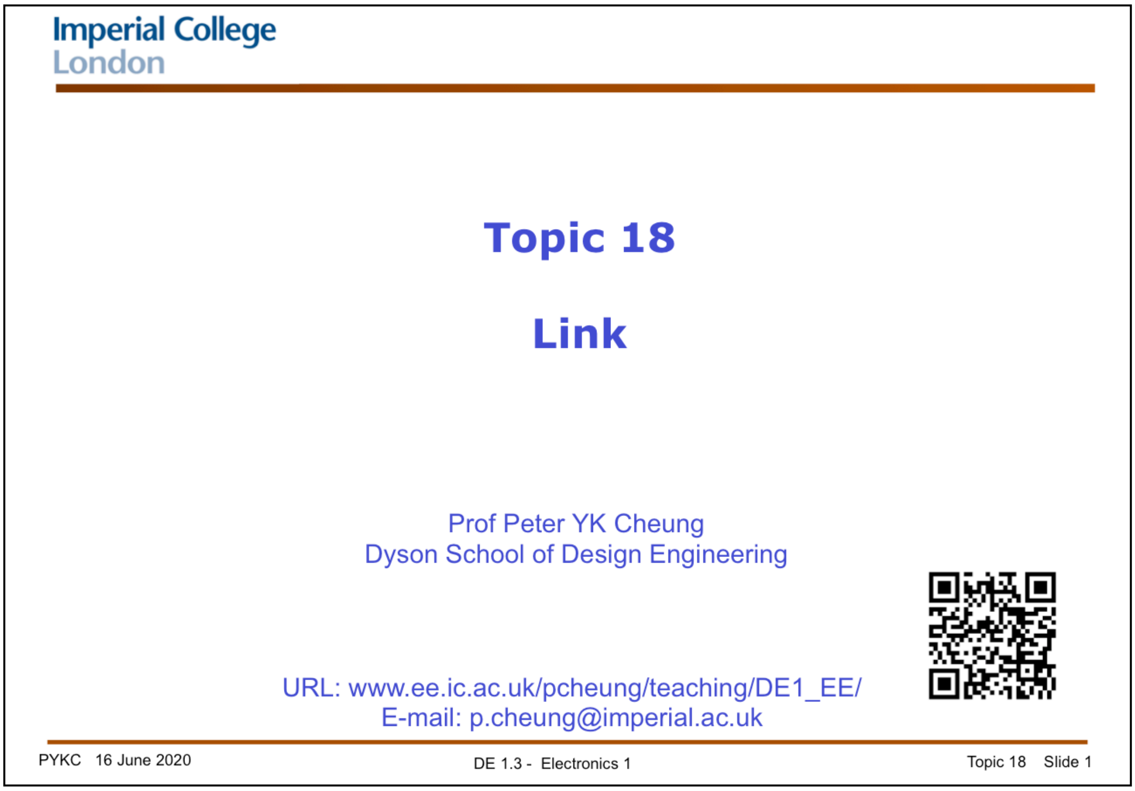
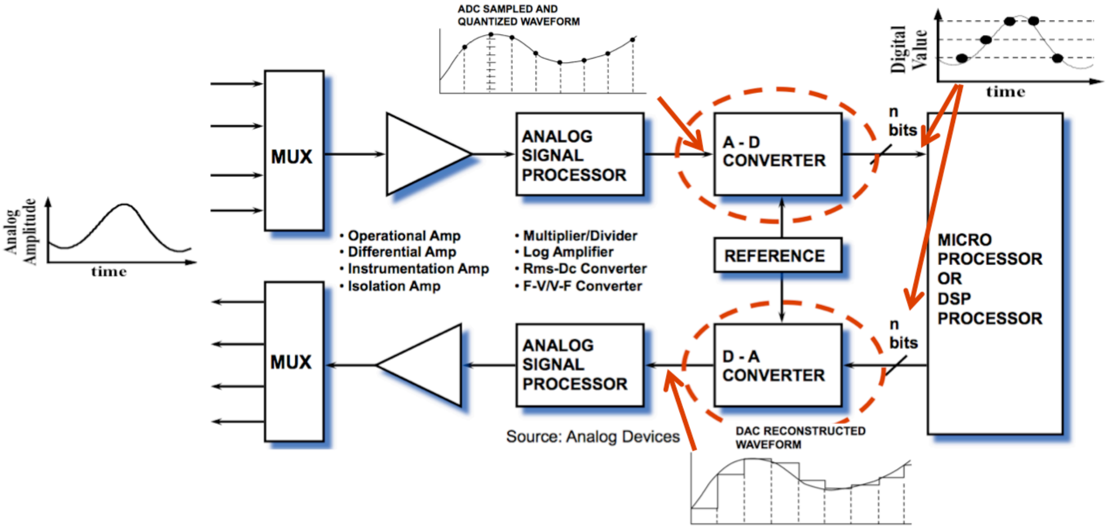


In this lecture, we will look at how different electronic modules communicate with each other. We will consider the following topics:

- Links between Digital and Analogue
- Serial vs Parallel links
- Flow control
- Wired serial links:
    - UART
    - SPI
    - I2C
- Wireless links:
    - Bluetooth and BLE
    - Radio Frequency Identification (RFID)

Linking between analogue and digital domains

- Much of the physical world is analogue in nature.
- Linking digital electronics as used in microprocessors and the physical world is three digital-to-analogue and analogue-to-digital converters. (It is common that DACs and ADCs use the American spelling of analog.)



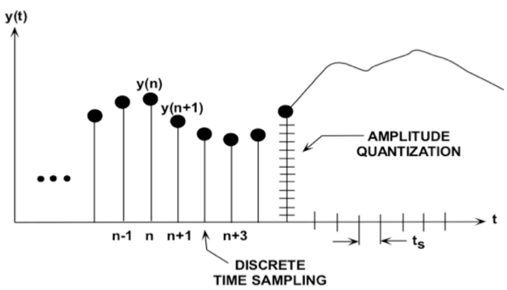
The real world problem consists of taking a continuous signal (analog) and applying decisions by means of digital signal processing to the signal. DSP allows for efficient and cost effective means of allocating information (bandwidth, capacity) correctly.

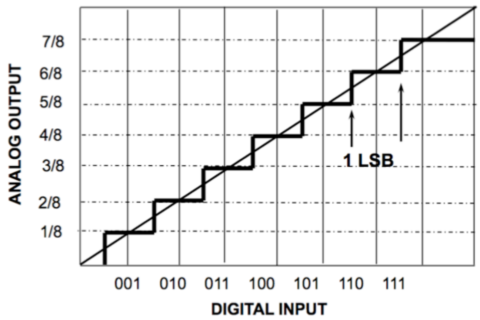
The basic measurement and control loop measures a process variable and determines what control action needs to be performed based on the processor's control algorithm. The measurement path takes an analogue variable (such as pressure, temperature etc), signal conditions it in the analogue domain before applying it to an analog-to-digital converter (ADC). The ADC provides a digital output corresponding to the value of the analogue input signal relative to the reference voltage. The resolution of the converter determines the number of bits (n) in the digital representation.

On the control side, the digital word from the processor is converted to an analogue value using a digital-to-analog converter (DAC). The number of digital bits which are convereted to an analogue value is determined by the resolution of the DAC.

Sampling and Quantization

- **Sampling Process**: convert a continuous time domain signal at discrete and uniform time intervals.
- Determines maximum bandwidth of sampled (ADC) or reconstructed (DAC) signal (**Nyquist Sampling Theorem**: sampling frequency must be at least twice that of maximum signal frequency)
- Frequency Domain- "Aliasing" for an ADC and "Images" for a DAC
- **Quantization Process**: convert an analogue signal with infinite resolution with a digital word having finite resolution and an analogue output which only exists in discrete levels
  - Determines Maximum Achievable Dynamic Range
  - Results in Quantization Error/Noise

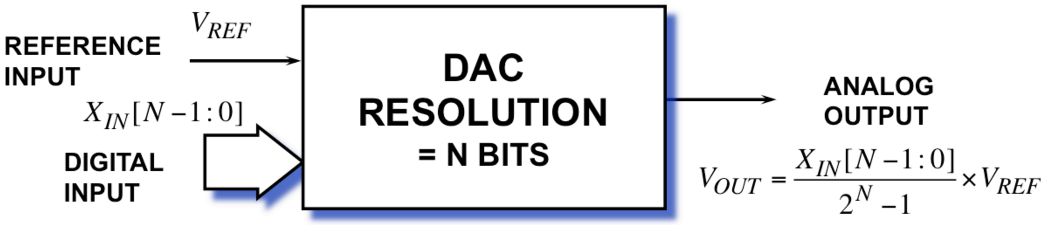




The **sampling process** is the representation of a continuous time domain signal at discrete and uniform time intervals. The maximum amount of information content, or bandwidth, is determined by the **Nyquist sampling theory** which states that, in order to preserves the information of the signal being sampled, one must sample the signal at a rate is that 2X or higher than the maximum frequency component in the signal. What this implies is that all analogue signals MUST be first filtered to remove all components above ½ of the sampling frequency (known as the Nyquist frequency). If any signal components above the Nyquist frequency remain, the original continuous time signal will be corrupted when converted into discrete time – a phenomenon known as **aliasing**. We will consider this in details in your second year course as part of the EE topics.

The quantization process is the representation of the magnitude, in digital form, of the continuous analogue signal. The number of bits in the quantization process determines the number of discrete levels and, therefore, it determines the smallest resolvable signal. Note that in the diagram above, an analoueg output in the range can only exist at the discrete levels shown.

A simplified view of a DAC



- A digital-to-analog converter (DAC) produces a quantized (discrete step) analogue output (voltage or current) in response to binary digital input.
- A reference quantity (either voltage or current) is accurately divided into binary and/or linear segments.
- The digital input drives transistor switches that connect an appropriate number of segments to the output.
- 2^N discrete values resulting in quantization errors.
- Sampling and quantization impose fundamental but predictable limitations in the system.
- The microcontroller on the Pyboard has 2 x 12 bit DAC converters.

A **digital-to-analog converter** produces an analogue output which corresponds to the value of the digital input signal. The analogue output value corresponds to the relative value of the digital input signal with respect to a fixed reference value V_{REF} and, in its most basic form, is determined by the relationship above.

A key element of the transformation from the digital domain to the analogue domain is that a series of finite discrete values is now represented by an analogue variable.

Resolution in various forms

- **Resolution** of a ADC or DAC is dependent on the number of input data bits.
- This defines the quantization step, which is expressed as LSB voltage (least significant bit).
- Resolution can also be expressed as %, parts per million (ppm), or dB relative to full scale (FS).

Resolution, Bits (n)	2^n	LSB, mV (10V FS)	% Full Scale	ppm Full Scale	dB Full Scale
8	256	39.1	0.391	3906	-48.0
10	1024	9.77	0.098	977	-60.0
12	4096	2.44	0.024	244	-72.0
14	16,384	0.610	0.006	61	-84.0
16	65,536	0.153	0.0015	15	-96.0
18	262,164	0.038	0.00038	3.8	-108.0
20	1,048,576	0.0095	0.00010	1.0	-120.0

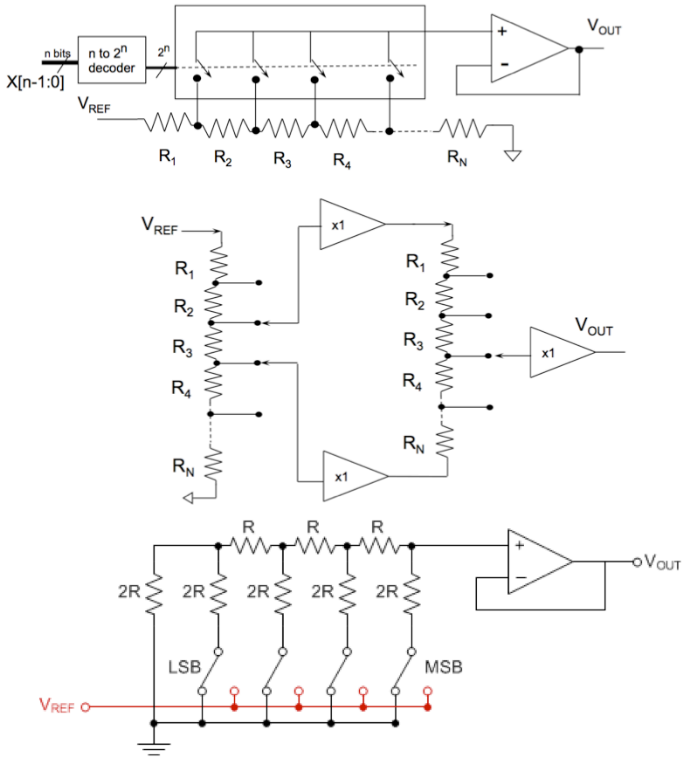
This table shows the relationship between the analogue output range, or full scale, and the LSB size for different resolutions. The full scale in this case is 10V.

LSB size can be expressed in voltage, percentage of full scale, parts per million (ppm) of full scale or ratio to full scale in dB.

For example, if you need a resolution of ±5% (i.e. step of 10%), and this corresponds to 1 LSB of the DAC. From the table, it is clear that a 10-bit DAC should be sufficient.

Common DAC architectures

- **Simple voltage divider DAC:** dividing reference voltage into equal steps.
- Output is guaranteed to go up if input goes up – i.e. monotonic.
- Number of taps increases exponentially (2^n) with number of bits.
- **Voltage segment DAC**: use two sets of voltage dividers, one for coarse steps and another for fine steps.
- Number of resistors reduced from 2^n to $2 \times 2^{n/2}$ for two segments.
- Could use three or more segments.
- **R-2R Ladder DAC**: same network as used in Lab 2, Task 3.
- Each stage has a voltage (or current) half that of the previous stage.
- Complexity is **O**rder(n) or **O**(n), where n is the number of input bits.



The simplest DAC circuit is that of a **voltage or potential divider**. Here all resistors are identical and V_{REF} is simply divided into N equal steps. This architecture is also known as **a String DAC** (having a string of identical resistors). The complexity grows exponentially with number of input data bits. We call this of Order(2^n).

This is also the architecture used in digital potentiometers. In an ideal potentiometer, all 0s and all 1s codes should connect the variable tap to one or other end of the string of resistors. So a digital potentiometer, while basically the same as a general purpose string DAC, has one fewer resistor and neither end of the string has any other internal connection.

An improved version of the DAC architecture is one that divides the voltage into coarse segment of equal voltage steps, and then each coarse segment into fine steps of equal size as shown here. This is known as a **Segment DAC**, and its complexity is Order ($2^{n/2}$).

Finally, if you use the circuit you tested in Lab 2, Task 3, you will discover that by connect a network with R and 2R resistors as shown above, you divide an input voltage (or current) by a factor of 2 each time. This is known as the **R-2R Ladder DAC**. It is highly efficient in resistor count, and has a complexity that increases linearly with the number of input data bits. We say that the complexity is of Order(n), where n is the number of bits.

A simplified view of a ADC

$$Y[N-1:0] = \frac{V_{IN}}{V_{REF}} \times (2^N - 1)$$

- An ADC produces a digital output corresponding to the value of the signal applied to its input relative to a reference voltage.
- There are 2^N discrete values, therefore introducing error known as quantization noise.
- Similar effect in sampling and quantization as found in DAC.
- The microcontroller on the Pyboard has 3 x 12-bit ADC, capable of taking 2.4 Msamples per second.

An analogue-to-digital converter produces a digital output which corresponds to the value of the analogue input signal. The digital output value corresponds to the relative value of the analogue input signal with respect to a fixed reference voltage and, in its most basic form, is determined by the relationship above.

A key element of the transformation from the analogue domain to the digital domain is that an analogue variable of infinite resolution is now represented by finite discrete values. The analogue input is quantized into 2^N discrete levels, where N is the resolution of the converter. This results in a quantization error or uncertainity from the A/D conversion process.

The quantization and sampling of the input signal thus impose fundamental limitations on the A/D conversion process. However, these limitations are predictable. Let's look first at the quantization process.

The A/D conversion process also changes a continuous analogue signal in the time domain to a digital signal which is represented by values which occur at discrete intervals. The continuous analogue signal is sampled and converted to a digital word at these discrete time intervals.

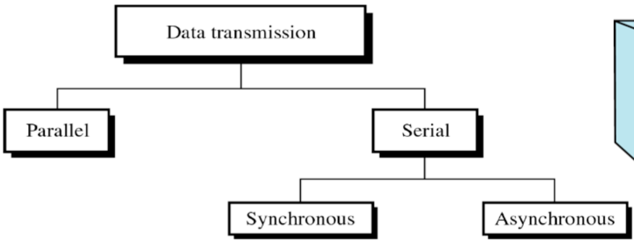
Video on DAC converter

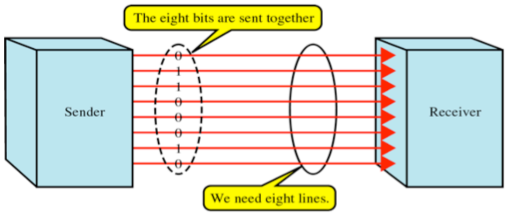
This is a Youtube video about DAC:

https://www.youtube.com/watch?v=b-vUg7h0lpE

Parallel Data Links

- The transmission of binary data across a link can be accomplished either in parallel mode or serial mode.
- In parallel mode, multiple bits are sent with each clock period.
- In serial mode, one bit is sent with each clock period.
- There are 2 subclasses of serial transmission: synchronous and asynchronous.





- Binary data may be organized into groups of n bits each.
- By grouping, we can send data n bits at a time instead of one. This is called parallel transmission.
- The advantage of parallel transmission is speed but its disadvantage is cost in interconnect resources.

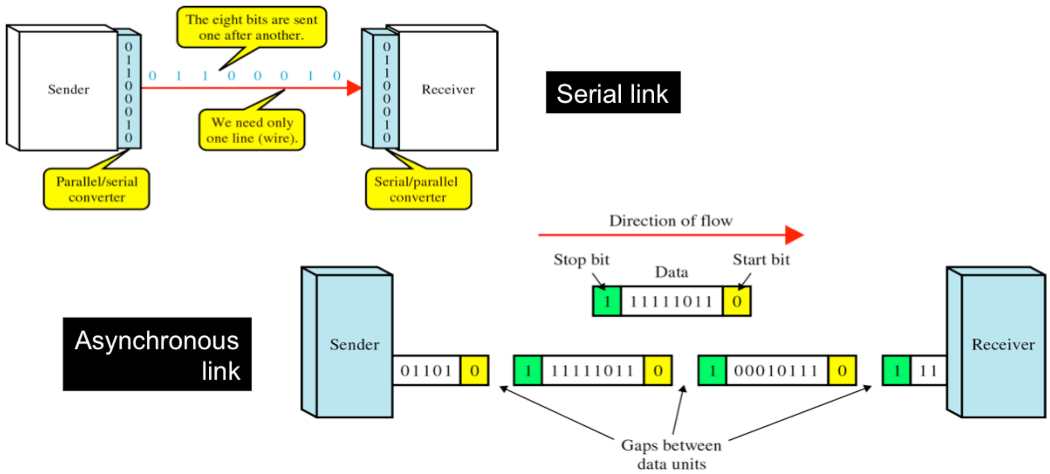
PYKC 16 June 2020 DE 1.3 - Electronics 1 Topic 18 Slide 9

Transferring data between two devices or modules can be done in two main methods: parallel and serial. In parallel transmission, a data word (which has n bits) are send together all at once. The advantage is speed of transfer. Example of this is the memory interface with microprocessor. The disadvantage is that you need n wires, which could be costly. For example, pins on the packaging of a chip is expensive.

The alternative is serial communication, where data is sent one bit at a time.

Serial Data Links

- In serial transmission, one bit follows another, so we need only one communicating channel (or wire) to transmit data between 2 communicating modules.
- The advantage of serial transmission is the reduction of interconnect resources, but it takes N times longer to send the information, where N is the number of bits of data.
- Serial transmission occurs in one of 2 ways: **asynchronous** or **synchronous**.



PYKC 16 June 2020 DE 1.3 - Electronics 1 Topic 18 Slide 10

In serial transmission, one bit follows another, so we need only one communicating channel (or wire) to transmit data between 2 communicating modules. The advantage of serial transmission is the reduction of interconnect resources, but it takes N times longer to send the information, where N is the number of bits of data.

Serial transmission occurs in one of 2 ways: asynchronous or synchronous.

In **asynchronous link**, the timing of a signal is unimportant. Information is received and translated by agreed-upon patterns. Patterns are based on grouping the bit stream into bytes. The sending system handles each group independently, relaying it to the link whenever ready, without regard to a clock signal.

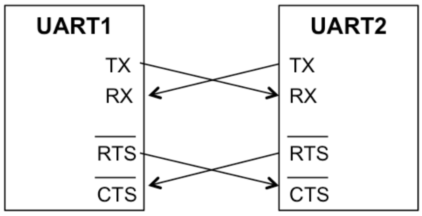
To alert the receiver to the arrival of a new group, an extra bit called **start bit** is added to the beginning of each byte. To let the receiver know that the byte is finished, one or more additional bits called **stop bits** are appended to the end of the byte.

This mechanism is called **asynchronous** because at the byte level, sender and receiver do not have to be synchronized. But within each byte, the receiver must still be synchronized with the incoming bit stream.

The bit rate (i.e. bits per second) in a UART is also known as baud rate. For example, in the Lab, we used 9600 bauds per second. This is equivalent to sending around 960 bytes per second, each having at least 10 bits (8 bits data, 1 start bit and 1 stop bit).

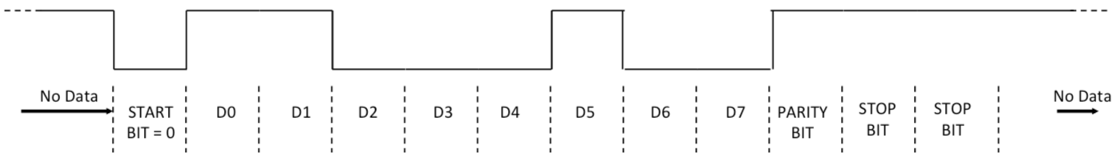
UART and flow control

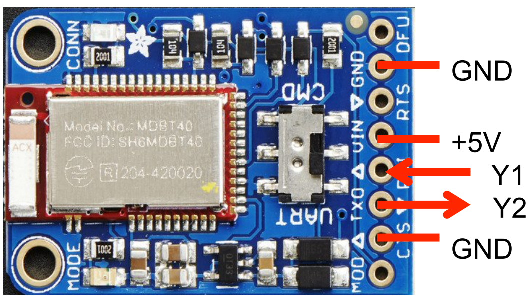
- UART is a module found in many microcontroller that uses an asynchronous serial method for transferring data.
- The ESP32 uses UART to talk to your laptop via the USB cable.
- The UART signal waveform is shown below. (You have also seen this in Lab 1).
- Physical connections between two UARTs consist of four signals:
  - RX – data receive signal (input)
  - TX – data transmit signal (output)
  - CTS – clear to send signal (input)
  - RTS – ready to send signal (output)
- Both devices can simultaneously send and receive data. This is known as a full duplex link.
- RTS and CTS are active-low signals to indicate when a device is ready to transfer data.
- By connecting, say, the UART on the ESP32 to the laptop, one can control the flow of data between the two. This is called **flow control** mechanism.

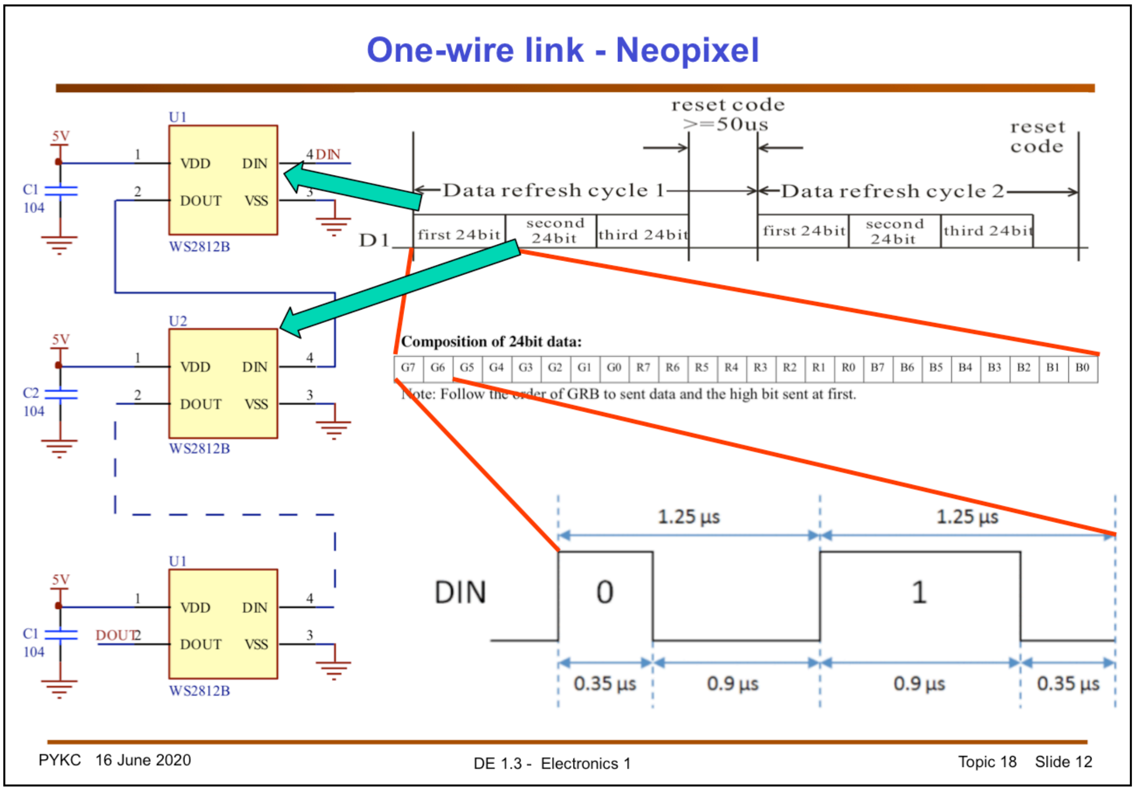


We have already examined the UART data format in the Lab 1. I will not repeat it here. Instead I want to point out that physically, an UART has two other signals, RTS and CTS, which are both active (i.e. true = low voltage level). They are used to signal to the other device when it is ready to receive data. So, if UART_1 RTS signal is high, UART_2 will not send any data. When RTS is low (active), UART_2 will start sending data. In this way the flow of data is being controlled. Hence this is known as **flow control** between the two UARTs.

Shown here is a UART module from Adafruit that last year DE1 used. You will be using this module next year.







Neopixel is another example of asynchronise link that uses only one data wire. Neopixel is actually a fancy name for the WS2812B three colour LED. The Neopixel strip you received in the Home Lab Kit has 8 LEDs connected in series (called daisy-chain) as shown in the slide. Each LED module contains three separate LEDs: R, G and B.

To specify the intensity of each LED, you send it a 24-bit data packages in sequence as shown in the top timing diagram. First 24-bit is for the LED module closest to the input DIN. Second 24-bit packet is for the 2nd LED etc. The timing shows three such packets for three LEDs. For our Neopixel strip, we will be send eight 24-bit sequence together.

The second timing diagram show the 24-bit data format. They are in sequence of Green-Red-Blue. Each colour intensity is specified with an 8-bit number. The most significant bit is sent first.

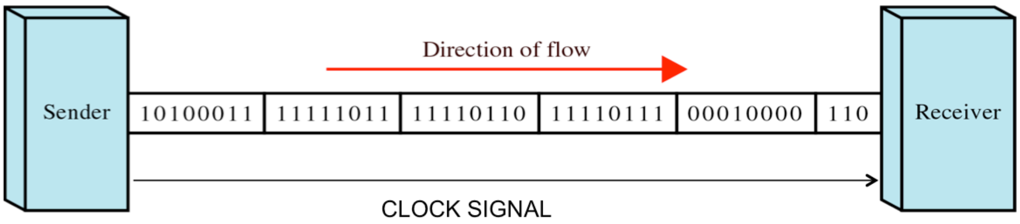
Each individual bit is sent not as a straight forward binary 1 and 0. Instead a 0 bit is sent as a pulse with a period of 1.25us (or 800kHz) and a pulse with of 0.35us. A 1 bit is sent with the same period but the pulse width is now 0.9us.

The first pixel is strip off the first 24-bit data, and pass through to the second pixel the remaining data etc.

See the datasheet for WS2812B on the course webpage.

Synchronous Serial Link

- In a synchronous link, the bit stream is combined into longer frames which may contains multiple bytes.
- Each byte is introduced onto the transmission link without a gap between it and the next one.
- It is the responsibility of the receiver to reconstruct the information, usually under the control of a clock signal.
- Without gaps and start/stop bits, timing becomes very important therefore the accuracy of the received information is completely dependent on the ability of the receiver to keep an accurate count of the bits as they come in

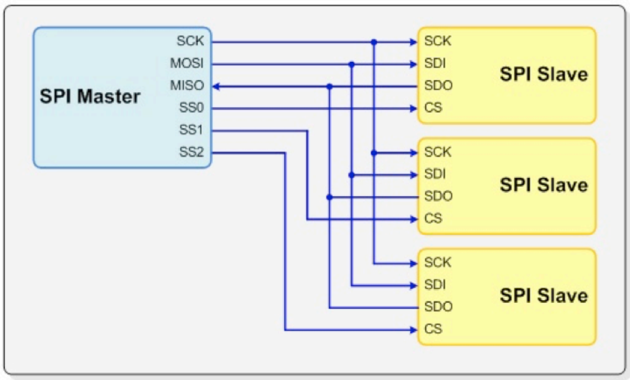


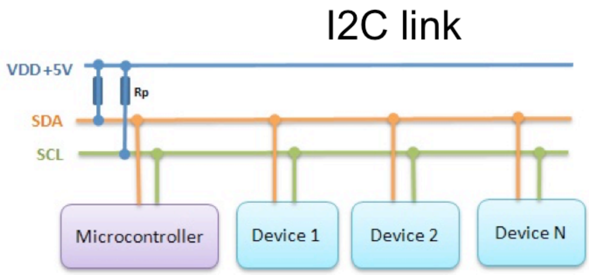
PYKC 16 June 2020 DE 1.3 - Electronics 1 Topic 18 Slide 13

Asynchronous serial transmission is only suitable for low data rate link between modules. For higher speed communication between two modules on the same chip or on the same PCB, a synchronous serial link is much better. The transfer of data in a synchronous serial link requires the use of a clock signal from one module to another module. The module that produces the clock signal is known as the **master** device. The other module is known as the **slave device**.

I2C and SPI synchronous serial links

- I2C stands for Inter-Integrated Circuit serial protocol. Also known as I^2C or Two Wire Interface (TWI).
- Originally from Philips Semiconductor, and it is now an industrial standard.
- Allows up to 127 devices to be connected, each having a unique address.
- Up to 400kHz data rate.
- SPI stands for Serial Peripheral Interface Bus.
- Both of these are common synchronous serial links for connecting to other chips in the systems, such as ADC, DAC and other sensors.





Two of the most common synchronous serial links used in industry are: I2C (Inter-Integrated Circuit) and SPI (serial peripheral interface).

I2C was proposed by Philips Semiconductor and uses only two wires: SCL for the clock signal and SDA for the data signal. The data wire is bi-directional (i.e. data could be programmed to go in or out of a device). The protocol allows up to 127 devices to be connected on the I2C interface bus as shown above.

The SPI interface also has a clock signal SCLK. However this standard uses separate serial data input and serial data output pins, thus allowing simultaneous communication in both directions. It further requires the master to produce a separate chip select (CS) signals for each SPI slave device.

SPI has higher data rate than I2C and in general is easier to use. I2C is more efficient on pins because it uses only two per device.

The Pyboard provides 2 of each through its ST Micro controller chip.

Comparison of the three serial links

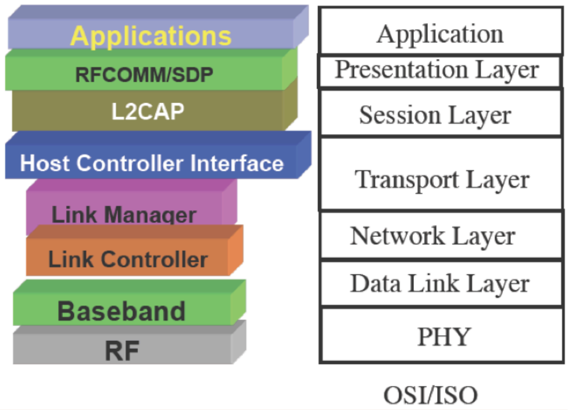
- The Pyboard has 4 UARTs, 2 each of SPI and I2C links.
- Micropython provides library functions to drive these interfaces.
- We will only use UART on this course.

	Synro nous		Asynchronous
	SPI	I2C	UART
Max Bit Rate	10Mbit/s	1Mbit/s	500kbit/s
Max devices	Limited by pins	127 devices	point to point
No. of pins	3 + n	2	2
Pros	Simple, low cost high speed	Low pin count; allows multiple masters	Long distance, good noise tolerance
Cons	Single master, short distance	Slow, short distance	Only point-to-point, slow
Applications	Connect to peripherals on PCB	Share bus connection with periperals on same PCB	Connect computers to terminals and other slower systems

This table summaries the key characteristics of the three serial links found on the Pyboard. For our project, we only use the UART to link to the Bluetooth module.

Bluetooth wireless links

- Bluetooth is a cable replacement technology.
  - 1Mb/s, range ~10m, much lower power than wifi and cheaper
- We use **Bluetooth Low Energy**, a version of Bluetooth that uses much less power, but also has much lower data rate (realistically less than 100kbit/s)
- Bluetooth standard defines a **protocol stack** to enable different types of devices to communicate.
- The Bluetooth stack includes protocols for the radio layer all the way up to device discovery, service discovery, etc.



PYKC 16 June 2020 DE 1.3 - Electronics 1 Topic 18 Slide 16

So far we have only looked at wired links. Modern electronic equipment often communicate with each other through wireless links. Among wireless links, Bluetooth is one of the most popular. You will find Bluetooth capability on almost all smart phones, tablets and laptops.

I will not go into details about Bluetooth protocol because it is actually very complex. However, it is worth you to know that regular Bluetooth has a maximum data rate of around 1Mbits/sec. The range is around 10m, which is much shorter than, say, wifi and cellular data network.

Bluetooth standard is defined in various level of abstraction, from the level (also called layer) concerning detail signal characteristic, known as the physical layer (PHY) to the software interface that a user uses (known as application layer). Together these different layers from hardware to software is known as the **Bluetooth Stack**.

If you want to use Bluetooth on the Pyboard via the BlueFruit module, the bluetooth stack is already implemented on the module itself, making its use very simple. Similarly using the Bluetooth link on your phone is also made simple by the manufacturer such as Apple or Android, through their implementations of the Bluetooth Stack. So all you see is the application level interface with the Bluetooth link.

A video on Bluetooth Low Energy (5 min)

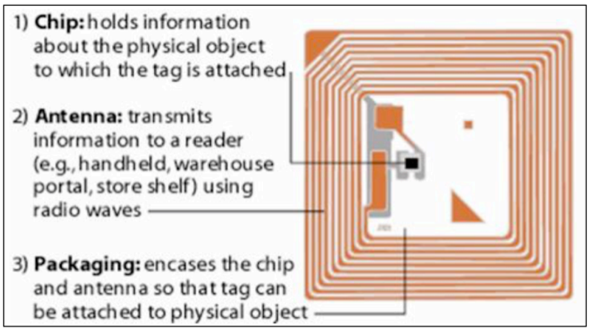
PYKC 16 June 2020 DE 1.3 - Electronics 1 Topic 18 Slide 17

This video can be found on:

https://www.youtube.com/watch?v=ltV08vGqACM

What is RFID?

- Radio Frequency Identification (RFID) is a technology that employs a microchip with an antenna that broadcasts its unique identifier and location to receivers.
- It employs a microchip called a smart tag, broadcasts unique 96-bit identifier to receiver.
- The receiver relays the data to a computer.
- A RFID Tag contains two main parts:
  - Silicon chips
  - Antennas
- These components enable tags to receive and respond to radio frequencies queries from RFID transceivers.
- Two types of RFIDs:
  - Passive
  - Active



PYKC 16 June 2020 DE 1.3 - Electronics 1 Topic 18 Slide 18

A **Radio-Frequency IDentification** system has three parts:

- A scanning antenna
- A transceiver with a decoder to interpret the data
- A transponder - the RFID tag - that has been programmed with information.

The scanning antenna puts out radio-frequency signals in a relatively short range of a few feet. The RF signals provides a means of communicating with the transponder (the RFID tag) AND It provides the RFID tag with the energy to communicate. (This is only true for passive RFID tags such as your Oyster card or wireless credit card.)
Passive RFID tags do not need to contain batteries, and can therefore remain usable for very long periods of time.
The scanning antennas can be fixed or mobile. They can take on many forms; for example, you could build them into a door frame to accept data from persons or objects passing through.
When an RFID tag passes through the field of the scanning antenna, it detects the activation radio frequency signal (RF) from the antenna. This “wakes up” the RFID chip, and it transmits the information on its microchip to be picked up by the scanning antenna.

RFID Tags types

- **Passive**
  - Have no internal power supply
  - Electrical current induced in antenna by the incoming signal provides power for integrated circuit in tag to power up and transmit response
  - Very Small, Limited Range, Unlimited Life
- **Active**
  - Have their own internal power source
  - Many operate at fixed intervals
  - Also called beacons (broadcast own signal)
  - Large (coin), Much larger memories, Longer range

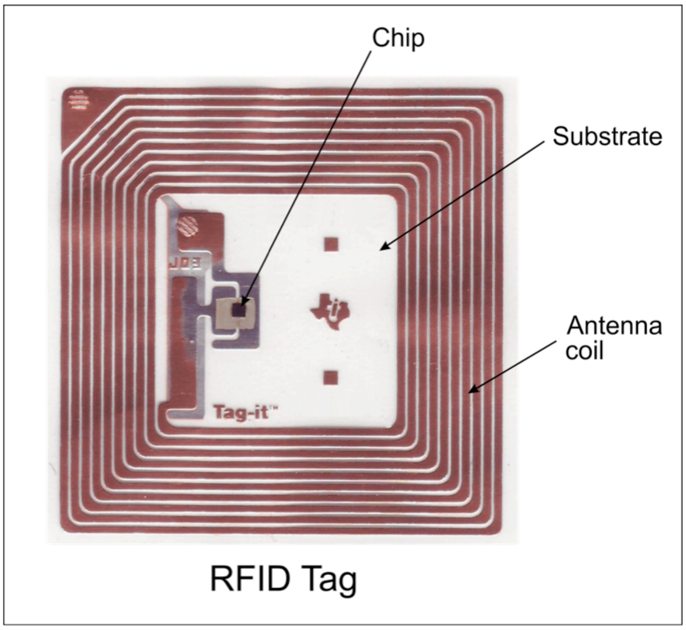
Four main frequencies:

	Frequency	Distance	Example Application
LF	125khz	Few cm	Auto-Immobilizer
HF	13.56Mhz	1m	Building Access
UHF	900Mhz	~7m	Supply Chain
μwave	2.4Ghz	10m	Traffic Toll

PYKC 16 June 2020 DE 1.3 - Electronics 1 Topic 18 Slide 19

A RFID tag may be of one of two types. **Active** RFID tags have their own power source; the advantage of these tags is that the reader can be much farther away and still get the signal. Even though some of these devices are built to have up to a 10 year life span, they have limited life spans. However, the most common are the passive RDIF tags which do not require batteries, and can be much smaller and have a virtually unlimited life span.
RFID tags are much better than bar codes and are replacing them. Unlike bar codes, a tag need not be on the surface of the object (and is therefore not subject to wear). The read time is typically less than 100 milliseconds. Large numbers of tags can be read at once rather than item by item.

A video about RFID (4 min)



This video can be found on:

https://www.youtube.com/watch?v=VpEVkiMU18s