Now that you have a basic understanding of QRNG’s, lasers and beam splitters, you can try:

1. *Improving your beam splitter*: Experiment with different coatings, surface preparations, and angles to optimize your beam splitter's performance.

2. *Building a simple interferometer*: Use your beam splitter to create a simple interferometer, which can help you visualize the principles of wave interference.

3. *Exploring quantum randomness*: Once you have a reliable beam splitter, you can start exploring quantum randomness using photon detection and timing electronics.

Some consumer devices that might contain a photoelectric diode that can be reused:

1. CD/DVD/Blu-ray Drives:
Optical disc drives use photoelectric diodes to read data from the discs. You can salvage the photodiodes from old CD, DVD, or Blu-ray drives.

2. Laser Printers:
Some laser printers use photodiodes to detect the laser beam's position or to monitor the printer's internal sensors.

3. Optical Mice:
Older optical mice used photodiodes to detect movement and track the mouse's position.

4. TV Remote Controls:
Some TV remote controls use photodiodes to receive infrared signals from the remote.

5. Smoke Detectors:
Some smoke detectors use photodiodes to detect smoke particles or infrared radiation.

6. Security Systems:
Some security systems, like motion detectors or intrusion detectors, might use photodiodes to detect changes in the environment.

7. Old Scanners:
Flatbed scanners or document scanners might contain photodiodes to detect the scanned document's presence or to monitor the scanner's internal sensors.

When salvaging photodiodes from these devices, make sure to:

1. *Identify the correct component*: Verify that the component you're removing is indeed a photodiode.

2. *Handle the component carefully*: Photodiodes can be sensitive to static electricity, light, or physical damage.
3. *Test the component*: Before using the salvaged photodiode, test it to ensure it's functioning correctly.

Remember to always follow proper safety precautions when working with electronic components and devices. Remember Alderaan!

Arduino Support:
Arduino boards can work with photoelectric diodes, and there are various libraries available to simplify the process. You can connect a photoelectric diode to an Arduino board and use it to detect light or infrared signals.

Raspberry Pi Support:
Raspberry Pi boards also support photoelectric diodes, and you can connect them to the board's GPIO pins. There are Python libraries available, such as RPi.GPIO, that make it easy to work with photoelectric diodes on the Raspberry Pi.

Pre-built Modules:
If you prefer a more plug-and-play solution, there are pre-built modules available for both Arduino and Raspberry Pi that incorporate photoelectric diodes. These modules often include additional features like amplifiers or filters to improve the signal quality.

Some examples of photoelectric diode modules include:

- *Photodiode Modules*: These modules usually contain a photodiode, a resistor, and sometimes an amplifier. They can be connected directly to the Arduino or Raspberry Pi boards.

- *IR Receiver Modules*: These modules are designed specifically for infrared communication and often include a photodiode, a filter, and an amplifier. They're commonly used in remote control applications.

Keep in mind that when working with photoelectric diodes, you'll need to consider factors like the type of diode, the wavelength of light it's sensitive to, and the circuit design to ensure optimal performance.

Further Info and credits: https://programmingdigest.com/photodiode-phototransistor-and-ir-sensor-with-arduino/

Back to quantum number generation. After connecting two photoelectric diodes to an arduino board, and splitting a single laser beam onto each diode, we look for the signal. What type of signal will you be looking for prior to processing in order to infer random quantum numbers? Don't most diodes provide a yes/no response to light?

Photodiodes typically provide a binary response to light, indicating whether light is present or not. However, when working with quantum randomness, you're interested in exploiting the inherent noise and fluctuations in the photodiode's response.

Signal Characteristics:
To infer random quantum numbers, you'll be looking for:

1. *Signal Noise*: Random fluctuations in the photodiode's current due to the quantized nature of light.
2. *Photon Arrival Times*: The time intervals between consecutive photon arrivals, which can be used to generate random numbers.

Photodiode Response:
When a photodiode detects photons, it generates a current pulse. The pulse's:

1. *Amplitude*: Will vary randomly due to shot noise.
2. *Timing*: Will be random due to the Poisson distribution of photon arrivals.

Signal Processing:
To extract random quantum numbers, you'll need to process the photodiode's signal using:

1. *Amplification*: To increase the signal-to-noise ratio.
2. *Filtering*: To remove unwanted noise and frequencies.
3. *Thresholding*: To convert the analog signal into a digital signal.
4. *Time-to-Digital Conversion*: To measure the time intervals between photon arrivals.

Arduino Implementation:
On an Arduino board, you can use:

1. *Analog-to-Digital Conversion (ADC)*: To digitize the photodiode's signal.
2. *Interrupts*: To detect the rising or falling edges of the photodiode's signal.
3. *Timing Functions*: To measure the time intervals between photon arrivals.

Random Number Generation:
Once you've processed the photodiode's signal, you can use various algorithms to generate random numbers, such as:

1. *Von Neumann Extractor*: To extract random bits from the photodiode's signal.
2. *Hash Functions*: To generate random numbers from the photodiode's signal.

Keep in mind that generating truly random numbers from quantum fluctuations requires careful consideration of the experimental setup, signal processing, and algorithmic extraction.

The noise in the signal is the key. So really we don't want to filter the noise or amplify the on/off signal of the diode, rather we are wanting to focus on signal anomalies. In quantum random number generation, the noise in the signal is the key to generating truly random numbers. This noise is a result of the inherent randomness in the quantum world and the quantum field.

To generate truly random numbers, you want to preserve the entropy in the signal, which is a measure of the uncertainty or randomness. Filtering or amplifying the signal can reduce the entropy, making the signal less random.

Raw Signal Acquisition:
To capture the noise and anomalies, you'll want to acquire the raw signal from the photodiode, without applying any significant filtering or amplification. This will allow you to capture the inherent randomness in the signal.

Post-Processing:
After acquiring the raw signal, you can apply post-processing techniques to extract the random numbers. This might involve:

1. *Signal processing*: Techniques like Fourier analysis or wavelet transforms to extract the noise and anomalies.
2. *Entropy extraction*: Algorithms like the Von Neumann extractor to extract the entropy from the signal.
3. *Random number generation*: Using the extracted entropy to generate truly random numbers.

What is truly random? Complexity can appear to be randomness but complexity can be understood with sophisticated tools. As we study the Quantum Field and interact with it we approach a deeper understanding of true randomness in our world, and how to harness it for our safety and security.

- Adrian.