PHY405-L08

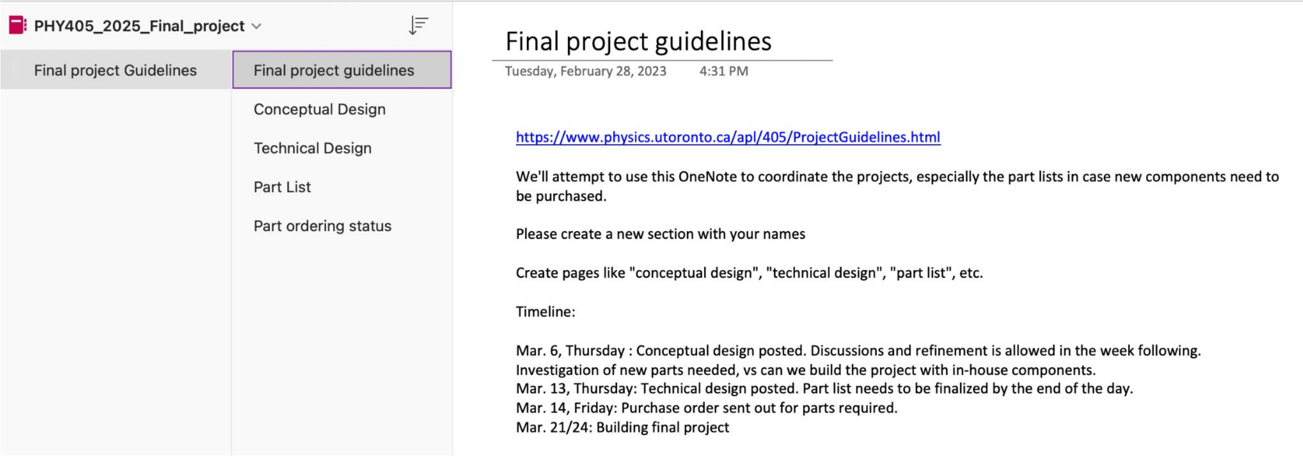
“Fast” pulses, “High” voltage, & Digitization

Logistics -- Final project

- 2 more regular labs to go, final project afterwards.
  - See the project guidelines for more information.
- Coordinate with this OneNote page
- We'll follow the steps of
  1. Conceptual design -- due **today**
     a. Please discuss and come up with a project
  2. Technical design -- due Mar 13th
     a. Try to use in-house components as much as possible
     b. Part list finalized by the end of this day
     c. Order will go out on Friday. Usually takes a week to get back
  3. Realization -- the two weeks following

Project coordination

- New section with your name
- Add pages of
  - Conceptual design
  - Technical design
  - Part list
  - Etc.
- I’ll collect part list by the end of Mar. 14



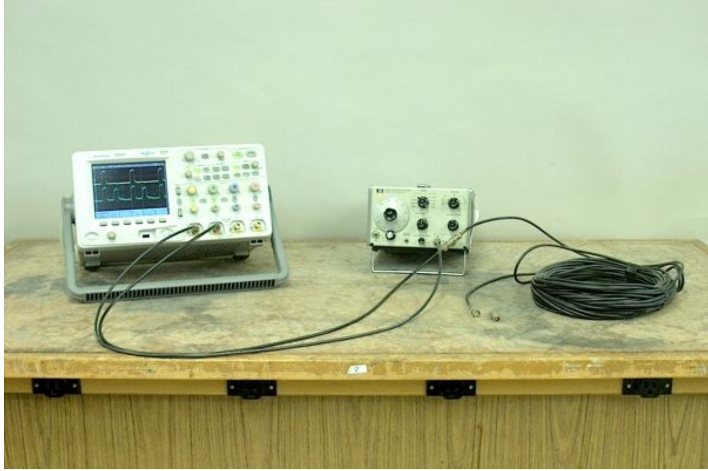
Lab 7 – Fast Pulses

- Labs 5 and 6 were small complete systems, and Lab 7 is the most complex yet.
- Build voltage multiplier to convert 5V amplitude AC into almost 30V DC.
- Use this voltage to power a red LED working as a Single Photon Avalanche Detector (SPAD).
- Feed the SPAD pulses into a discriminator circuit that produces a digital output pulse when the input is large enough.
- Count these digital pulses with your Arduino.

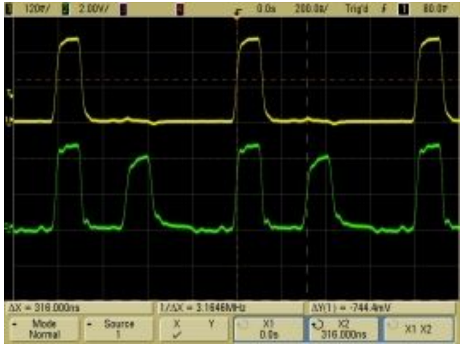
Fast Signals

- Many physics and communication signals are fast, i.e. very short or high frequency. Need to worry about
  - Capacitances
  - speed of light (foot/ns)
  - dispersion
  - digital processing speeds
  - …
- Lab 7 looks at short pulses produced by single photons interacting in an LED.
- This requires higher voltages, so we will also need to build a voltage multiplier.

Fast pulse in cables

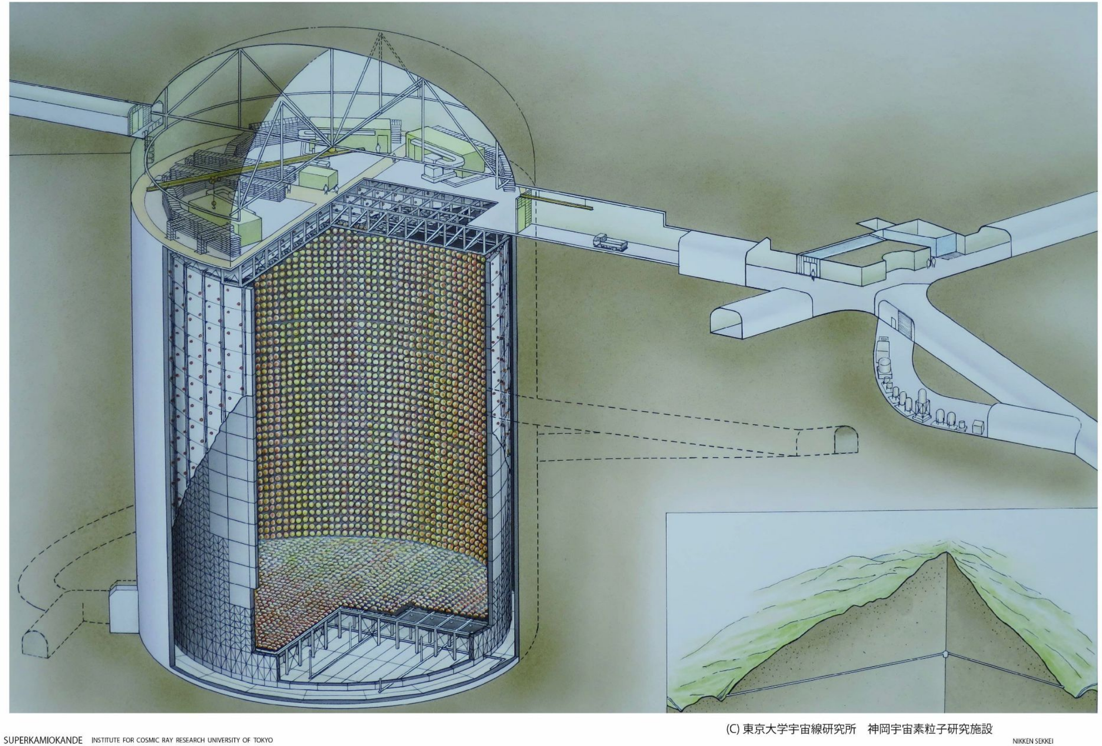


- High frequency signals behave more like EM waves
- They take time to travel through the circuit.
    - Speed of light ~ 1 ft/ns
    - Velocity factor in cable ~60-80%
- Cables have characteristic impedance
    - Signal will reflect if impedance mismatches
    - Short circuit reflects it flipped, open circuit reflects it in phase. Other impedances does something in between.

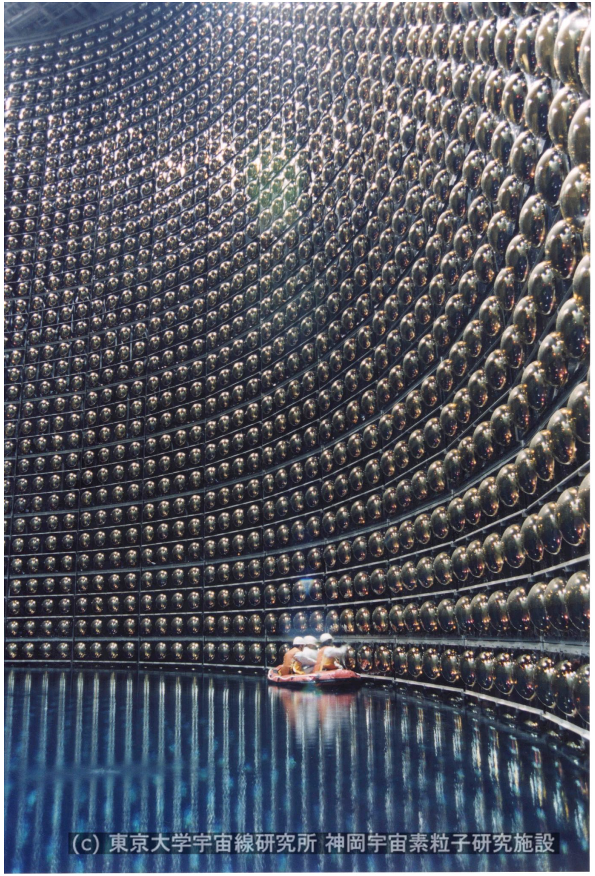


Source: UCSB physics website

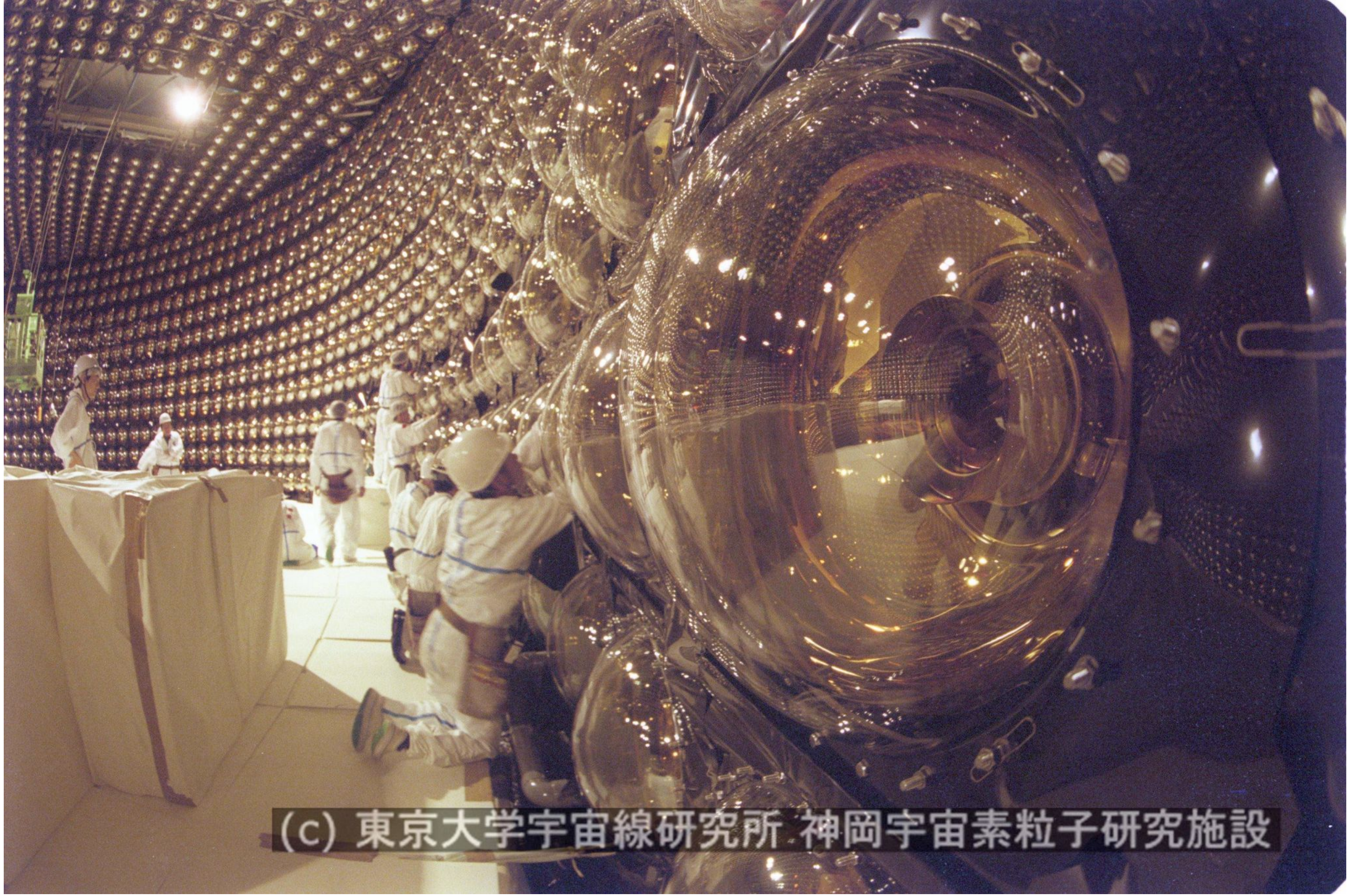
SuperKamiokande



Super-Kamiokande neutrino and proton decay detector
https://www-sk.icrr.u-tokyo.ac.jp/en/sk/

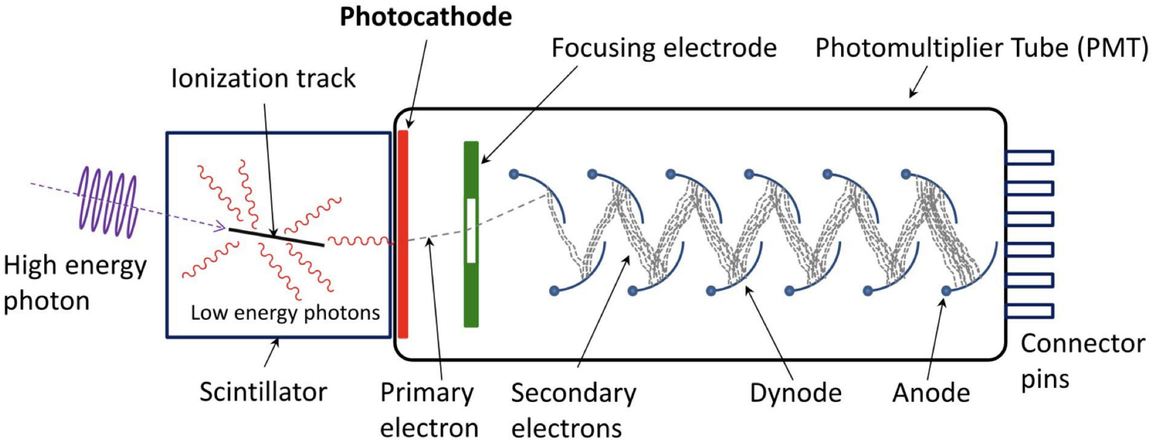


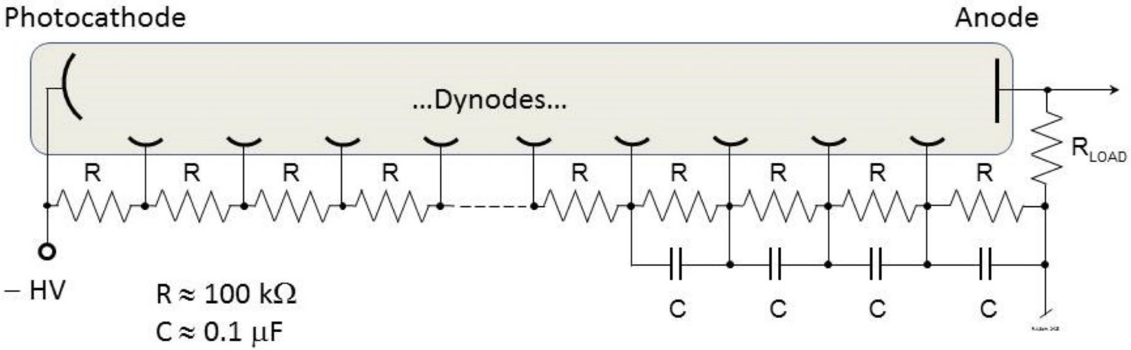
SuperK interior



SuperK phototube installation (from ICRR U. Tokyo)

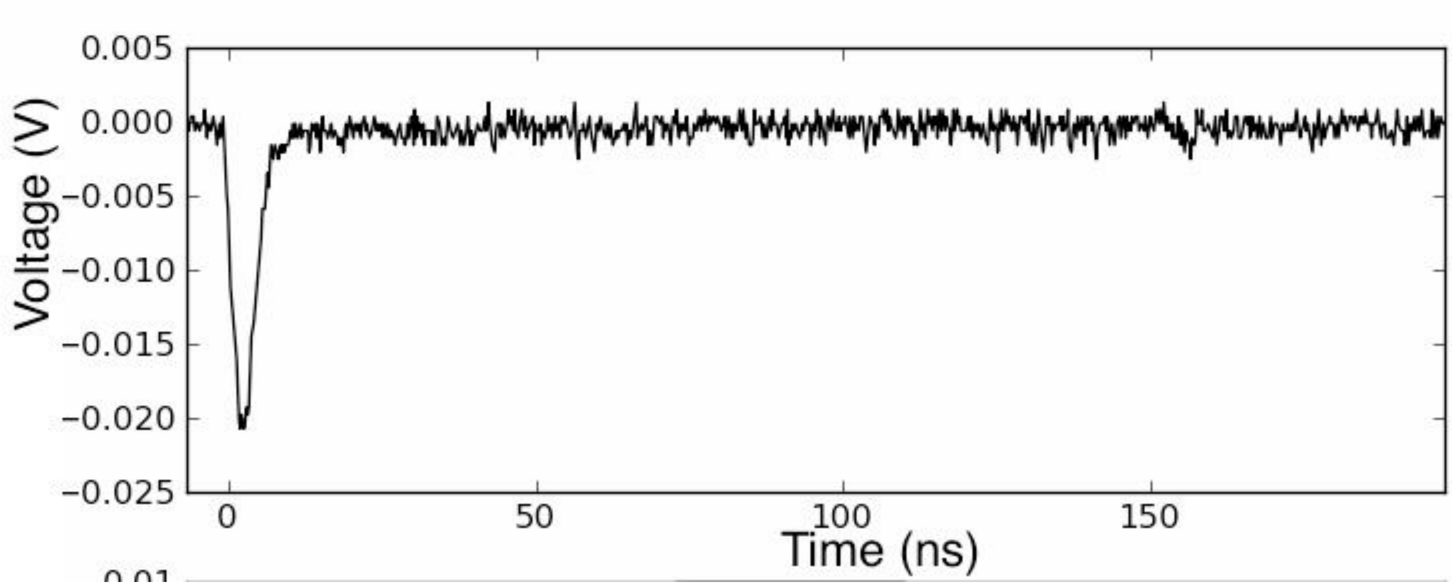
Photomultiplier Tube





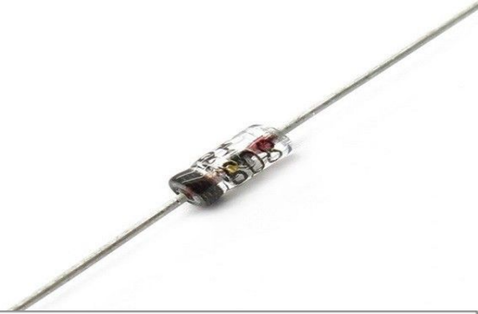
Photomultiplier tube structure and circuit (from wikipedia)

Photomultiplier Tube Pulse

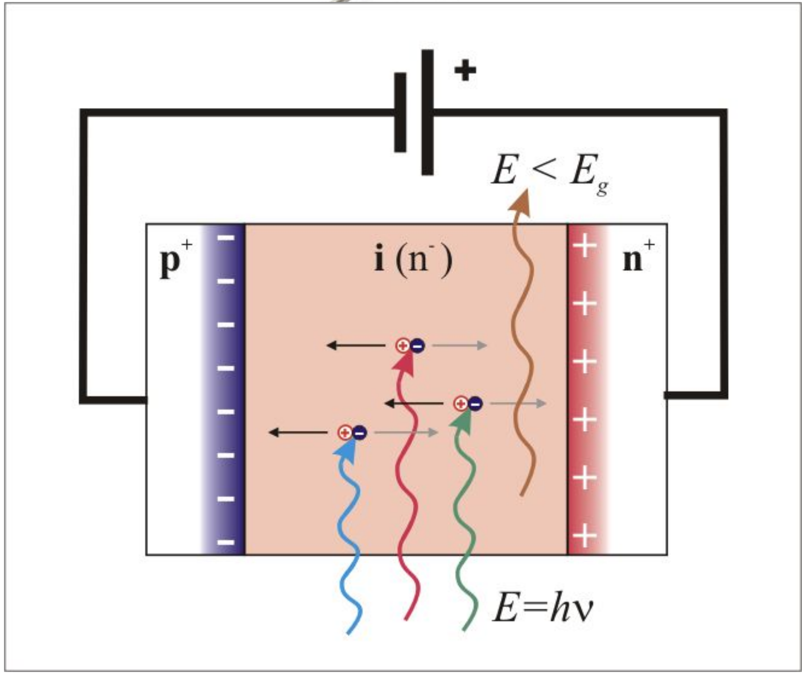


R11780 12 inch Hamamatsu1 photomuliplier tube pulse (from J. Brack et al., Nucl. Instrum. Methods Phys. Res. 712A (2013) 162-173)

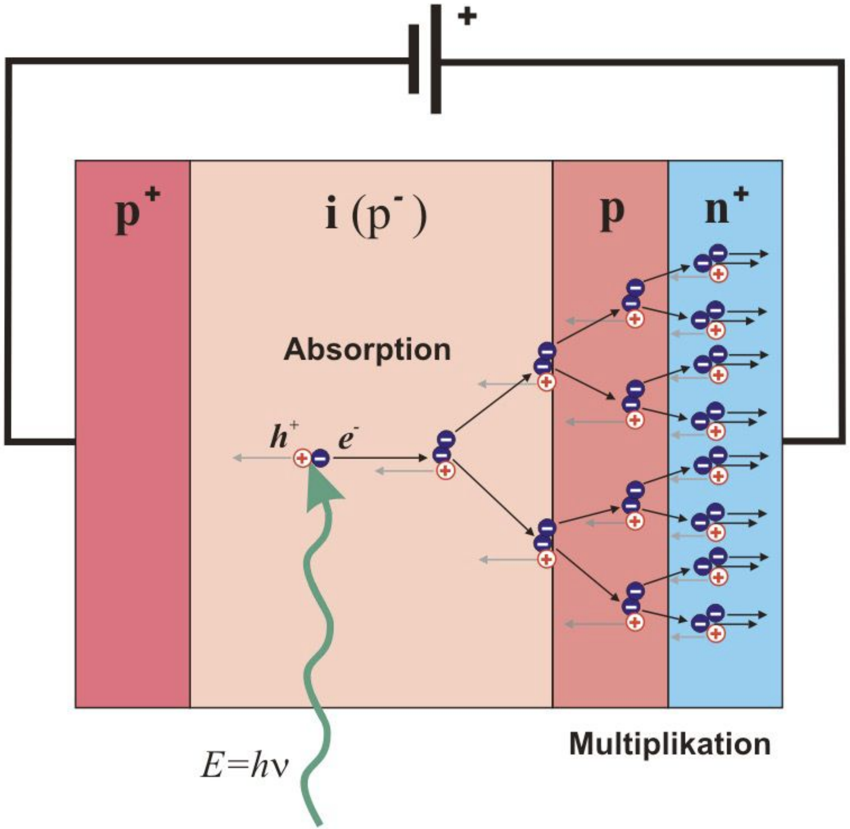
PIN Photodiode



- PIN (p-type – intrinsic – n type) photodiode.
- When light produces electron hole pairs in the intrinsic region, a current flows. (source: [Wikipedia]).
- Often used as a photon detector

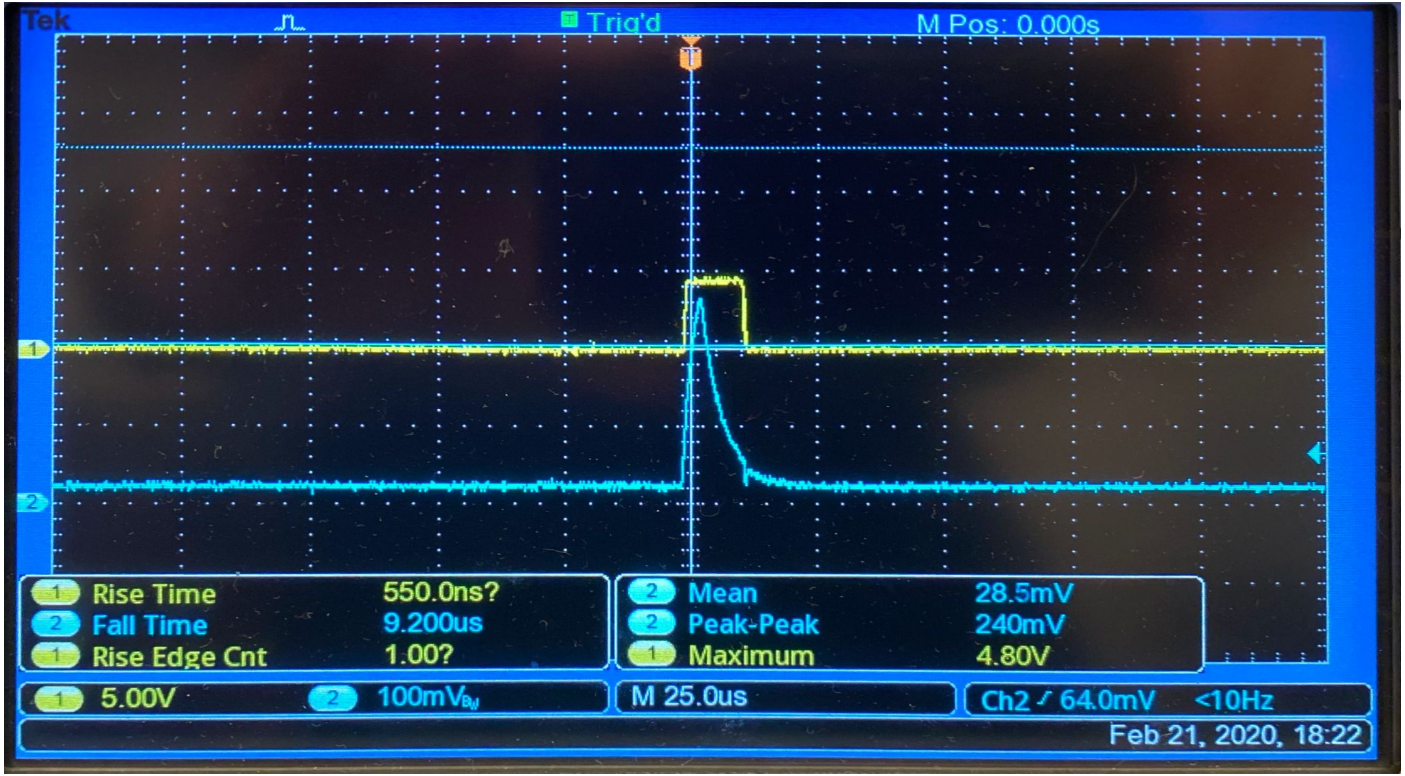


Avalanche Photodiodes (APD or SPAD)



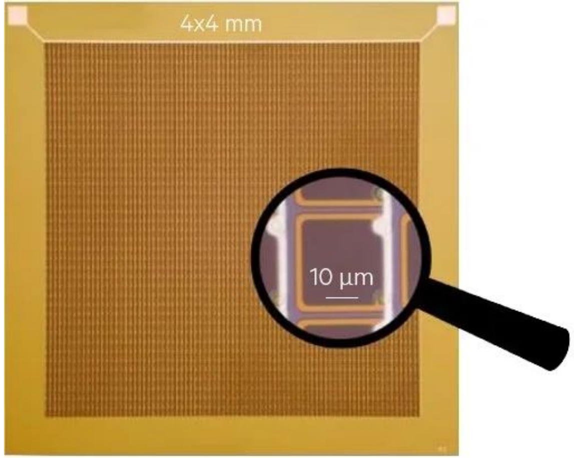
Photodiode Avalanche
(from wikipedia)

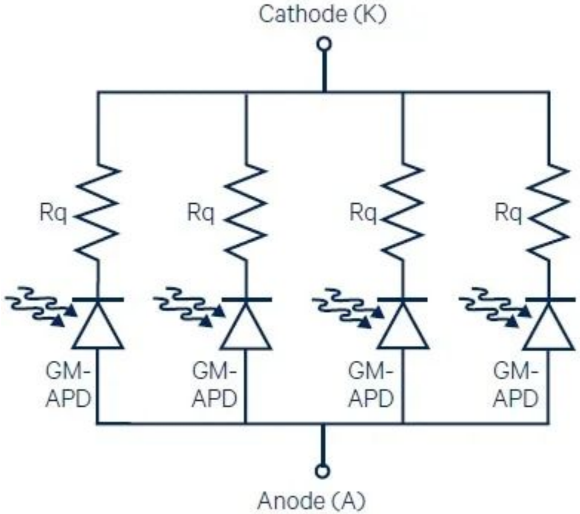
“SPAD” pulse from red LED

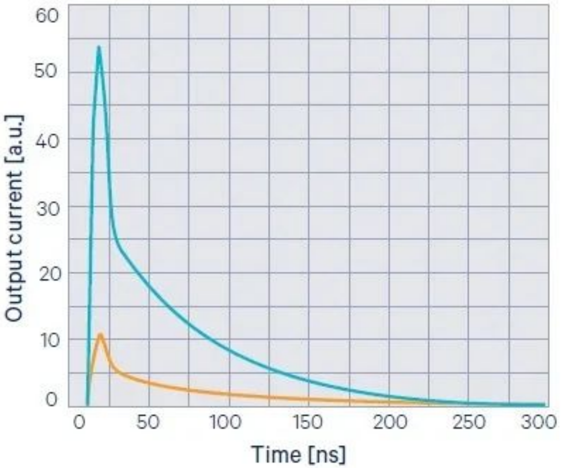


A single photon pulse from a reverse biased LLL-4223 2.5V / 10mA red LED.

Silicon Photomultipliers

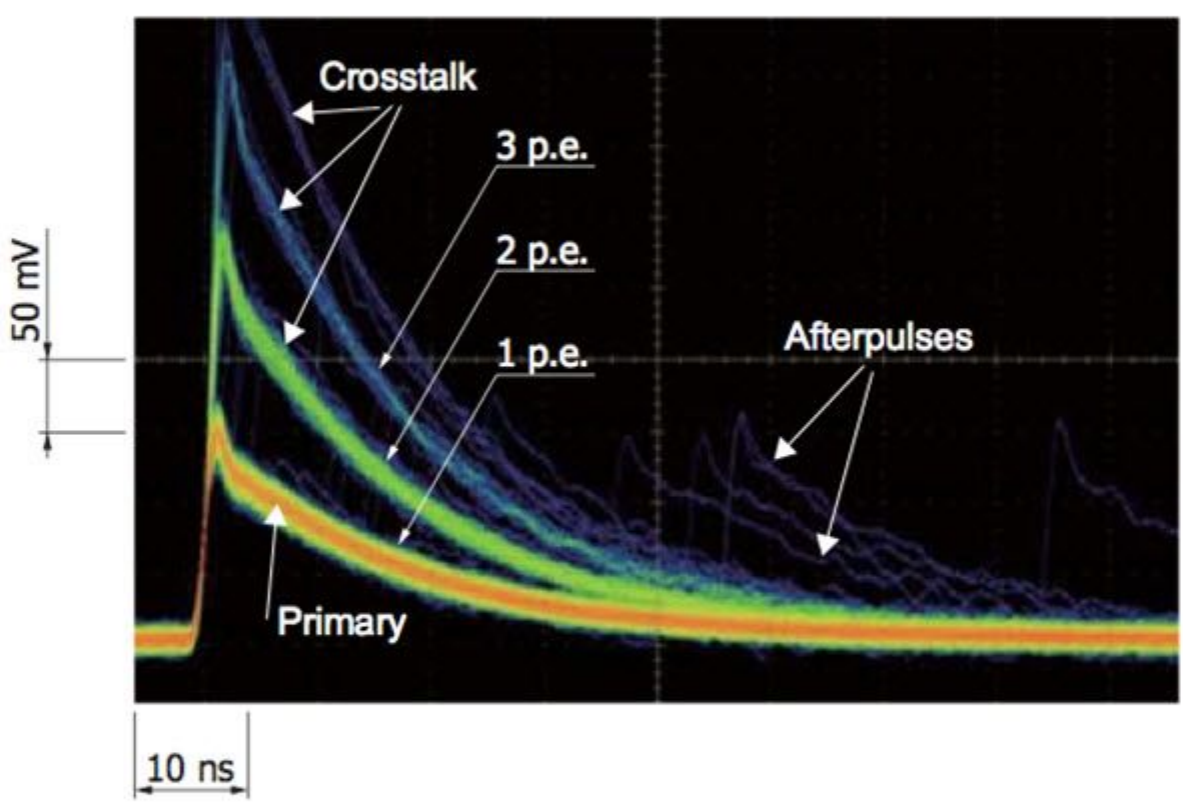






https://www.azosensors.com/article.aspx?ArticleID=865

SiPM Pulses

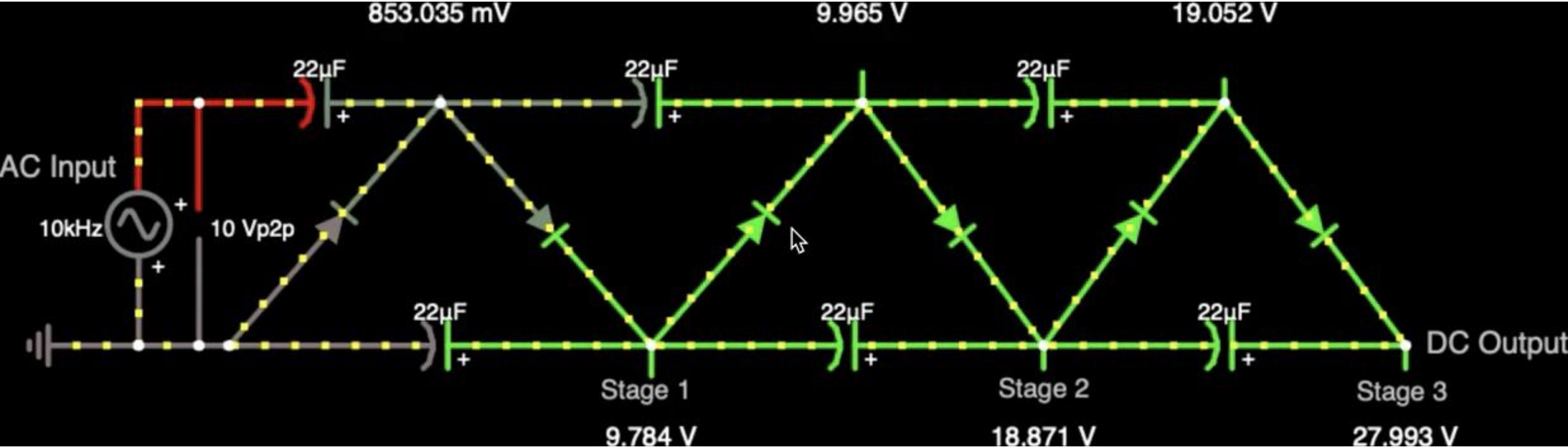


https://hub.hamamatsu.com/us/en/technical-notes/mppc-sipms/what-is-an-SiPM-and-how-does-it-work.html

Photon Detectors

- Always have a dark current - an output even when there are no photons - caused by thermal excitations, quantum tunnelling….
- Afterpulsing is when an initial real photon causes additional correlated output pulses from a detector
  - e.g. because photons are produced in an avalanche and these interact and trigger a new avalanche.
- Continuous breakdown occurs at high enough voltages.
  - Zener diodes are essentially fine-tuned avalanche diodes that break down at well-defined low voltages.

Cockcroft-Walton Voltage Multiplier



Key principles

- diodes only conduct one way
- capacitors block DC and pass AC
- capacitors hold their charge

600 kV Cockroft Walton@CERN vs. 870 kV @ PSI

Arduino ADC

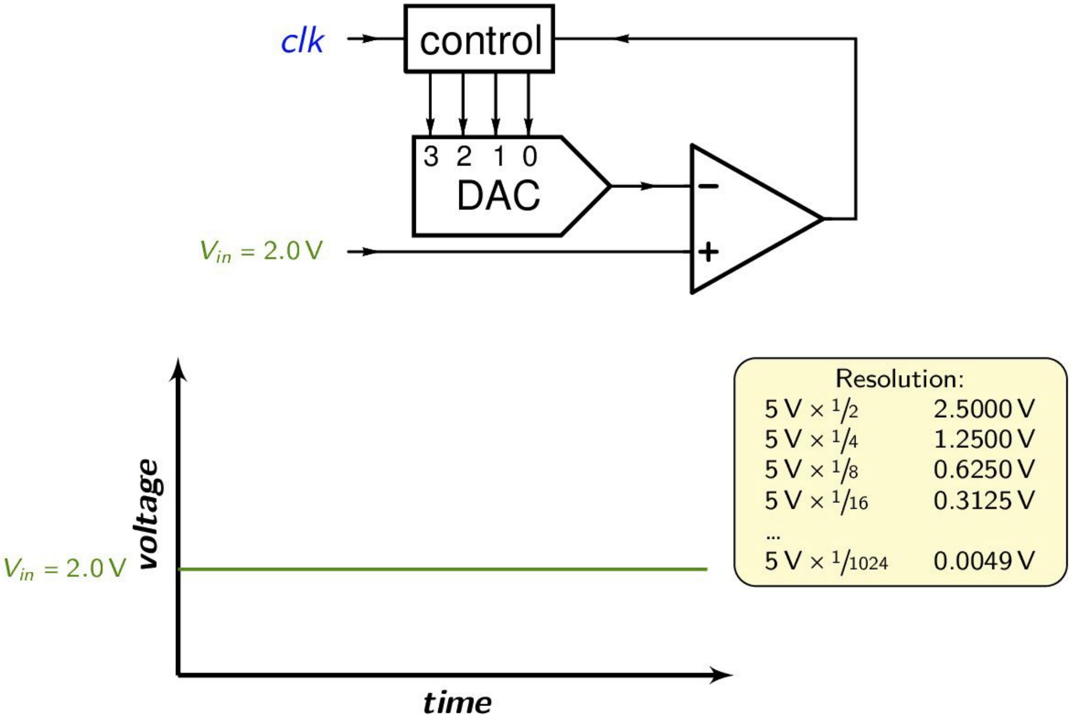
- Arduino Uno has one 10-bit Analog-to-Digital Converter (ADC) that is multiplexed to its 6 analog input pins (A0-A5).
  - It takes 13 ADC clock cycles to digitize an input voltage.
    - Except takes 25 cycles when ADC first initialized
  - Each ADC clock cycle is 128 Arduino clock cycles.
  - The Arduino Uno default clock speed is 16 MHz
- So ADC digitizes at 16MHZ/128/13 = 9615 Hz.

Arduino ADC

- Why so slow?
- Arduino ADC works by successive approximation.
    - Takes at least N_{bits} + 1 ADC clock cycles
    - ADC clock cycle == many processor cycles because of processing involved.
    - e.g. 4-bit example (from [wikipedia])
- Much deadtime at high rates
    - Dead time is the fraction (DT) of time that a detector is unavailable to record new data, because it is processing previous data or needs to reset a voltage or ….
    - If the true rate is R_{true}, and each detected event makes the detector unavailable for a time τ_{busy}, then in the simplest approximation the deadtime

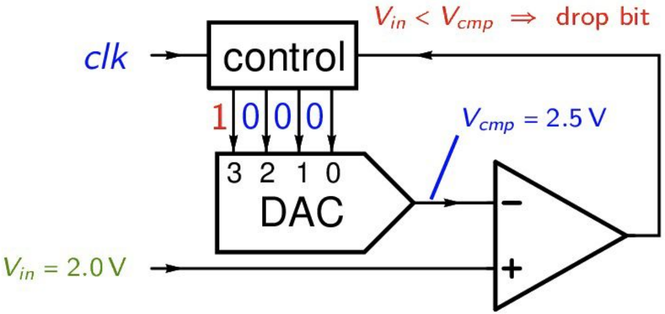
$$DT = \frac{R_{true}}{1+R_{true}\tau_{busy}}$$

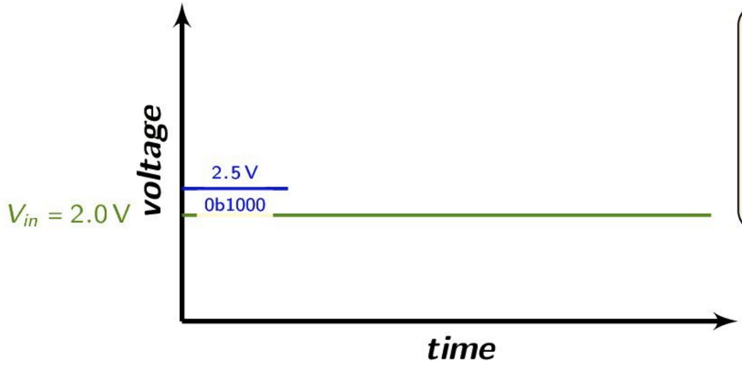
4-bit ADC Successive Approximation - Step 1



Step 1: 2.0 V input voltage on ADC input. (from wikipedia)

4-bit ADC Successive Approximation - Step 2

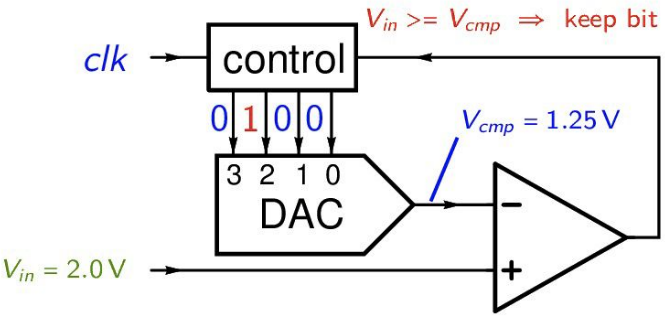


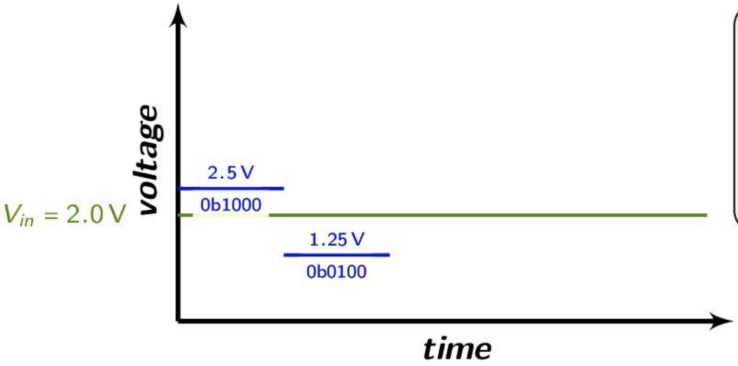


Resolution:	
$5\,V \times 1/2$	2.5000 V
$5\,V \times 1/4$	1.2500 V
$5\,V \times 1/8$	0.6250 V
$5\,V \times 1/16$	0.3125 V
...	
$5\,V \times 1/1024$	0.0049 V

Step 2: Control Register sets highest bit (3) on 5V Digital-to-Analog Converter (DAC),which outputs 2.5 V. Both Input and DAC voltages are input to a comparator whose output determines whether the Register keeps the bit set. In this case, no.

4-bit ADC Successive Approximation - Step 3



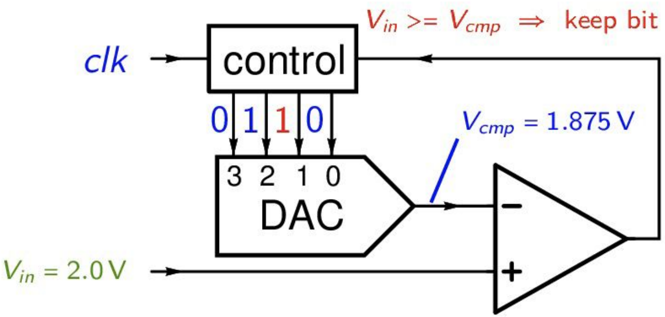


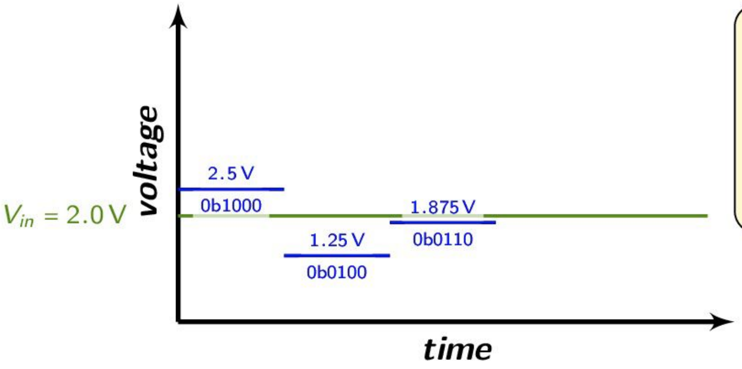
Resolution:

$5\,V \times 1/2$	2.5000 V
$5\,V \times 1/4$	1.2500 V
$5\,V \times 1/8$	0.6250 V
$5\,V \times 1/16$	0.3125 V
...	
$5\,V \times 1/1024$	0.0049 V

Step 3: DAC bit 3 is turned off and bit 2 turned on and DAC output compared. This time the bit is kept.

4-bit ADC Successive Approximation - Step 4



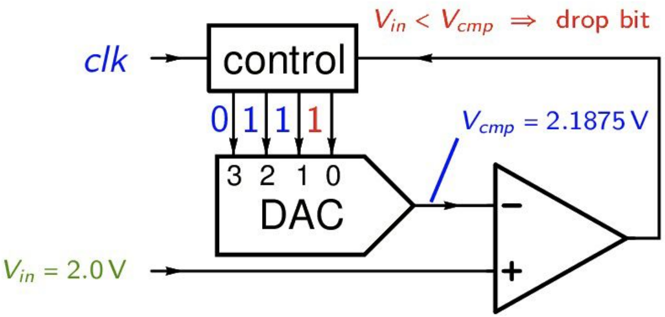


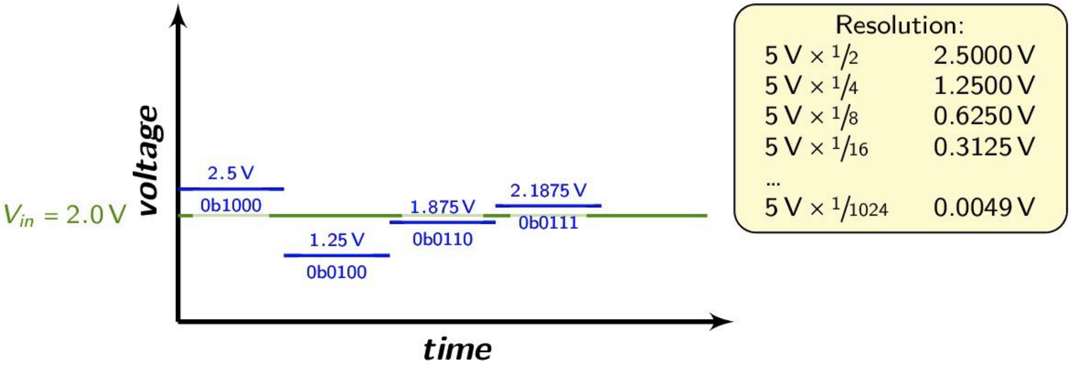
Resolution:

$5\,V \times 1/2$	2.5000 V
$5\,V \times 1/4$	1.2500 V
$5\,V \times 1/8$	0.6250 V
$5\,V \times 1/16$	0.3125 V
...	
$5\,V \times 1/1024$	0.0049 V

Step 4: DAC bit 2 kept; bit 1 turned on and compared.

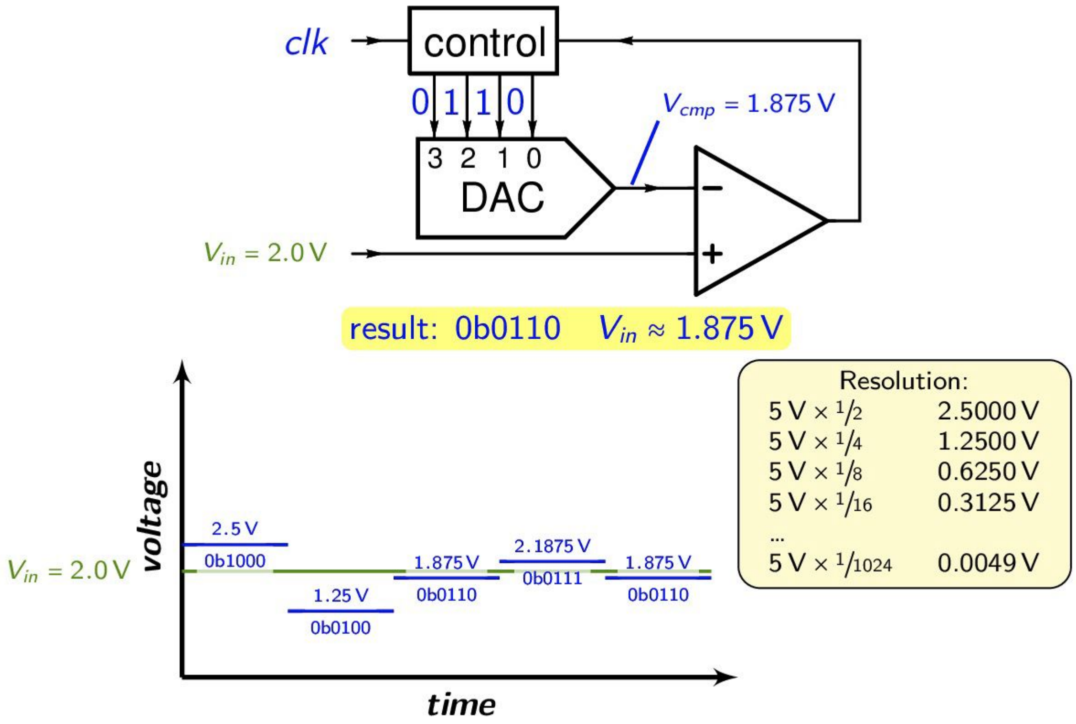
4-bit ADC Successive Approximation - Step 5





Step 5: DAC bit 1 kept; bit 0 turned on and compared.

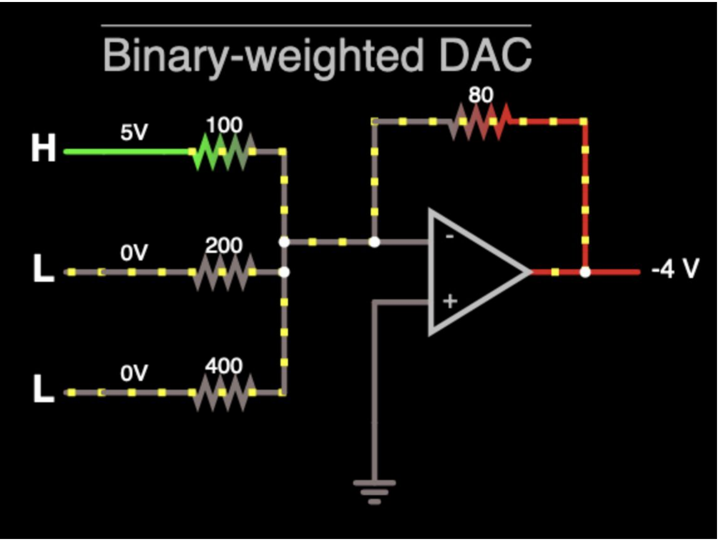
4-bit ADC Successive Approximation - Step 2

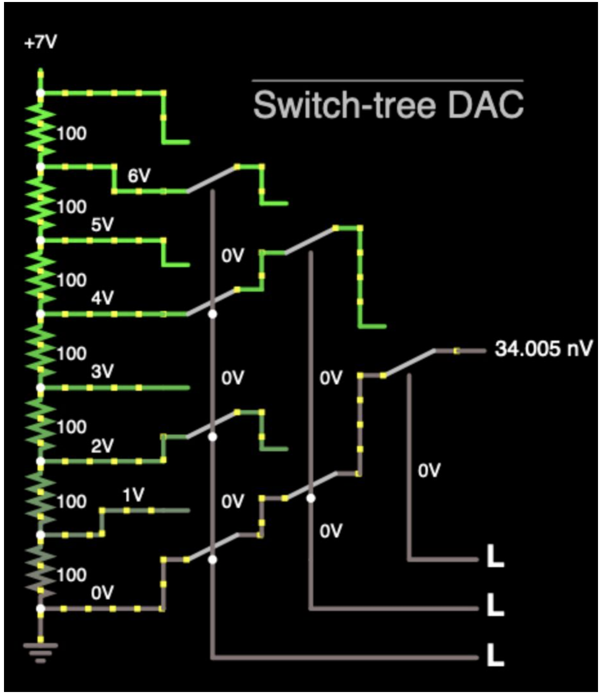


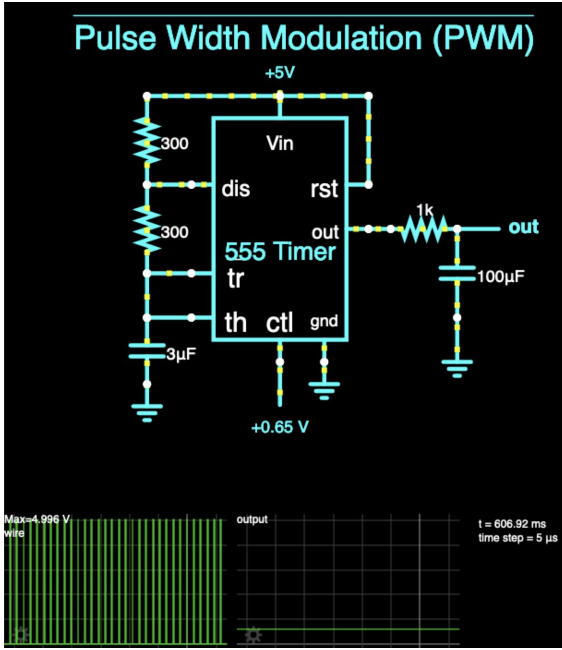
Step 6: DAC bit 0 turned off, and final result set.

Digital to Analog Converter (DAC)

Converts input digital value (i.e. an integer) into an output voltage.







Arduino Digital Pins

- Digital reads are much faster than analog reads
    - processor just needs to know whether the signal is ON or OFF.
- Arduino digitalRead
    - ~ 50 (16 MHz) clock cycles, 3.125μ s,
        - potentially up to 320 kHz.
    - Accessing the digital ports directly can read/write to the digital pins even faster
        - as little as 2 clock cycles → 8 MHz

Higher digitization rate

InfiniiVision Real-Time Oscilloscopes

50 MHz to 6 GHz

Infiniium Real-Time Oscilloscopes

500 MHz to 110 GHz

DCA Equivalent-Time (Sampling) Oscilloscopes

10 GHz to >100 GHz

Handheld, Modular, and USB Oscilloscopes

100 MHz to 1 GHz

Or… design it yourself…

Questions?