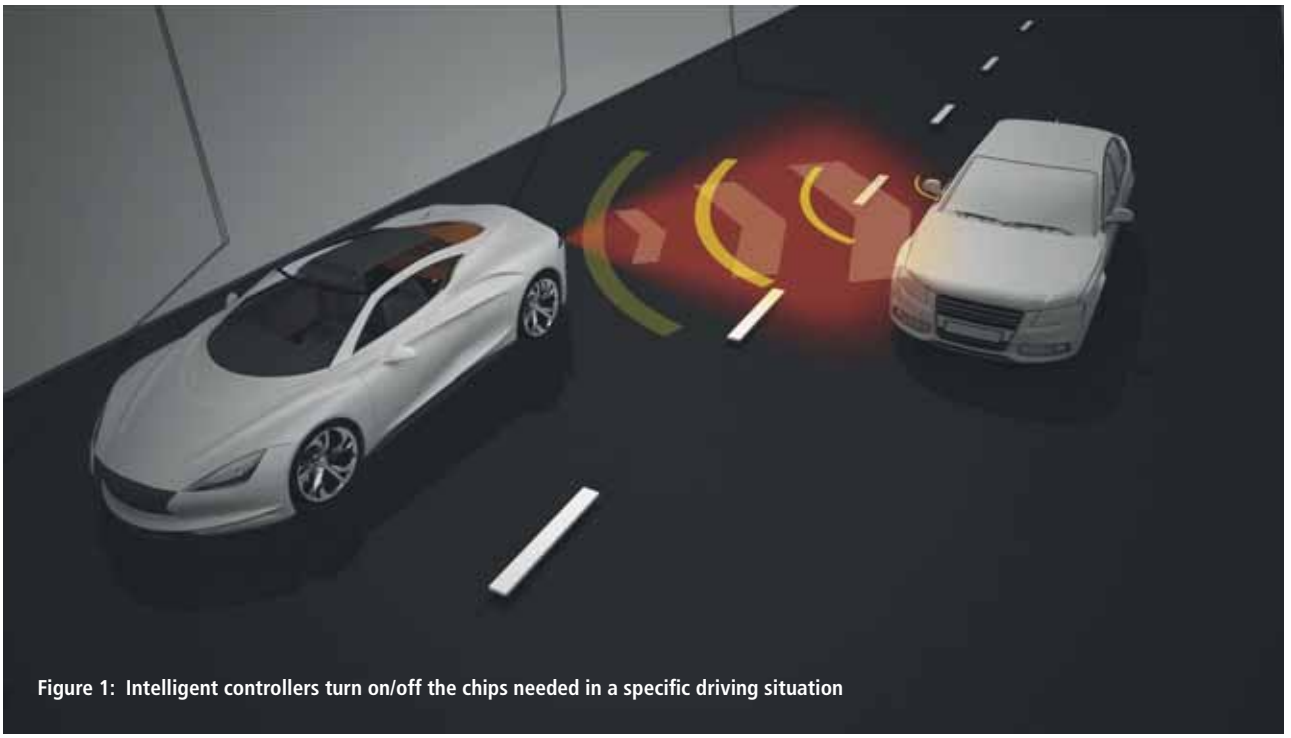


**Figure 1: Intelligent controllers turn on/off the chips needed in a specific driving situation**

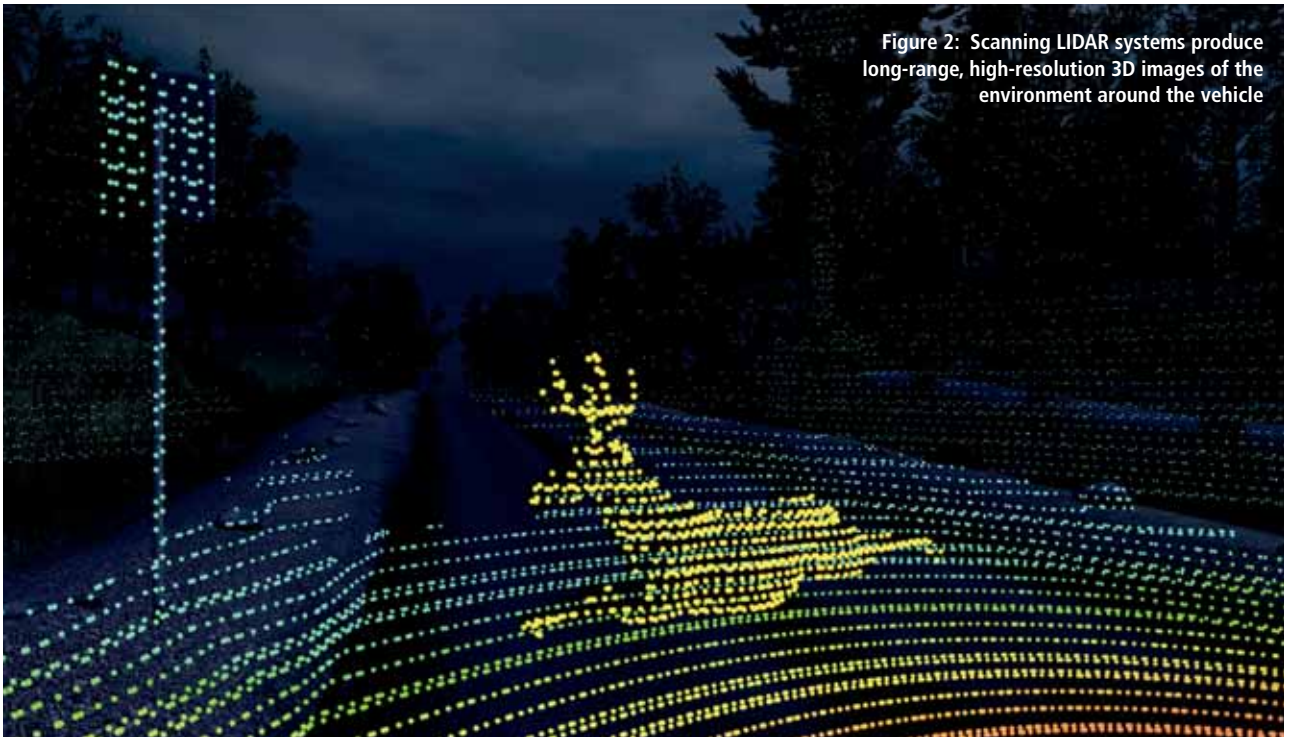


Figure 2: Scanning LIDAR systems produce long-range, high-resolution 3D images of the environment around the vehicle

Major industry players, including Uber, Tesla and Google, have begun testing their own self-driving cars, as regulators and OEMs begin to examine how autonomous vehicles, human drivers and other users can share the road.

However, despite the excitement at the potential that autonomous vehicles offer, areas of concern remain around the reliability of the underpinning technology and the impact of widespread adoption on society.

Sensors are a key enabling technology in self-driving vehicles, since they help manage the car to avoid obstacles and navigate safely through various road conditions. LIDAR (Light Detection and Ranging) systems and automotive-qualified high-power pulse lasers are commonly implemented by autonomous vehicle manufacturers since they offer some clear benefits over other sensing technologies, including radar and cameras.

Scanning LIDAR systems – the form that will likely dominate the autonomous vehicle sector – use a pulsed laser beam that scans across the field of view in small angular increments. A fast sensor or sensor array then detects the reflected beam to produce long-range, high-resolution 3D images of the environment around the vehicle.

The latest generation of LIDAR technology features 5ns pulse lasers, a laser bar with four individually-addressable laser diodes and a driver ASIC integrated into the module. The entire module is surface-mountable, which reduces assembly costs and eliminates the need to adjust individual light sources.

Together with its partner Innoluce, Osram Opto Semiconductors has developed a reference design of a MEMS-based scanning-LIDAR using a 4-channel laser. This system has shown very good performance with a detection range of more than 200m for a car and 70m for a pedestrian. In addition, its angular resolution is below 0.5 degrees, with a scan rate of 2kHz.

## Future Needs

While scanning LIDAR systems are on the way to providing the performance and reliability necessary for entirely driverless cars, LIDAR is already being implemented in some semi-autonomous applications and will likely be the backbone of the sensing technology in autonomous vehicles. To complement scanning LIDAR, a range of sensor technologies will be combined in Level-5 autonomous vehicles to ensure absolute safety and redundancy, including radar sensors, cameras and flash LIDAR.

**LIDAR will likely be the backbone of the sensing technology in autonomous vehicles**

Despite the progress made in the development of sensor technology, some concerns remain with autonomous vehicles. Data taken from driverless cars shows that some sensors struggle to recognise cyclists – and it can be difficult for autonomous vehicle technology to predict their behaviour. Further innovation in sensor and laser technology is needed so that autonomous vehicles can operate safely on the road.

But one thing is certain: LEDs and optoelectronics are set to play a key role in the development of the car of the future, since they deliver improved levels of safety for road users and greater autonomy within vehicles. ●

IOT AND SMART LIGHTING CONTROL TECHNOLOGIES CAN HELP LIGHTING EQUIPMENT MANUFACTURERS IMPROVE THEIR PROFITS, SAYS **ERIK DAVIDSON**, DIRECTOR, MARKETING & PRODUCTS, CORTET BY CEL

# The time is now for lighting manufacturers to grow their business

Right now, there is a uniquely strong opportunity for lighting luminaire and lamp OEMs to enter the smart building control market with their own IoT-based lighting solutions.

For all those luminaire/lamp OEMs hesitant to plunge into smart lighting control (SLC), or have perhaps already been unsuccessful at it, here are several reasons why now is the right time.

Latest data shows that this market is set to grow by over 35% to $3.7bn by 2020, with a growing number of startups forming to take advantage of it. However, lighting luminaire or lamp OEMs have two unique advantages over newcomers: unmatched knowledge of the devices being controlled, and deep, long-standing, industry relationships.

## Old-School Advantages

An established OEM understands better than anyone what it takes to design, manufacture, install, troubleshoot, maintain and upgrade lights, and, perhaps most importantly, this understanding is gained from a widespread, hard-won customer base.

These advantages help OEMs define the right strategy, giving them something most difficult to attain: in-depth understanding of the customer and how to deliver a superior experience to

them. This in turn leads to another major advantage over newcomers: established OEMs have already built and optimised a successful and far-reaching sales and distribution network for specific market segments. Thus, OEMs have an enormous head-start over newly-formed firms. For one, lighting agents are regional with regional exclusivities, and it takes years to establish national and international reach. Dealing directly with distributors and contractors can be just as complex and fragmented; startups and new entrants don't have the robust channels OEMs have.

## The Right Time

So, why is now the right time?

There are three primary reasons:

1. Lighting control system features and capabilities have become more sophisticated, and their technology is more accessible. As often happens, disruptive technologies – including early IoT systems – tend to be overly complex, difficult to integrate, not supported well and in limited supply. But, with time, as the technologies mature, these circumstances tend to improve.

   In the IoT sector, increased engineering rigour is being applied right now to many IoT systems.
2. The growing number of new entrants and approaches in lighting control has vastly increased the number of available partners, creating new ecosystems. Just a few years ago this domain was fragmented, with a very small number of serious participants, few partnering opportunities and even fewer mature technology choices.

   That has changed; there are varied and plentiful IoT hardware, software and service companies across an increasingly crowded landscape, giving non-IoT OEMs an easier, more profitable and less risky way to enter new markets with IoT-enabled products.
3. Most roadblocks and pitfalls to enter this market are now going away, due to this expanding partner ecosystem. With a growing number of experienced, capable and trustworthy partners to choose from, the days of "going it alone" are over.

## Support and Guidance

Today more companies offer OEMs support and guidance in the critical areas that history has shown to be the most difficult for IoT newcomers – and notorious for torpedoing even the best go-to-market strategies – including after-sale support, technology integration, ongoing system upgrades and maintenance, adjusting to rapidly-evolving standards, developing long-term strategic product roadmaps, and more.

The real question is, how can a strategically-designed SLC portfolio expand overall sales, increase profit margins and begin building recurring, subscription-based revenue.

For those who produce and sell luminaires and/or lamps, there's an immediate opportunity to add a SLC system to their portfolio to increase top line sales, primarily generated by two factors:

1. Revenues from selling SLC systems; and
2. Added revenues from the SLC-linked luminaire and lamps, and even shares from SLC projects.

But, is the opportunity big enough and how fast is the SLC opportunity growing relative to luminaire sales?

Data shows that by 2022 the TAM for SLC systems will exceed $19bn. Data from Strategies Unlimited suggests that the LED luminaire market has been growing at an 11% CAGR to reach TAM of $45bn by 2022. This means that CAGR for SLC system sales will likely be more than double the already-impressive growth rate of LED luminaire sales over at least the next several years. To beat the market as a manufacturer of LED luminaires, the present target should be an ambitious 15-20% CAGR, and with SLC a more aggressive 30-35%.

## Expanding the Wallet Share

There another important financial opportunity to note, and that's the ability to expand the "wallet share" with end customers. The phrase refers to the percentage a company receives of the end-customer's total project budget.

Let's say an office renovator has a lighting retrofit budget for a small office of $200,000 and it includes plans for adding SLC. Originally, perhaps you'd be getting around 25% of that budget ($50,000), primarily for providing luminaires. By adding an SLC solution pre-integrated with luminaires, a greater share of the total contract budget could be taken away from competitors, by both selling more luminaries and the SLC system too.

**Data shows that by 2022 the TAM for SLC systems will exceed $19bn**

Our latest research shows that, on average, your wallet's total share could be increased by 30-40% by adding an SLC solution. This means that instead of making a total of $50,000 on the example contract above, some $70,000, or 35%, could be gained of the contract. Over 100 similar small projects, sales would increase by $2m, which is not bad at all.

And, there's more good news: In addition to larger TAM, CAGR and 'wallet share' potential, adding an SLC solution significantly increases profit margins. First, when building a strategy for entering the SLC market, try to IoT-enable the most successful and highest profit-contributing lamps and luminaires in the portfolio. Offering these products initially integrated with an SLC platform increases the chances of success for the total offering, since the original devices are already well received, in high demand and generating profit.

Our research shows that by combining "winning" devices with a white-labelled SLC platform (such as

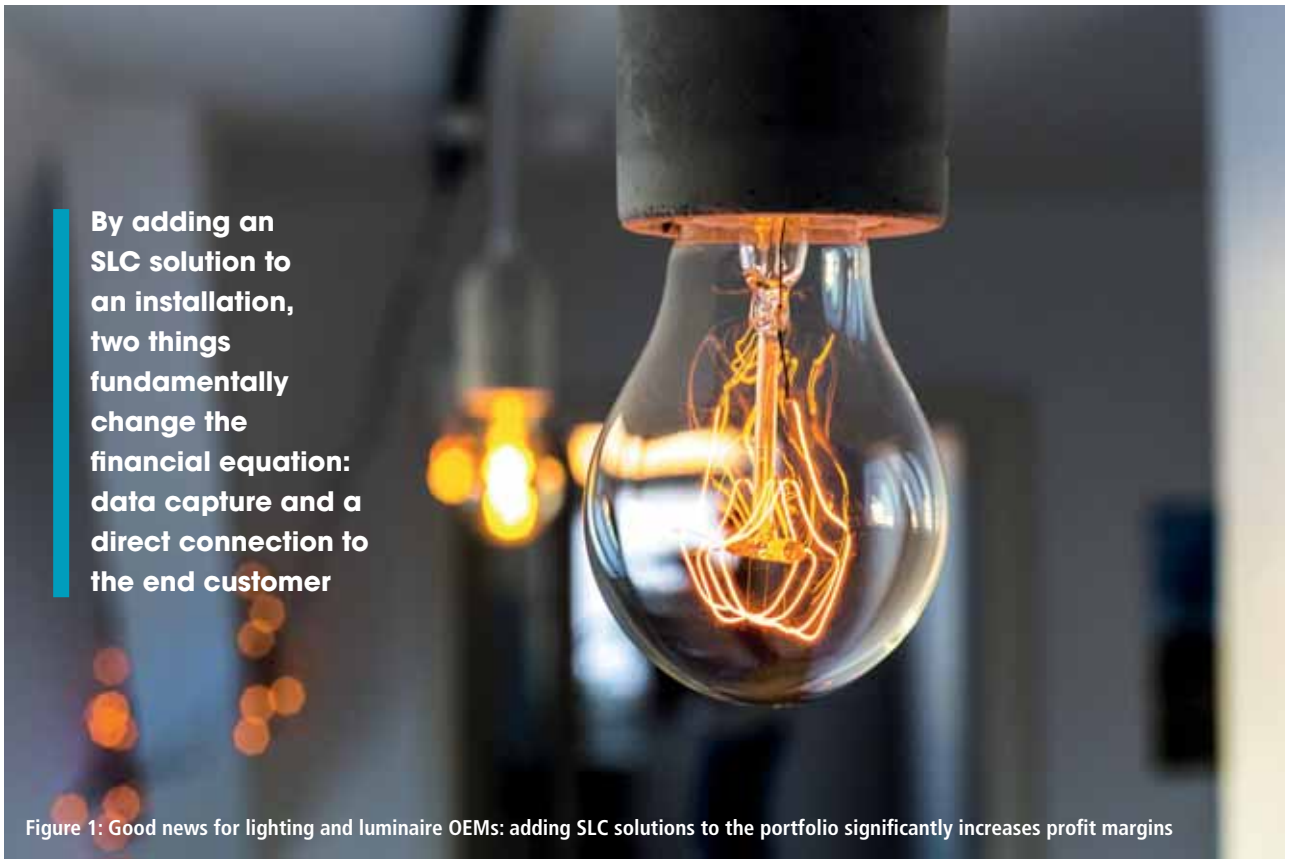



Figure 1: Good news for lighting and luminaire OEMs: adding SLC solutions to the portfolio significantly increases profit margins

the Cortet Lighting Control Solution), total profit margin can increase by as much as 50% over standalone lamps and luminaires. This is especially with luminaire and lamp manufacturers seeing their profit margins squeezed by rapidly increasing competition.

## Growth Wave

SLC also enables the growth of subscription-based, recurring revenue streams over time. This aspect of the SLC opportunity is just beginning to emerge, but stands to be one of the greatest future financial benefits of IoT-enabled lighting, being ready when the growth wave comes.

By adding an SLC solution to an installation, two things fundamentally change the financial equation: data capture and a direct connection to the end customer.

Since the devices (luminaires, lamps, sensors, networking devices, etc.) are networked, an SLC solution has the inherent ability to capture information from them, which can be saved, analysed and used to create more value-added, subscription-based Internet services.

Furthermore, new features can be delivered using networked devices such as location-based services (LBS), artificial-intelligence- (AI-) driven energy, space optimisation services, and so on. Although such features are only now starting to emerge, future possibilities seem limitless.

Perhaps by partnering with a software-as-a-service (SaaS) and data analytics firm, you begin offering a subscription-based service for optimising energy efficiency and cost savings. The sensors from an installed SLC solution provide real-time usage and building occupancy data, combined with, say, data on sunlight levels. The solution then automatically adjusts luminaire brightness levels in the building for optimum energy usage without sacrificing productivity.

Most system management is remote, via the web and through mobile devices. For example, let's say an end-customer pays a monthly subscription of $10, or $120 annually. In return, the customer saves $160 in energy costs each year, so the solution pays for itself. For 1,000 such contracts, $120,000 would be added in highly-profitable recurring software revenue.

This is just one example of the subscription-based revenue opportunities that SLC can bring.

Now is the time to expand the business beyond lamps, luminaires and accessories. Industry, office and manufacturing plants and facilities are rapidly moving to an IoT model. Smart lighting control technology is here, and to be competitive and successful, it's vital to take advantage of this new market opportunity. ●

# Mix and Math

**Swathi Sridhar** and **Namrata Dalvi** from Microchip Technology describe a method for RGBA colour mixing using Bluetooth low energy communications

Controlling the colour balance of light-emitting diodes (LEDs) accurately and wirelessly can be achieved using an 8-bit microcontroller and a Bluetooth 4.1 low-energy (BLE) module.

The demonstration board shown in Figure 1 has four LEDs – one each red, green, blue and amber. The brightness of each LED is controlled by a pulse-width modulation (PWM) duty cycle.

The Microchip's PIC16F1579 microcontroller has four 16-bit PWM outputs that drive the LEDs. The 16-bit PWMs allow for precise control over the intensity of each LED and the mixing of different red, green, blue, alpha (RGBA) brightness levels to create different colours.

Microchip's mTouch capacitive touch sensing technology operates the two capacitive touch sliders. The on-board RN4020 Bluetooth module receives the PWM values from an Android mobile application or desktop program using BLE communication. The board is powered by a 1.5V AAA battery.

## Lighting

The light produced by the LEDs varies due to several factors: The brightness, measured in lumens, varies for LEDs of different types and even between LEDs of the same type. For colour LEDs, the specific colour measured by the chromaticity values can also differ from one LED to another.

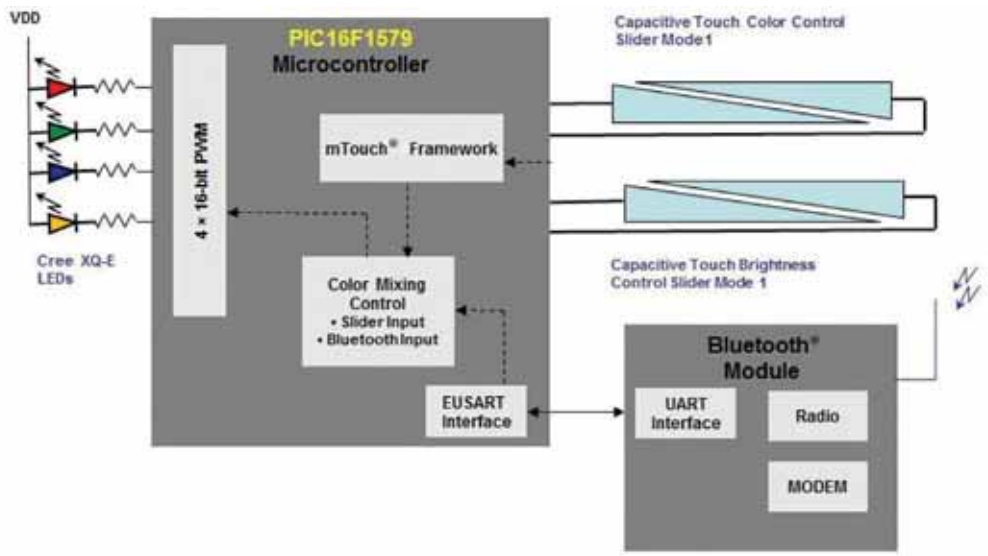



**Figure 1: RGBA colour mixing demonstration board**

Small samples of a particular LED were measured to develop a brightness and chromaticity profile. The values were then used as typical in the hardware design and in the software's chromaticity calculations. This process is called "colour tuning".

The resistor values were fixed so that each colour produced the same number of lumens. The LED series resistors are thus red 820Ω, blue 400Ω, green 500Ω and amber 500Ω.

## Operating Modes

There are two modes of operation: the first is hue saturation value plus white (HSVW) and brightness sliders mode; the second is chromaticity selector using BLE.

There are two capacitive touch sliders on the board, one for colour adjustment and the other for brightness levels; the board initially powers-up in the first mode.

If the first slider is touched while in slider mode, the colour selected on the slider is output on the LEDs, being displayed until another input is received. The brightness of a particular colour can be controlled with the other slider.

In the second mode, the colour values (PWM) are selected using Android-based mobile or Windows-based desktop applications. The respective PWM values are then sent to the board via Bluetooth.

The application uses the CIE 1931 XY chromaticity chart, shown in Figure 2. The exact PWM values for the selected colour and brightness levels are computed and sent to the RGBA board over the Bluetooth connection. The Bluetooth module on the board receives the PWM values, which are then used by RGBA board firmware to display the selected colour.

The chromaticity selector application GUI consists of the CIE 1931 xy chromaticity chart. The CIE 1931 colour space shows a wide range of colours in terms of chromaticity (x) and luminance (y). The colour and brightness levels of red, green and blue LEDs mapped onto the CIE colour space define a triangle that encompasses all possible shades that can be generated by the output of three devices; this is known as the colour gamut.

To obtain a better range of colours, an amber LED has been added. Its xy data are mapped onto the CIE 1931 xy colour space. This defines another triangle between the red, amber and green coordinates. Mixing red, amber and green in different proportions produces the colours within the colour gamut of Figure 2.

The PC GUI and the Android applications used in this mode implement the colour-mixing algorithm to calculate the PWM duty cycle values necessary to produce the desired colour.

The chromaticity selector application sends the PWM values over a Bluetooth connection, which will be able to communicate with mobile phones and PCs with Bluetooth v4.0 (and higher) transceivers. The module is primarily used for receiving duty cycle values from master devices that run the chromaticity selector application. The pin-to-pin connections between the microcontroller and BLE module are shown in Figure 3.

## Bluetooth Communication

There are two types of Bluetooth devices – Bluetooth classic and Bluetooth low energy (BLE). A BLE device can only communicate with another BLE device or Bluetooth dual-mode device, with both classic and low energy capabilities. Hence the master host device must be BLE or Bluetooth dual-mode to communicate with the RN4020 module used on the RGBA board.

The module complies with the Bluetooth core v4.1 specification and is controlled through input and output lines and a UART interface. The UART supports ASCII commands to control or configure the module for any requirement based on the application.

## Application Software

When the board is operating in the second mode, the desired LED colour is selected from the chromaticity chart in the chromaticity selector application either from the RGBA colour mixing desktop or Android application. The red, blue, green and amber PWM duty cycles are calculated by the application. Duty cycle values are passed on to the board by a BLE connection. The desktop application in this example was developed using Visual Studio C#.NET. The application follows the MVC principle with various classes.

The RGBA view controller class acts as the GUI or view manager and as the controller of the application. This class is at the top of the hierarchy responsible for making new objects of classes and performing dependency injection; it also handles all the GUI events and calls appropriate methods.

The RGBA calculation class is responsible for determining if

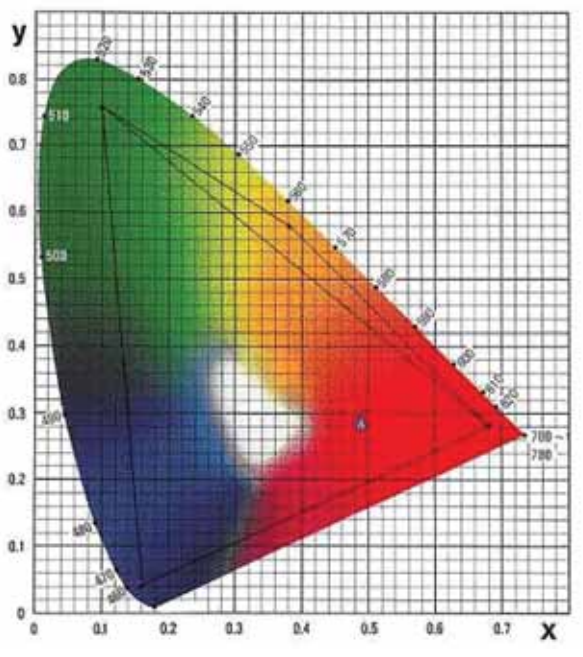



**Figure 2: CIE 1931 colour space with RGBA LED colour gamut**

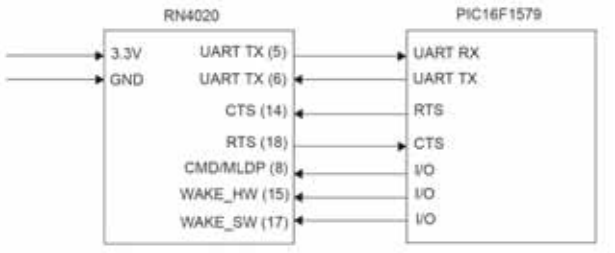



**Figure 3: Interface between the Bluetooth low energy (BLE) module on the left and a microcontroller**

the selected point is inside or outside the RGB or RGA triangles, and calculates the duty cycle per colour for all LEDs.

The matrix 3x3 class implements math operations such as inverse, determinant, transpose, co-factor and multiply. The Vector 3 class implements a column vector of size three to be used in matrix maths for the matrix 3x3 class. The RGBA data class is a custom data type to store the duty cycle values of all colours.

In the wireless communications wrapper class, the interface contains all the methods required by the wireless communication to implement the RGBA application. This interface can be used by any wireless communication method such as BLE or Bluetooth classic. The BLE communications are done using the RN4020 PICtail card through RS232 connection by implementing this interface for the RGBA board.

The programmer can make a new class to implement wireless communication through built-in BLE libraries in Visual Studio or third-party libraries. This interface decouples implementation of the communication from the actual controller, so if new communications are implemented, the view controller and other classes will not change.

The RGBA BLE communications via the RN4020 device class implement the wireless communications wrapper interface for BLE communication with the RGBA board. The PICtail card is used and connected to a PC via the UART or RS232 port. Serial communications are established and commands sent for BLE communication.

The BLE device information class stores the basic information about the remote connectivity device, such as its name, address and supported server service, used to identify and connect to a remote device.

In the search result delegate class, the delegate services the event from the BLE class when it finishes the search for devices, which are then available as a list for the user. The search time is ten seconds.

With the connection state-change delegate class, the delegate services the event from the BLE class to determine if the master PICtail card is connected to a remote device, and displays the current connection state to the user.

The constants class stores all the constants required for the application, such as RN4020 module commands and responses, service and characteristic UUIDs, and so on.

The Java application class for the Android operating system follows the MVC principle closely as well, using Android activity classes that are structurally similar to the desktop application. However, the Android application uses the built-in BLE hardware of the Android phone. The Android OS provides all the necessary libraries for BLE communications with all required events and call-backs.

The RGBA view activity class is similar to the view controller class on a desktop, except for the GUI controls, which are defined in an XML file instead of a class.

## Precise Control

This article has shown how 16-bit PWM can precisely control the intensity of each LED. The RGBA LED-colour-mixing board described has slider, capacitive touch-buttons for colour input and brightness control. A Bluetooth 4.1 low energy module is used for communication, enabling the user to send PWM values to the RGBA board to output the desired colour. The colour was selected on a chromaticity selector application on a Windows desktop and an Android-based phone. ●

# Electronic phosphor offers many new capabilities

By **Marián Štofka**, Slovak University of Technology, Bratislava, Slovakia

Phosphorescence in some minerals and chemical compounds is the ability to emit a faint, yet visible light after being exposed to daylight or other, stronger, light sources. One such widely used material is zinc sulphide (ZnS), later replaced by strontium aluminate ($SrAl_2O_4$) that emits more intense light. Man-made electronic phosphor, on the other hand, emulates phosphorescence by lighting up an LED after a solar cell is illuminated.

## Phosphors

Phosphors or luminescent materials are used in modern lighting, especially when the light output needs a hue, akin to that of daylight, such as in white LEDs (WLEDs).

The semiconductor chip of a typical WLED emits highly monochromatic blue light that passes through a layer of translucent phosphor, which then converts part of it into yellow, creating a mix of the two colours to reproduce daylight. For WLEDs, ceramics phosphors such as YAG (Yttrium-Aluminum Garnet) are used, doped with $Ce^{3+}$ (Cerium). Other phosphors based on LuAG (Lutetium-Aluminum Garnet) convert blue-LED light into either green or red-orange, depending on the dopant.

## Terminological Maze

As mentioned earlier, in modern WLEDs phosphor is a key material. The general term describing the activity of a phosphor in a WLED is photoluminescence, while in a semiconductor chip it is electroluminescence.

More specific terms for photoluminescence in this case are phosphorescence and fluorescence, where phosphorescence is associated with a significant afterglow – more than 10ms, and fluorescence less than 10ns.

Fluorescence is a widely accepted term associated with WLEDs, although the afterglow of a typical phosphor YAG:$Ce^{3+}$ is around 70ns.

Interestingly, Encyclopaedia Britannica states that the "distinction between the terms phosphorescence and fluorescence is still open to discussion". Until such a discussion is settled, we should accept that a typical WLED contains phosphor that fluoresces.

## Electronic Phosphor

An optoelectronic circuit comprising a receiving or harvesting part of a photodiode(s) and an emitting part formed by an LED(s) can be considered "electronic phosphor" if it reacts

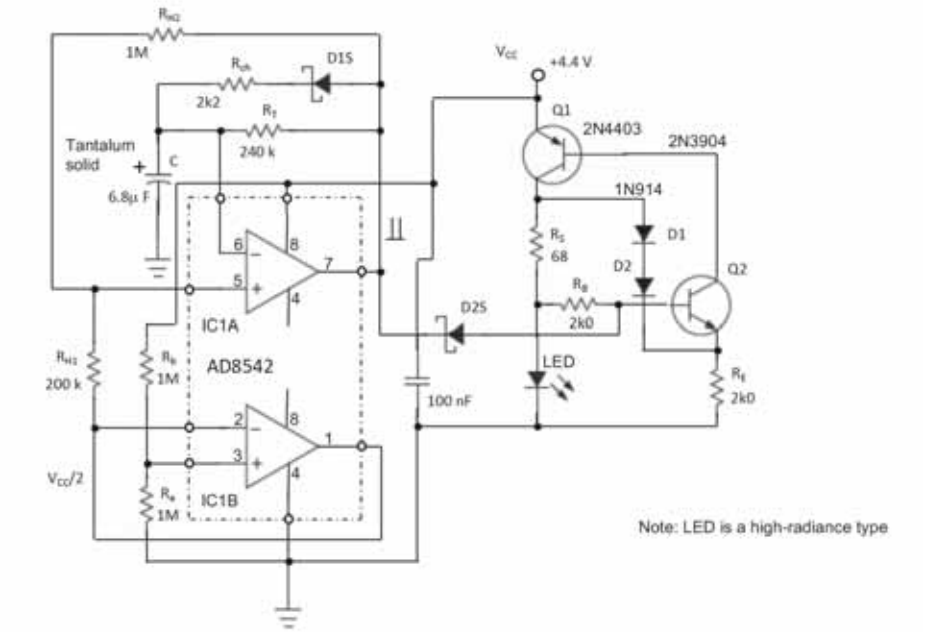
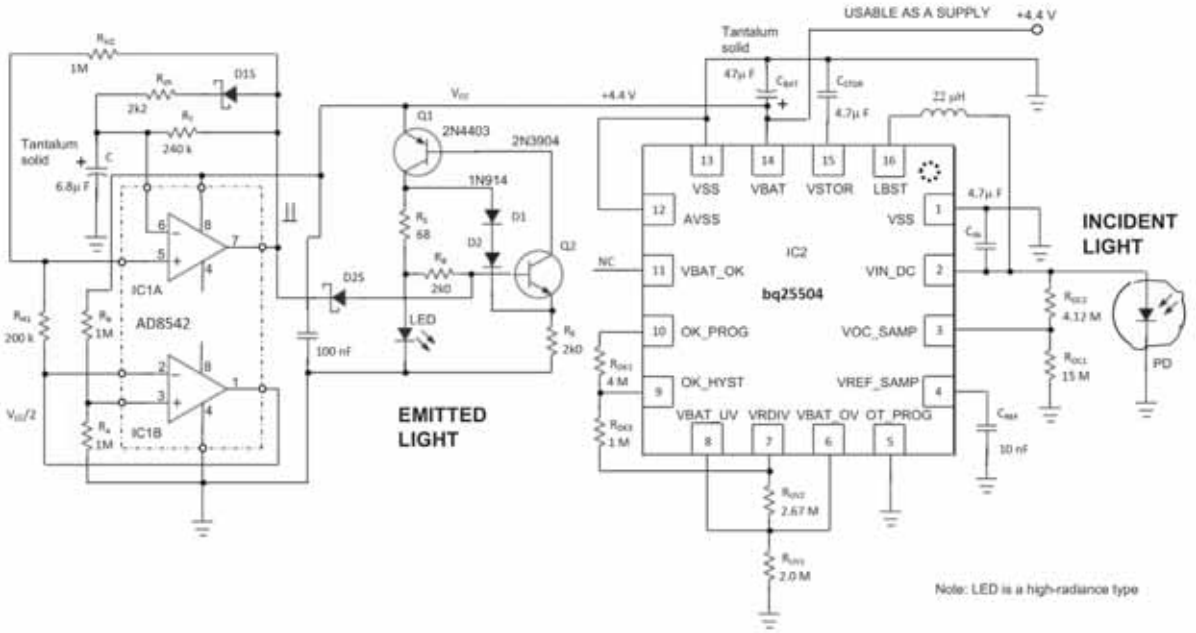



**Figure 1: The LED blinks with a frequency of about 2Hz, while the mean total power consumed is a mere 700µW**

Figure 2: The circuit idles in the dark. Upon illumination, for example by a searchlight, the LED blinks with a frequency of 2Hz




sufficiently fast to sudden illumination. Unlike phosphor the material, which converts light of one wavelength into another – specifically a longer one, electronic phosphor offers vast opportunities to become "smarter". It can also convert light from longer to shorter wavelength, such as red to blue light – a task that can scarcely be fulfilled by nature's phosphors.

Further, the output of an electronic phosphor can be made to pulsate. If this pulsed output is modulated, e. g. PWM, it could carry information from the surroundings of the electronic phosphor. Also, the light output could be delayed for an arbitrary time or re-started upon cease of illumination.

For an electronic phosphor, the emitting part must be of high efficiency to allow operation, even at lower intensity of illumination, therefore an efficient pulse generator with a micro-power operational amplifier is used, as discussed next.

## LED Blinker as a Micropower Circuit

The circuit in Figure 1 is a high-efficiency light source. IC1 is a dual micropower op-amp, in this case AD8542 by Analog Devices. The IC1B op-amp is connected as a voltage follower, which together with the resistive divider comprising $R_a$ and $R_b$ resistors forms a low output impedance source of common voltage:

$$V_{COM} = \frac{V_{CC}}{2}.$$

A hysteresis of the "inverter" with the IC1A is created by a positive feedback, formed by $R_{H1}$ and $R_{H2}$ resistors, with a value of:

$$V_{HYST} = V_{CC} \times \frac{R_{H1}}{R_{H1} + R_{H2}} \quad (1)$$

The $V_{HYST}$ value has been chosen as 0.748V and, for supply voltage, $V_{CC}$ = 4.4V.

It follows from Equation 1 that the resistor ratio:

$$\frac{R_{H2}}{R_{H1}}$$

should be:

$$\frac{R_{H2}}{R_{H1}} \cong 4.9$$

What cannot be seen directly from the circuit schematic is that the supply current of the IC1A remains low, regardless of the voltage at its inverting input varying continuously up and down around the supply's mid-voltage $V_{CC}/2$. The net result is blinker power consumption of only about 700µW.

This circuit's complexity might seem a bit high, especially by adding four resistors, but its footprint rises negligibly when using surface-mount devices.

## Creating Electronic Phosphor

The Texas Instruments bq25504 energy-harvesting DC-DC converter IC2 in Figure 2 does not operate in the dark, and the LED is off. When a source of visible or infrared light

illuminates the solar cell PD, IC2 starts up and feeds the LED blinker formed by IC1 and transistors Q1 and Q2. The LED blinks with a duty cycle of about 1% and repetition frequency of about 2Hz.

Response of the LED blinker is delayed by $T_D$, calculated with:

$$T_D \cong \frac{\frac{1}{2} C_{BAT} \times V_{BAT}^2}{\eta(P_{PD}) \times P_{PD}} \quad (2)$$

where $P_{PD}$ is the power delivered by the photocell PD from incident light and $\eta(P_{PD})$ is the power efficiency of the DC-DC converter. For

$$C_{BAT} = 47\mu F, V_{BAT} = 4.4V, P_{PD} = 10mW, \eta(10mW) \cong 0.8,$$

$T_D$ is 57ms, which is acceptable for human-machine interaction. Experiments were performed after dusk in a room using only one desktop lamp with a 75W incandescent bulb. The phosphor reliably responded by starting up the blinker at a 1.2m distance between bulb and photocell and light incidence angle of about 60°.

The operation continued at increased distance of 2m with incidence angle of around 30°, although at this position,

$$\frac{1}{\sqrt{3}} \left( \frac{1.2}{2} \right)^2 \cong 0.208$$

At about 1.5m with 45° angle of incidence, upon shielding the solar cell for about 5s, circuit operation stops, without a restart after uncovering the PD.

Experiments show that for a source of fainter light or increased operable distance, two photovoltaic diodes should be used connected in series, because a single silicon photovoltaic diode under faint light conditions has low open-source voltage – about 0.29V.

## Added Capabilities

The circuit of Figure 2 has a blink-only capability, but by adding further micro-power sensors and comparators, many other capabilities can be introduced, such as changing the blinking frequency on crossing a certain level of ambient temperature, changing the colour of emitted light to blue upon detecting frosty conditions in the environment, and more.

This circuit is particularly suitable for autonomous signage in dark conditions, such as in speleological research, for example.

# Introduction to LEDs and their applications

BY **STOJCE ILCEV DIMOV**, DURBAN UNIVERSITY OF TECHNOLOGY (DUT), SOUTH AFRICA

The lighting sector has undergone a substantial transformation and improvements in recent years thanks to the light emitting diode (LED), with new possibilities and functions for lighting aplenty. Advances in LED technology allowed for a shift in lighting functions from purely brightening to human-centric; for the first time in history lighting is not only for illumination, but now adds to the emotional and biological wellbeing of people and even plants.

## LEDs in Detail

In the mid 1920s, Russian radio researcher and inventor Oleg Vladimirovich Losev was studying the phenomenon of electro-luminescence in diodes used in radio sets. He created the first LED, but his research was ignored at that time. In 1927, he published a paper called *'Luminous carborundum [silicon carbide] detector and detection with crystals'*, and while no practical LED was created based on this work, his research influenced future inventors.

In 1961, Robert Biard and Gary Pittman invented and patented an infrared LED for Texas Instruments. This was the first commercial LED; however, being infrared, it was beyond the visible light spectrum and therefore not directly visible. Ironically, Baird and Pittman only accidentally invented the light emitting diode while attempting to create a laser diode.

An LED emits light when current passes through it. Early LEDs produced only red light, but modern LEDs can emit other colours, including green and blue, and more recently white light, too.

LED circuits are commonly used for indicator lights on equipment and electronic devices, but also for electronic signs and cars, among others. Since LED circuits are energy-efficient and have a long lifespan (often over 100,000 hours), they are beginning to replace traditional light bulbs in many areas, including street lighting and displays. LEDs produce brighter light than other sources while using less energy. As such, traditional flat-screen liquid-crystal displays (LCDs) are being replaced by LED displays, where LED circuits are used in backlighting. In addition, LED TV screens and computer monitors are typically brighter and thinner than their LCD counterparts.

## LED Circuit Basics

The LED structure and current flow are shown in Figure 1, with an LED circuit diagram and main components in Figure 2.

An LED consists of a semiconductor structure called a p-n junction that emits light when under current.

The simplest circuit to power an LED is a voltage source with a resistor – often called a ballast resistor – and an LED in series. The ballast resistor is used to limit the current through the LED and to prevent burnout. If the voltage source is equal to the LED voltage drop, no resistor is required.

The ballast resistor's value is easy to calculate with Ohm's and Kirchhoff's laws:

$$R = V - V_{LED}/I$$

where V is voltage source, $V_{LED}$ is LED voltage and I LED current. The voltage drop across an LED is relatively constant over a wide range of operating currents, so a small increase in applied voltage greatly increases the current.

Very simple circuits are used for low-power indicator LED devices, whereas more complex ones are required when driving high-power LED devices to achieve correct current regulation.

The specific wavelength or colour emitted by the LED depends on the materials used to make the diode. Red and infrared LEDs use

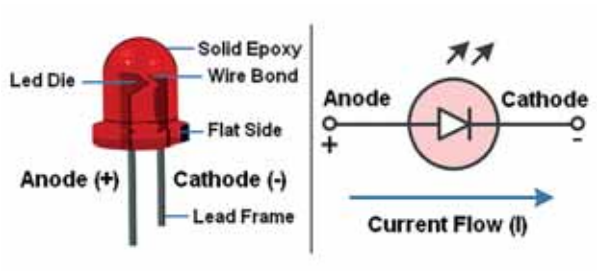



Figure 1: LED structure and current flow

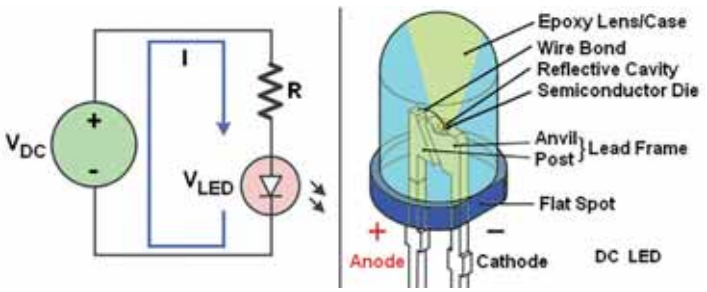



Figure 2: Circuit diagram and main components of an LED

aluminum gallium arsenide (AlGaAs); blue, green and ultraviolet high-brightness LEDs are made from indium gallium nitride (InGaN); yellow, orange and red high-brightness LEDs are based on aluminum gallium indium phosphide (AlGaInP); whereas yellow and green LEDs on gallium phosphide (GaP).

## LED Types and Applications

Properly powering LEDs is different from powering most other electronic devices: while most electronics require a constant voltage source, an LED requires a constant current source.

Most power supplies will provide a set voltage, such as 5V, and have a maximum current rating. The power supply will maintain this voltage and provide enough current to power the circuit, which means a dedicated power supply needs to be implemented to drive any LED.

For an LED to properly operate, the power supply must have a high enough voltage to turn it on and provide a controlled constant current through it. The voltage required to power the LED is called forward voltage ($V_f$), and the amount of current the LED uses when illuminated is called forward current ($I_f$).

Forward current is typically specified as a maximum value, and current above this rating can damage the LED. Equally, the LED light output varies with the forward current; decreasing the forward current decreases light output.

Early LED devices were often used as indicator lamps for electronic devices, replacing small incandescent bulbs. They were soon packaged into numeric readouts as seven-segment displays, commonly seen in digital clocks.

Recent developments have produced LED devices suitable for environmental and task lighting. Unlike a laser, the colour of light emitted from an LED is neither coherent nor monochromatic, but the spectrum is narrow with respect to human vision. LEDs have led to new displays and sensors, while their high switching rates are useful in advanced communications technology.

LED products can be divided into two categories, indication and illumination.

Indication LED devices are very low power, with a forward current requirement of typically between 10mA and 20mA. This doesn't have to be precise, as normally there's only one LED used for each indication, and slight variations in luminous intensity from one indicator to another are not important, as long as they are of relatively same brightness.

Illumination LED devices are used as a light source to view other objects, such as general lighting found in most rooms, or task lighting found on many desks; see Figure 3.

## LED Power Sources

Much more power is needed to power lighting LED devices than indicator LED circuits, dividing them further into two groups, low and high-powered LED circuits.

Low-powered LED devices use 1.5mA in a bicolour LED, and some even 0.5mA; see Figure 4, left.

High-powered LED devices, such as iJDMTOY Super-Bright and high-power LED bulbs produced by CREE or Osram (Figure 4, right),

**Figure 3: LEDs for indication and illumination**

which can produce at least twice as much light compared to SMD LED emitters, for example. Operational life can be up to 50,000 hours compared to incandescent bulbs' 500 hours.

The main function of LED drivers is to deliver constant current across a range of operating conditions, regardless of the input or output. The driver requirements are determined by the type of LED device and its operating parameters.

LED driving techniques are based on several methods:

- Resistor-limiting, where a resistor acts as a current limiter, but is not used as a current control circuit. Such a circuit is simple and cheap, but has the disadvantages of poor efficiency, current variations due to changes in forward voltage, and resistor heat generation. The simplest way to power an LED with a DC constant-voltage source that's already powering other devices in the circuit. Current can be controlled with a series resistor; see Figure 2, left.
- Linear regulation is another simple method, where a resistive LED device uses a linear IC with constant current source to control its current, providing constant current to the LED over the entire supply voltage range. This also provides diagnostic functions, and external resistors can be added to adjust the LED current.
- A DC/DC converter can provide constant current to the LED over the entire supply voltage range; see Figure 5. There are several topologies, including buck and boost. This method reduces power loss by optimising the LEDs' chain length. It provides highest efficiency and high-temperature protection, but adds to the component count.

**Advances in LED technology allowed for a shift in lighting function from purely brightening to human-centric**

Rather than use a current-limiting resistor with a constant voltage source for powering LED devices, it would be better to design a constant-current power supply. Although there are very simple linear constant-current supplies available, a switched mode power supply (SMPS) is more efficient (Figure 6). For example, if a linear regulator converts 12V to 3.5V for a load of 350mA, the total power consumed is 12V x 0.350A = 4.2W.

SMPS controllers are of course more complex than linear regulators and typically consist of a controller IC with a high-side MOSFET, catch diode or a low-side MOSFET, inductor, and resistors

Figure 4:
(left) Low- and
(right) high-powered LED devices

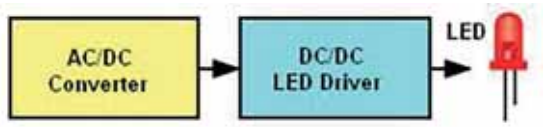



Figure 5: Block diagram of DC/DC converter for driving an LED

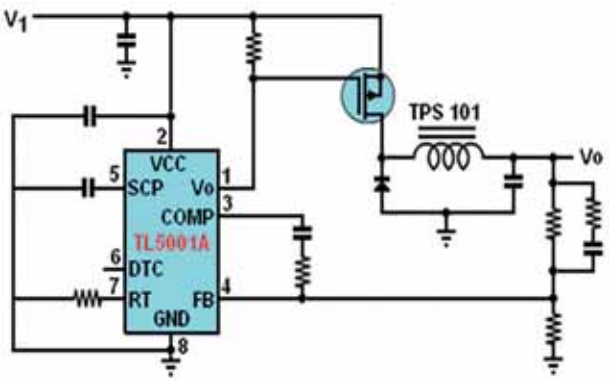



Figure 6: Constant-current switch-mode power supply

and capacitors for feedback, compensation and filtering.

Depending on the SMPS type, either a catch diode or a low-side MOSFET can be used, although some integrate a high-side MOSFET.

## LED Device Connections

In applications with multiple LED devices, another consideration is whether to connect these devices in series or parallel.

A circuit with LED devices in series consists of multiple LEDs connected to a single voltage source, where multiple resistors share the same current; see Figure 7, left. Because the current through all the devices is equal, they should be of the same type.

Note that lighting one LED in this circuit uses just as much power as multiple LED devices in series. The voltage source must provide a large enough voltage for the sum of voltage drops of the LED devices plus the resistor. Typically, the voltage source is 50% higher than the sum of the LED voltages.

A circuit with LED devices in parallel is more problematic; see Figure 7, right. The forward voltages of the LED devices must closely match, otherwise only the lowest-voltage LED lights up and may burn out from the larger current. Even if the LED devices all have the same specification, they can have bad matching I-V characteristics due to production variations, causing the LED devices to pass a different current. To minimise current, LEDs in parallel normally have a ballast resistor for each device.

Disadvantages in powering LEDs in parallel include varied light output between LEDs; if an LED fails open, the other LED devices can be damaged; and the amount of current required from the power supply increases with each additional LED.

On the other hand, LED assemblies often require high power and attention to the quality of light output. Efficient power supplies are imperative; SMPSs are very common and can have efficiencies above 90%. Thus, connecting LED devices in series eliminates variations in current from one LED to another, the need to fault monitor individual LED devices, and the need for high-current components.

For high-power lighting applications requiring multiple LED devices, configuring them in series and powering them with an efficient constant-current SMPS should be the first approach. ●

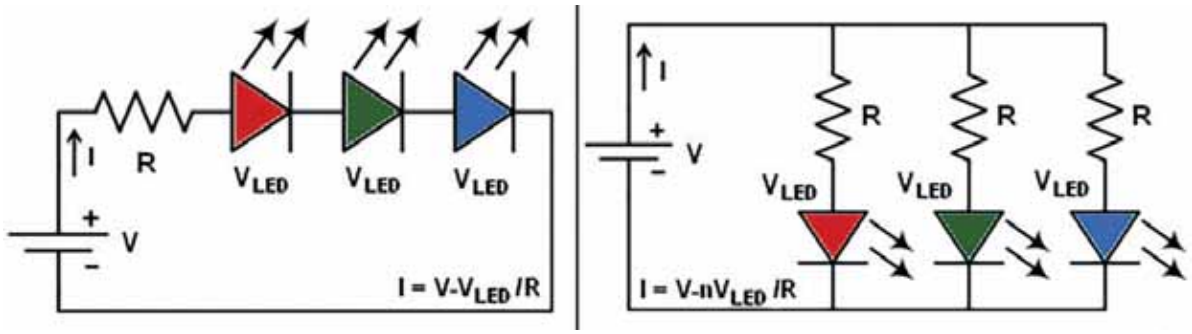



Figure 7: LEDs connected in series and parallel

# Driving an LED beam with a single MCU port line

By **Huijie Li, Junting Li** And **Zhiqi Lin**, Changchun University of Technology, China

An LED beam is made up of lights from several LEDs. The beam improves on analogue indicators used in various applications for several reasons. First, the beam is bright and prominent, hence easy to see from a distance. Then, it doesn't need any mechanical parts, and, lastly, it gives accurate indication and correct reading through a chip matrix array.

On the flip side, however, controlling an LED beam requires several port lines between the LEDs and an MCU. Here, we discuss a new method of controlling an LED beam, based on a modified boost circuit that reduces the number of MCU ports.

## Power Stage

The boost circuit is a popular non-isolated power topology, sometimes called a step-up power stage. Power supply designers choose the boost power stage when the required output is higher than the input voltage. The input current for a boost power stage is continuous, or non-pulsing, because the output diode conducts only during a segment of the switching cycle. The output capacitor supplies the entire load current for the remainder of the switching cycle.

Figure 1 shows a simplified schematic of the boost power stage. Inductor L and capacitor C make up the output filter. The capacitor equivalent series resistance (ESR), $R_C$, and inductor DC resistance, $R_L$, are also included in the analysis. Resistor R represents the load seen by the power supply output.

A power stage can operate in continuous or discontinuous inductor-current mode. In continuous inductor-current mode, current flows continuously through the inductor during the entire switching cycle in steady-state operation. In discontinuous mode, this current is zero for a part of the switching cycle; it starts at zero, reaches a peak value and then returns to zero, for each switching cycle.

Because the power stage frequency response changes significantly between the two modes of operation, it is desirable that the power stage stay in only one mode for its operation. However, the standard boost circuit can only boost voltage, so it's not free to control several LED lights.

In this article we discuss how to improve the boost circuit to achieve automatic voltage levels to control several LEDs.

## LED Light Uses

LED lights are widely used in instruments. One example is the light-column liquid-level display instrument consisting of four main sections: pressure sensor, MCU, power and passive components. The pressure sensor measures the liquid level; its output goes to the MCU, which analyses the data and then correspondingly controls the LEDs.

An LED beam comprises multiple LED outputs, with each LED needing a port line to connect to the MCU. The connection between MCU and display is normally through a specific interface, but this increases circuit costs. We propose a simple and inexpensive

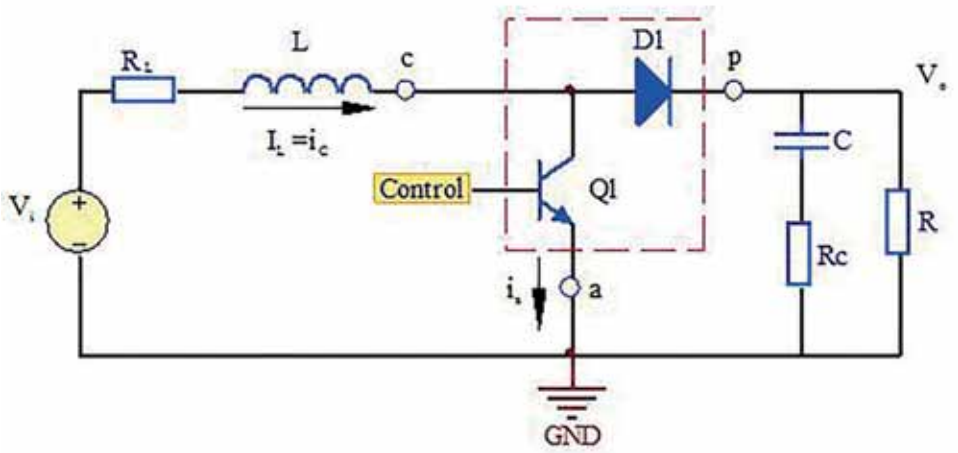
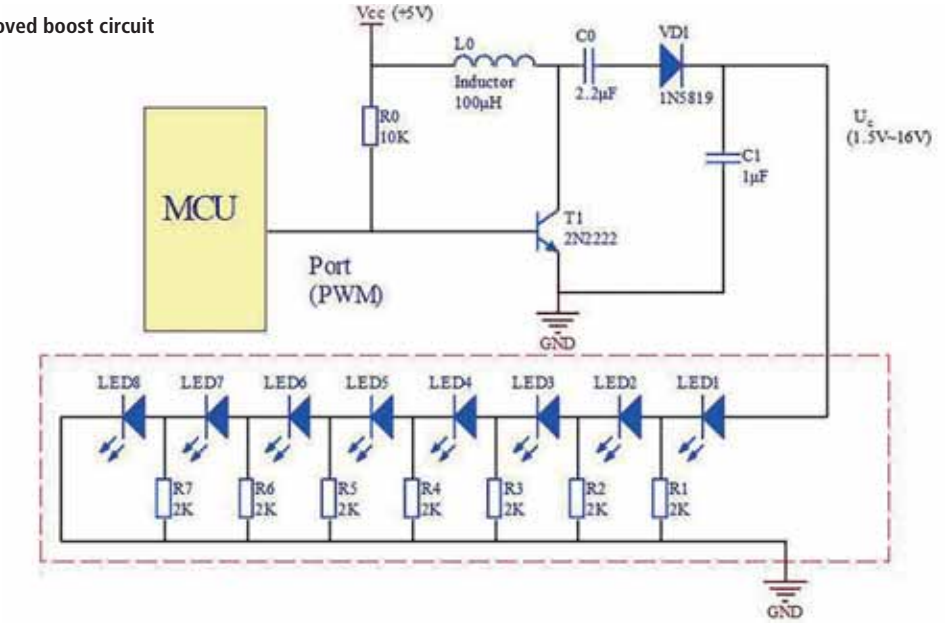



**Figure 1: Boost power stage schematic**

Figure 2: Improved boost circuit



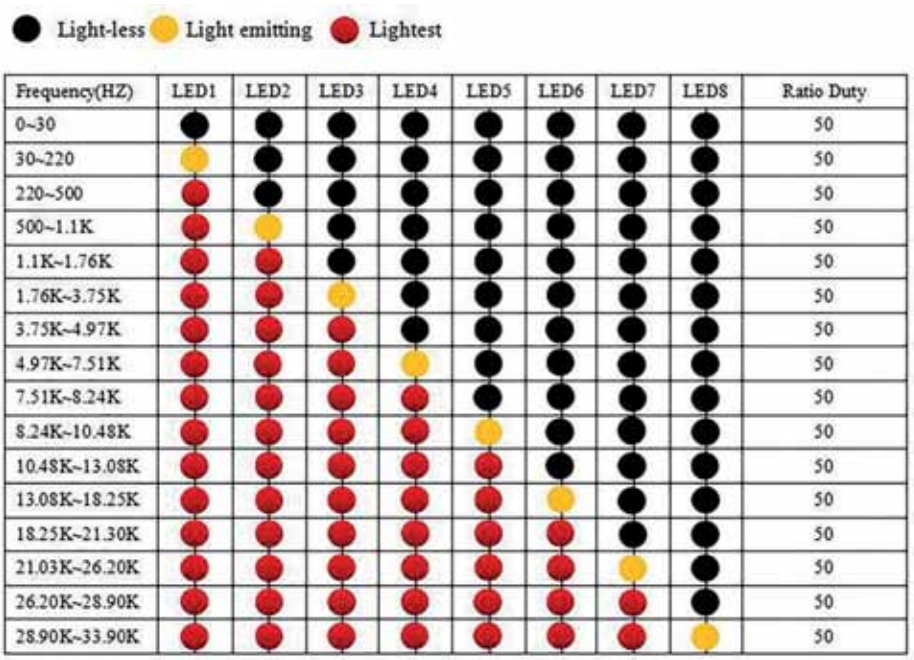

method for interfacing an LED circuit using a single port of the MCU and resistors, capacitors and transistors, as shown in Figure 2.

In the figure, the part inside the red dashed line is the LED beam circuit, where voltage $U_c$ controls the LEDs. When this voltage is between 0 and $U_d$ (which is each LED's voltage drop), LED1 forms a loop through R1 and lights up. Because the voltage is lower than $U_d$, the rest of the LEDs don't turn on.

When $U_c$ is between $U_d$ and $2U_d$, LED1 and LED2 light up, and so on. With increased $U_c$, LED1~LED8 light up in turn, forming a functioning LED beam.

The upper part of the circuit in Figure 2 is a DC/DC converter. Its output voltage is controlled by the MCU through PWM to control $U_c$ with only one port line, which controls the beam.

As shown in Figure 2, the DC/DC converter circuit requires




| Frequency(HZ) | LED1 | LED2 | LED3 | LED4 | LED5 | LED6 | LED7 | LED8 | Ratio Duty |
|---|---|---|---|---|---|---|---|---|---|
| 0~30 | ● | ● | ● | ● | ● | ● | ● | ● | 50 |
| 30~220 | ● | ● | ● | ● | ● | ● | ● | ● | 50 |
| 220~500 | ● | ● | ● | ● | ● | ● | ● | ● | 50 |
| 500~1.1K | ● | ● | ● | ● | ● | ● | ● | ● | 50 |
| 1.1K~1.76K | ● | ● | ● | ● | ● | ● | ● | ● | 50 |
| 1.76K~3.75K | ● | ● | ● | ● | ● | ● | ● | ● | 50 |
| 3.75K~4.97K | ● | ● | ● | ● | ● | ● | ● | ● | 50 |
| 4.97K~7.51K | ● | ● | ● | ● | ● | ● | ● | ● | 50 |
| 7.51K~8.24K | ● | ● | ● | ● | ● | ● | ● | ● | 50 |
| 8.24K~10.48K | ● | ● | ● | ● | ● | ● | ● | ● | 50 |
| 10.48K~13.08K | ● | ● | ● | ● | ● | ● | ● | ● | 50 |
| 13.08K~18.25K | ● | ● | ● | ● | ● | ● | ● | ● | 50 |
| 18.25K~21.30K | ● | ● | ● | ● | ● | ● | ● | ● | 50 |
| 21.03K~26.20K | ● | ● | ● | ● | ● | ● | ● | ● | 50 |
| 26.20K~28.90K | ● | ● | ● | ● | ● | ● | ● | ● | 50 |
| 28.90K~33.90K | ● | ● | ● | ● | ● | ● | ● | ● | 50 |

Figure 3: LEDs, their frequencies and ratio duty

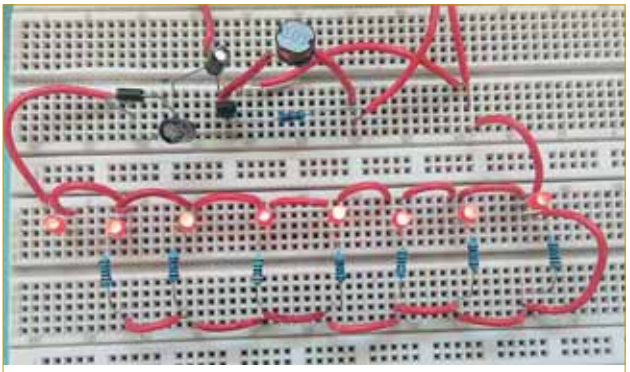


Figure 4: MCU output frequency at 33.4kHz

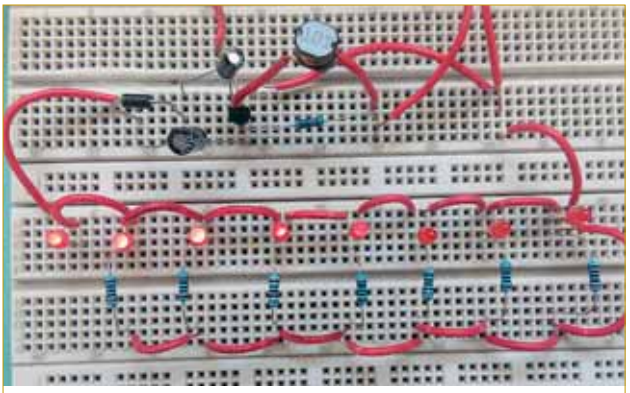


Figure 5: MCU output frequency at 8.24kHz

voltage between 0 and 15V. The MCU system is usually powered by 5V, so the converter circuit is required to automatically adjust the voltage rise and fall; for this it uses a modified boost circuit, since a standard boost circuit can only boost voltage but not buck it.

When the BJT is off, the input and output ports of the inductor are connected directly, and the output voltage is at its lowest, equal to Vcc. To change the minimum output voltage, a capacitor C1 should be cascaded with a diode, thus changing the output voltage by changing the output frequency of the MCU, when the duty ratio is stable. The relationship between the number of lit-up LEDs, frequency and duty ratio is shown in Figure 3.

## The Hardware

We built the circuit of Figure 2 on a breadboard and tested it for two different MCU output frequencies, 33.4kHz and 8.24kHz; see Figures 4 and 5.

Figure 6 shows an application of a single line driving the LED beam circuit. It adopts the 3-8 decoder output by an OC, to achieve a dynamic scan of eight branches of a light-beam circuit, such as the 74LS156; its truth table is shown in Table 1.

The working principle of each light beam is the same as in the circuit of Figure 2. It can be used in a spectrum display accompanying music for example, as shown in Figure 6.

Only one beam is selected at a time, with the working principle analogous to that of a digital display tube, but with a much simpler and cheaper circuit. ● >

Table 1: The truth table for 74LS156

| Input | | | | Output | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Select | | | Strobe or Data | 2Y0 | 2Y1 | 2Y2 | 2Y3 | 1Y0 | 1Y1 | 1Y2 | 1Y3 |
| C1 and C2 connected | B | A | G1 and G2 connected | Hoc | Hoc | Hoc | Hoc | Hoc | Hoc | Hoc | Hoc |
| × | × | × | H | Hoc | Hoc | Hoc | Hoc | Hoc | Hoc | Hoc | Hoc |
| L | L | L | L | L | Hoc | Hoc | Hoc | Hoc | Hoc | Hoc | Hoc |
| L | L | H | L | Hoc | L | Hoc | Hoc | Hoc | Hoc | Hoc | Hoc |
| L | H | L | L | Hoc | Hoc | L | Hoc | Hoc | Hoc | Hoc | Hoc |
| L | H | H | L | Hoc | Hoc | Hoc | L | Hoc | Hoc | Hoc | Hoc |
| H | L | L | L | Hoc | Hoc | Hoc | Hoc | L | Hoc | Hoc | Hoc |
| H | L | H | L | Hoc | Hoc | Hoc | Hoc | Hoc | L | Hoc | Hoc |
| H | H | L | L | Hoc | Hoc | Hoc | Hoc | Hoc | Hoc | L | Hoc |
| H | H | H | L | Hoc | Hoc | Hoc | Hoc | Hoc | Hoc | Hoc | L |

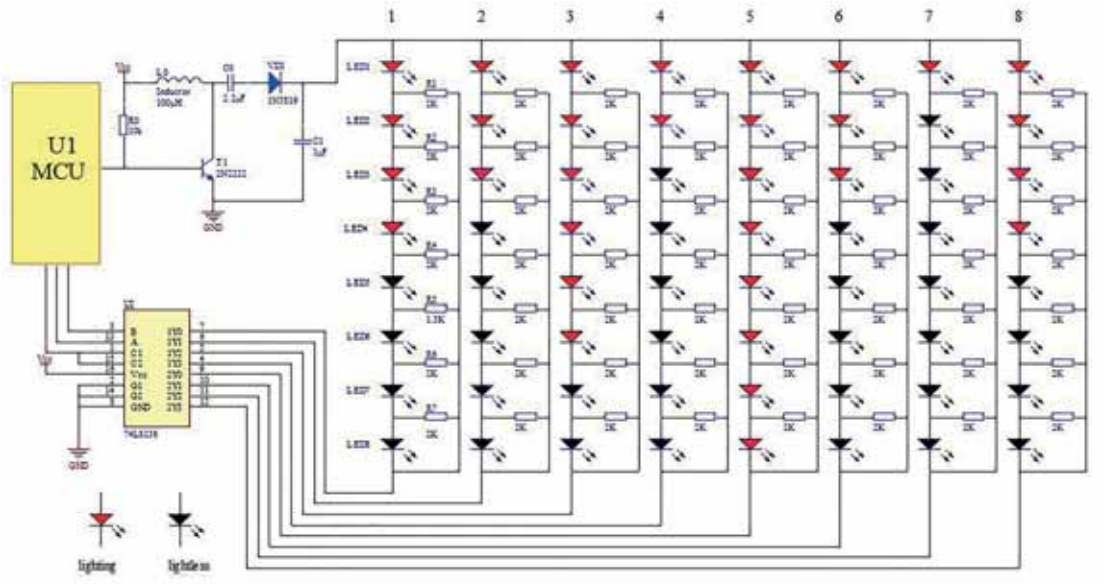



Figure 6: Circuit diagram of the spectrum display accompanying music

## NEW FAMILY OF UNIQUE 2D HALL-EFFECT SPEED AND DIRECTION SENSOR ICS

Allegro MicroSystems Europe has released a new family of 2-dimensional (2D) Hall-Effect Latch ICs that feature both vertical and planar Hall elements. The APS12625 and APS12626 sensor ICs help reduce system size and bill-of-materials (BOM) cost, along with increased performance and flexibility due to 2D sensing. The ICs are developed in accordance with ISO 26262 and qualified per AEC-Q100, making them suitable for automotive and other safety systems.

The APS12625 features speed and direction outputs whereas the APS12626 has quadrature (Channel A and B) outputs. A unique, optional feature allows the host system to restore the previous state of the sensor after a power-cycle. This reduces the potential accumulation of lost counts in intelligent motion applications such as window lifts with anti-pinch requirements.

Each IC contains a pair of sensor ICs that are orthogonal to one another.

**www.allegromicro.com**

## RITTAL HAS A NEW ONLINE PORTAL

Modernising data centres is a challenge, but one made considerably easier with Rittal's new IT microsite, which provides companies with a range of comprehensive information and advice. Visitors will get practical advice on topics such as Big Data, cloud computing and IT security, as well as case studies of real-life applications. There is a large library of white papers, brochures and references and a live-chat facility to raise questions directly with Rittal experts.

The site launch comes ahead of Rittal's attendance at this year's Data Centre World (DCW), where it will show its expertise as an enabler of innovative IT solutions, ranging from individual IT racks right through to turnkey data centres within containers which offer users maximum modularity, scalability and efficiency – all fully supported by its comprehensive consulting and service programmes.

**www.rittal.com/it-solutions/en**

# IoT Security Suite

## Making the Complex Simple


The IoT Security Suite for the SAMA5D2 MPU enables rapid and easy use of its advanced security features, such as ARM® TrustZone® technology and hardware cryptography, without a long learning curve. The suite covers the security requirements for IoT device manufacturers in a single, easy-to-use package. It supports storing, encrypting, decrypting and exchanging keys between devices and applications, and its easy-to-use APIs save you time.

SAMA5D2 Xplained Ultra Evaluation Board (ATSAMA5D2-XULT)

**Features**

- **Trusted Boot** – Root of Trust (RoT) verified startup
- **Firmware Protection** – Encryption and execution of authenticated firmware
- **Trusted Device ID** – Unique device certificate tied to the RoT
- **Secure Storage** – Secure storage of keys, certificates and data
- **Secure Communications** – Authenticated device pairing and IoT cloud communications
- **Secure Firmware Update** – Securely upgrade firmware remotely

Download the IoT Security Suite Evaluation Kit (free) to get started.

microchip DIRECT

MICROCHIP

**www.microchip.com/SAMA5D2**